ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р ИСО/ HL7 27932— 2015

Информатизация здоровья

СТАНДАРТЫ ОБМЕНА ДАННЫМИ

Архитектура клинических документов HL7. Выпуск 2

(ISO/HL7 27932:2009, Data Exchange Standards — HL7 Clinical Document Architecture, Release 2, IDT)

Издание официальное

Москва
Стандартинформ
2016

Предисловие

1 ПОДГОТОВЛЕН Федеральным государственным учреждением «Центральный научно-исследовательский институт организации и информатизации здравоохранения Министерства здравоохранения Российской Федерации» (ЦНИИОИЗ Минздрава РФ) и Федеральным бюджетным учреждением «Консультационно-внедренческая фирма в области международной стандартизации и сертификации «Фирма «ИНТЕРСТАНДАРТ» на основе собственного аутентичного перевода на русский язык стандарта, указанного в пункте 4

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 468 «Информатизация здоровья» при ЦНИИОИЗ Минздрава РФ — постоянным представителем ИСО ТС 215

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 28 декабря 2015 г. № 2224-ст

4 Настоящий стандарт идентичен международному стандарту ИСО/HL7 27932:2009 «Стандарты обмена данными. Архитектура клинических документов HL7. Выпуск 2» (ISO/HL7 27932:2009 «Data Exchange Standards — HL7 Clinical Document Architecture. Release 2»).

Наименование настоящего стандарта изменено относительно наименования указанного международного стандарта для приведения в соответствие с ГОСТ Р 1.5 (пункт 3.5).

При применении настоящего стандарта рекомендуется использовать вместо ссылочных международных стандартов соответствующие им национальные стандарты Российской Федерации, сведения о которых приведены в дополнительном приложении ДА

5 ВВЕДЕН ВПЕРВЫЕ

Информация об изменениях к настоящему стандарту публикуется в ежегодно издаваемом информационном указателе «Национальные стандарты», а текст изменений и поправок — в ежемесячно издаваемых информационных указателях «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ежемесячно издаваемом информационном указателе «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (www.gost.ru)

© Стандартинформ, 2016

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения 1
2 Нормативные ссылки 1
3 Термины и определения. 1
4 Сокращения 2
5 Архитектура клинических документов HL7, выпуск 2 3
5.1 Обзор АКД 3
5.2 Введение в технические артефакты АКД 12
5.3 Передача АКД-документов в сообщениях HL7 14
5.4 Модель АКД R-MIM 17
5.5 Иерархическое описание структуры АКД-документов 67
5.6 Реализация АКД на языке XML 67
5.7 Примеры. 68
5.8 Указания по реализации 69
6 Иерархическое описание АКД-документов и графическое представление модели R-MIM 99
6.1 Иерархическое описание АКД-документов (POCD_HD000040) 99
6.2 Графическое представление модели АКД R-MIM. 137
Приложение A (справочное) Эталонная информационная модель HL7 138
Приложение B (справочное) Типы данных. Абстрактная спецификация 222
Приложение C (справочное) Словарные домены HL7 366
Приложение D (справочное) Опечатки в АКД, выпуск 2 (по состоянию на 20 января 2009 г.) 374
Приложение E (справочное) Примеры АКД-документов 376
Приложение F (справочное) XML-схемы 434
Приложение G (справочное) Руководство по стандартам HL7, версия 3 518
Приложение ДА (справочное) Сведения о соответствии ссылочных международных стандартов национальным стандартам Российской Федерации 576

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информатизация здоровья

СТАНДАРТЫ ОБМЕНА ДАННЫМИ

Архитектура клинических документов HL7. Выпуск 2

Health informatics. Data Exchange Standards. HL7 Clinical Document Architecture, Release 2

Дата введения — 2016—11—01

1 Область применения

1.1 Архитектура клинических документов HL7, выпуск 2

Областью применения архитектуры клинических документов (АКД) является стандартизация клинических документов для целей обмена.

Формат данных клинических документов за пределами контекста обмена (например, используемый для хранения клинических документов) в настоящем стандарте не затрагивается.

АКД-документы могут быть переданы в сообщениях стандартов HL7, предназначенных для передачи клинических документов. Хотя детальная спецификация таких сообщений не входит в область применения АКД, настоящий стандарт предъявляет определенные требования к вложению АКД-документов в сообщения стандартов HL7.

В АКД обсуждается не создание документов или управление документами, а только разметка содержания документов в целях обмена. Хотя XML-схему АКД можно непосредственно использовать в программном обеспечении, предназначенном для конструирования документов, эта задача не является первоочередной для стандарта на АКД.

Управление документами взаимозависит со спецификациями АКД, но спецификация сообщений, предназначенных для управления документами, не входит в область применения АКД.

2 Нормативные ссылки

В настоящем стандарте использованы ссылки на следующие документы (для датированных ссылок следует использовать только указанное издание, для недатированных ссылок следует использовать последнее издание указанного документа, включая все поправки):

ИСО/HL7 21731:2006 Информатизация здоровья. Версия 3 HL7. Эталонная информационная модель. Выпуск 1 (ISO/HL7 21731:2006, Health informatics — HL7 version 3 — Reference information model — Release 1)

HL7 2008 Спецификация типов данных абстрактных типов данных (HL7 2008 Datatype Specification Abstract Data Types)

3 Термины и определения

Для описания базовых понятий здравоохранения, предназначенных для различных целей, организации ИСО, CEN, HL7, а также различные национальные и международные организации используют большое число разных терминов. Поэтому в настоящем стандарте не делается попытки заставить эти организации принять термины и определения, приведенные в настоящем стандарте. Они должны ис-

Издание официальное

пользоваться в сочетании с национальными или региональными требованиями и в случае конфликта национальные или региональные требования имеют преимущество. Учитывая это, для целей настоящего стандарта используются следующие термины и определения, облегчающие проверку интероперабельности и соответствия настоящему стандарту.

В конце настоящего стандарта приведены ссылки на словарь HL7.

Следующие термины были выбраны в связи с их релевантностью Архитектуре клинических документов.

3.1 **архитектура клинических документов** (clinical document architecture): Стандарт разметки документа, описывающий структуру и семантику клинических документов для целей их передачи.

3.2 **определение типа документа** (Document Type Definition, DTD): Определение типа XML-документа, содержащее по значению или по ссылке объявления разметки, задающие грамматику класса документов.

3.3 **медицинский работник** (healthcare professional): Лицо, наделенное правом прямого или косвенного предоставления определенных медицинских услуг субъекту медицинской помощи или популяции субъектов.

Пример — Квалифицированный медицинский работник, фармацевт, медицинская сестра, социальный работник, рентгенолаборант, медрегистратор.

[ENV 1613:1995], [ИСО 21574-7]

3.4 **регуляторный орган или уполномоченный орган** [regulatory agency (or authorities)]: Геополитические сущности создают агентства или органы, ответственные за регулирование обращения лекарственных средств и медицинских изделий. Собирательно такие агентства или органы называются регуляторными органами.

3.5 **эталонная информационная модель** (Reference Information Model, RIM): Информационная модель, разработанная комитетом HL7 и используемая для разработки всех остальных информационных моделей, например, RMIM, а также структуры сообщений.

3.6 **уточненная информационная модель сообщений** (Refined Message Information Model, RMIM): Информационная структура, описывающая требования к комплексу сообщений.

3.7 **стандарт** (standard): Техническая спецификация, определяющая комплекс требований к процессам деятельности, реализуемая в жизнеспособных промышленных изделиях или товарах, в известной степени практичная и утвержденная признанной организацией по разработке стандартов, например, ИСО.

3.8 **XML:** Расширяемый язык разметкм XML (eXtensible Markup Language), представляющий собой простой и очень гибкий формат текста, произведенный от языка SGML (ИСО 8879). Этот язык первоначально разработан для целей широкомасштабной электронной публикации, а затем стал играть все возрастающую роль в обмене разнообразными данными в сети Интернет и других коммуникационных средах.

3.9 **XML-схема** (XML Schema): XML-схемы задают словари общего пользования и позволяют машинам проверять правила, определенные людьми. XML-схемы предоставляют средства более детального описания структуры, содержания и семантики XML-документов.

4 Сокращения

Для целей настоящего стандарта применяются следующие сокращения терминов:

ISO — International Organization for Standardization (Международная организация по стандартизации, ИСО);

CDA — Clinical Document Architecture (архитектура клинических документов, АКД);

CEN — Comité Européen de Normalisation (European Committee for Standardization, a federation of 28 national standards bodies that are also ISO member bodies) (Европейский комитет по стандартизации, федерация из 28 национальных органов стандартизации, которые также являются членами ИСО);

EU — European Union (Европейский Союз);

HL7 — Health Level 7;

HMD — Hierarchical Message Description (иерархическое описание сообщений);

ICH — The International Conference on Harmonisation of Technical Requirements for Registration of Pharmaceuticals for Human Use (Международная конференция по гармонизации технических требований к регистрации лекарственных средств, предназначенных для применения к человеку);

ITS — Implementable Technology Specification (спецификация реализуемой технологии);

RIM — Reference Information Model (эталонная информационная модель);

RMIM — Refined Message Information Model (уточненная информационная модель сообщений);

SDO — Standards Development Organization (организация по разработке стандартов);

SGML — Standardized generalized markup language. An ISO standard for describing structured information in a platform independent manner (стандартизованный обобщенный язык разметки. Стандарт ИСО, предназначенный для платформенно-независимого описания структурированной информации);

SNOMED — The Systematized Nomenclature of Human and Veterinary Medicine (систематизированная номенклатура медицины и ветеринарии);

SNOMED-CT — Systematized Nomenclature of Medicine-Clinical Terms (систематизированная номенклатура медицины — клинические термины);

UML — Unified Modelling Language (унифицированный язык моделирования);

W3C — World Wide Web Consortium (консорциум World Wide Web);

XML — eXtensible Markup Language (расширяемый язык разметки).

5 Архитектура клинических документов HL7, выпуск 2

ANSI/HL7 CDA, R2-2005 4/21/2005

Основной/технический редактор: Robert H. Dolin, MD, BobDolin@gmail.com, Semantically Yours, LLC

Основной/технический редактор: Liora Alschuler, Liora@alschulerassociates.com, Alschuler Associates

Основной/технический редактор: Sandy Boyer, BSP, slboyer@attglobal.net, Consultant

Основной/технический редактор: Calvin Beebe, cbeebe@mayo.edu, Mayo Clinic

Технический редактор: Fred M. Behlen, PhD, fbehlen@laitek.com, American College of Radiology

Технический редактор: Paul V. Biron, Paul.V.Biron@kp.org

Технический редактор: Amnon Shabo (Shvo), PhD, SHABO@il.ibm.com, IBM Research Lab in Haifa

5.1 Обзор АКД

5.1.1 Что такое АКД

Архитектура клинических документов HL7 (АКД) является стандартом разметки, описывающим структуру и семантику клинических документов в целях их передачи. Клинический документ представляет собой совокупность сведений о состоянии здоровья, запланированной и оказанной медицинской помощи одному пациенту, обладающую следующими свойствами:

1) Постоянство. Клинический документ существует в неизменном состоянии в течение срока, определенного местными требованиями и нормативными документами (срок существования клинического документа не зависит от срока существования его электронной копии, представленной на языке XML в виде АКД-документа).

2) Ответственность. Ответственность за клинический документ возлагается на медицинскую организацию, оказывающую пациенту медицинскую помощь.

3) Потенциальная юридическая значимость. Клинический документ представляет собой совокупность сведений, предназначенную для придания им юридической значимости.

4) Контекст. Клинический документ идентифицирует определенный контекст, в котором по умолчанию представлено его содержание.

5) Целостность. Клинический документ аутентифицируется как единое целое, а не отдельными частями вне полного содержания документа.

6) Восприятие человеком. Клинический документ должен восприниматься человеком.

АКД-документ является определенным и полным информационным объектом, который может включать в себя текст, изображения, аудиозаписи и другое мультимедийное содержание.

5.1.1.1 Ключевые свойства АКД

АКД характеризуется следующими ключевыми свойствами:

АКД-документы кодируются с использованием расширяемого языка разметки XML.

Примечание — Если понадобятся альтернативные реализации, то будут выпущены соответствующие требования по соответствию, поэтому в будущем синтаксис может не ограничиваться только языком XML.

Структура машинно-обрабатываемого содержания АКД произведена из Эталонной информационной модели HL7 (Reference Information Model — RIM) и использует типы данных HL7 версии 3.

Спецификация АКД является очень гибкой и предоставляет широкие возможности описания документов. Для наложения ограничений на базовую спецификацию АКД могут использоваться шаблоны уровня документа, раздела и подраздела (см. 5.1.2.2).

5.1.1.2 Область применения АКД

Областью применения АКД является стандартизация передаваемых клинических документов.

Формат данных клинических документов вне контекста обмена (например, формат хранения клинических документов) в данной спецификации не рассматривается.

АКД-документы могут передаваться в сообщениях HL7, предназначенных для передачи медицинских документов. Хотя детальная спецификация таких сообщений не входит в область применения настоящего стандарта, в нем все же накладываются определенные требования на передачу АКД-документов в сообщениях HL7 (см. 5. 3).

В АКД не рассматриваются вопросы создания или управления документами. Она только определяет их разметку для передачи. Хотя XML-схемы АКД можно использовать непосредственно в среде конструирования документов, это не является основной целью ее спецификации.

Управление документами тесно взаимосвязано со спецификациями АКД, но спецификация сообщений по управлению документами не входит в область ее применения (см. 5.1.2.6).

Примечание — Несколько комитетов разрабатывают спецификации структурированных документов, которые частично перекрываются со спецификацией АКД. Технический комитет Structured Documents в сотрудничестве с комитетом Publishing и другими комитетами готовит главу «Инфраструктура структурированных документов», поясняющую связи между этими разработками. Ее предполагается включить в предстоящие издания.

5.1.1.3 Цели и принципы разработки

Разработка АКД преследует следующие цели:

- дать приоритет предоставлению медицинской помощи;
- обеспечить экономически эффективную реализацию электронных медицинских документов в максимально широком спектре информационных систем;
- обеспечить обмен человеко-читаемыми документами между пользователями с разным уровнем технической грамотности;
- способствовать длительному хранению всей информации, закодированной в соответствии с этой архитектурой;
- способствовать применению большого числа приложений, обрабатывающих документы после их получения;
- обеспечить совместимость с широким кругом приложений по созданию документов;
- способствовать обмену, не зависящему от механизмов, лежащих в основе передачи и хранения информации;
- обеспечить возможность достаточно быстрой разработки;
- предоставить лицам, принимающим решения, возможность предъявлять собственные требования к информации без расширения данной спецификации.

Из указанных выше целей вытекает следующий ряд принципов разработки:

- данная архитектура должна быть совместима с языком XML и моделью HL7 RIM;
- она должна быть совместима с представлениями клинической информации, вытекающими из разработок, осуществляемых другими комитетами HL7;
- технические барьеры по ее использованию должны быть минимизированы;
- архитектура описывает представление документов, необходимое для их передачи;
- она должна накладывать минимальные ограничения или требования к структуре и содержанию передаваемого документа;
- она должна быть расширяемой, чтобы позволять детальную разметку, требуемую для высоко структурированного текста или кодированных данных;

- спецификации документов, основанные на этой архитектуре, должны удовлетворять ограничениям и требованиям, предъявляемым соответствующими профессиональными, коммерческими и регулирующими организациями;

- спецификации создания и обработки документов, рассчитанные на их передачу, должны соответствовать данной архитектуре;

- должна быть обеспечена возможность чтения АКД-документов человеком, использующим общедоступные XML-совместимые веб-обозреватели, драйверы принтеров и типовую таблицу стилей АКД, написанную на стандартном языке описания стилей;

- должны использоваться открытые стандарты.

5.1.2 Общие понятия АКД

5.1.2.1 Основные компоненты АКД-документа

Этот подраздел служит обзорным представлением основных компонентов АКД-документа, каждый из которых будет описан повторно с большей степенью детализации. Его целью является ознакомление с высокоуровневыми понятиями, облегчающее понимание следующих подразделов.

В следующем скелетном примере показаны основные компоненты прототипа АКД-документа. (Для его упрощения многие обязательные компоненты опущены. Подробный пример, соответствующий спецификации, приведен в 5.7.

Содержание АКД-документа вложено в элемент <ClinicalDocument> и состоит из заголовка (см. 5.4.2) и тела (см. 5.4.3). Заголовок лежит между элементами <ClinicalDocument> и <structuredBody>, он идентифицирует и классифицирует документ и предоставляет информацию о его аутентификации, о случае медицинской помощи, о пациенте и других лицах и организациях, имеющих отношение к оказанию ему медицинской помощи.

Тело содержит клинические данные. Оно может представлять собой неструктурированный большой объект двоичных данных (BLOB — Binary Large Object Block) или содержать структурированную разметку. В примере показано структурированное тело, помещенное в элемент <structuredBody> и подразделяемое на рекурсивно вложенные разделы документа.

Содержание раздела АКД-документа помещено в элемент <section>. Оно может включать в себя единственный повествовательный блок (см. 5.4.3.5), любое количество подразделов АКД (см. 5.4.3.6) и внешних ссылок.

Повествовательный блок размещен в элементе <text> внутри элемента <section> и должен содержать человекочитаемое наполнение. Принципы представления повествовательного блока, требования к соответствию, предъявляемые к создателям его содержания и читающим его получателям см. в подразделах «Человекочитаемость и визуализация документов АКД» (см. 5.1.2.3) и «Соответствие АКД» (см. 5.1.3).

Повествовательный блок раздела документа содержит данные, предназначенные для непосредственного отображения, тогда как вложенные подразделы представляют собой структурированные данные, подлежащие дальнейшей компьютерной обработке (например, в приложениях поддержки принятия решений). Вложенные подразделы, как правило, кодируют содержимое повествовательного блока того же самого раздела. В примере показаны два вложенных подраздела: <observation> (исследование) и <substanceAdministration> (применение лекарственного средства), который в свою очередь содержит вложенный подраздел <supply> (запас). Присутствуют и некоторые другие вложенные подразделы.

Вложенные подразделы АКД-документа могут вкладываться друг в друга и ссылаться на внешние объекты. Внешние ссылки всегда включены в контекст вложенного подраздела. Они ссылаются на содержание, находящееся вне данного АКД-документа — например, на какое-то другое изображение, другую процедуру или другое исследование (идентификация которого указана в элементе <externalObservation> (внешнее исследование>)). Внешнее содержание не покрывается аутентификацией документа, который на него ссылается.

Пример 1 —

```
<ClinicalDocument>
 ... Заголовок АКД-документа ...
 <structuredBody>
  <section>
   <text>...</text>
   <observation>...</observation>
```

```
    <substanceAdministration>
      <supply>...</supply>
    </substanceAdministration>
    <observation>
      <externalObservation>...
      </externalObservation>
    </observation>
  </section>
  <section>
    <section>...</section>
  </section>
 </structuredBody>
</ClinicalDocument>
```

5.1.2.2 «А» в АКД

Для достижения целей, перечисленных в 5.1.1.3, понятие уровней, введенное в первом выпуске АКД, предполагало иерархию определений структуры документа XML DTD или XML-схем. Эта иерархия формировала «архитектуру», отсюда и появилась буква «А» в сокращении АКД.

Хотя смысл понятия уровня во втором выпуске АКД остался прежним, подход к представлению иерархий изменился. Текущая спецификация состоит из единой XML-схемы АКД, а архитектура обеспечивается возможностью применения одного или нескольких иерархического наборов шаблонов HL7, ограничивающих широту и гибкость АКД.

Примечание — Для ограничения спецификации АКД могут использоваться механизмы, определенные в документе HL7 V3 Refinement and Localization («Уточнение и локализация HL7 V3»). Во время разработки настоящего стандарта технические формализмы HL7 (например, спецификации шаблонов HL7, формата взаимообмена моделями HL7), обеспечивающие возможности ограничения спецификации АКД, находились в стадии разработки.

Класс InfrastructureRoot модели RIM содержит атрибут templateID, который доступен для использования в АКД. Таким образом, пока шаблоны HL7 находятся в стадии разработки, в АКД уже имеется возможность указать ссылку на шаблон или руководство по реализации, которому был присвоен уникальный идентификатор. Пока не будет составлена формальная спецификация шаблонов HL7, не существует стандартизованного процесса для проверки соответствия шаблонов, на которые даются ссылки.

Не требуется обязательно ограничивать спецификацию АКД. В реализациях, использующих элементы структурированных данных для управления автоматизированными процессами, обычно требуется, чтобы либо (1) эти элементы были ограничены с помощью соответствующей уточненной модели или другого языка ограничений, принятым в HL7, либо (2) эти элементы соответствовали детальному руководству по реализации, описывающему способ, которым должны быть представлены структурированные элементы и их предполагаемая интерпретация, в таком объеме, чтобы клиническая безопасность обеспечивалась на уровне, соответствующем конкретному варианту использования.

Существует много типов шаблонов HL7, которые могут быть созданы. Среди них две группы особенно важны для клинических документов: (1) те шаблоны, которые ограничивают разделы документа в зависимости от типа документа (шаблоны уровня раздела); (2) те шаблоны, которые ограничивают подразделы (entries), вложенные в разделы документа (шаблоны уровня подраздела). В таблице 1 приведено сопоставление между определением уровней, приведенном в первом выпуске АКД, и текущим определением, основанном на этих двух типах шаблонов HL7.

Таблица 1 — Эволюция понятий уровня от первого ко второму выпуску АКД

АКД, выпуск 1	АКД, выпуск 2
Уровень 1 (CDA Level One)	Спецификация АКД без дополнительных ограничений.
Уровень 2 (CDA Level Two)	Спецификация АКД, в которой заданы шаблоны уровня разделов.
Уровень 3 (CDA Level Three)	Спецификация АКД, в которой заданы шаблоны уровня подразделов (и необязательно шаблоны уровня разделов).

Ниже приведена иллюстрация одной из возможных иерархий АКД с шаблонами HL7.

Пример 2 — Схема АКД

- ***Схема АКД :: применен шаблон «Дневниковая запись» уровня раздела.***
  - ***Схема АКД :: применен шаблон «Дневниковая запись» уровня раздела и шаблон «Жизненно важные показатели» уровня подраздела.***
    - ***Схема АКД :: применен шаблон «Дневниковая запись эндокринолога» уровня раздела и шаблон «Жизненно важные показатели» уровня подраздела.***
    - ***Схема АКД :: применен шаблон «Дневниковая запись» уровня раздела и шаблон «Жизненно важные показатели в блоке интенсивной терапии» уровня подраздела.***
- ***Схема АКД :: применен шаблон «Дневниковая запись кардиолога» уровня раздела.***
  - ***Схема АКД :: применен шаблон «Дневниковая запись кардиолога» уровня раздела и шаблон «Осмотр кардиолога» уровня подраздела.***
- ***Схема АКД :: применен шаблон «Дневниковая запись эндокринолога» уровня раздела.***
  - ***Схема АКД :: применен шаблон «Дневниковая запись эндокринолога» уровня раздела и шаблон «Жизненно важные показатели» уровня подраздела.***

5.1.2.3 Человекочитаемость и визуализация документов АКД

Требование человекочитаемости АКД-документов гарантирует, что получатель документа сможет алгоритмически обработать и отобразить клиническое содержание документа в стандартном веб-обозревателе. Сочетание повествовательных блоков и подразделов, допускаемое во втором выпуске АКД, создает определенные сложности для выполнения данного требования.

Среди требований, влияющих на разработку второго выпуска АКД, можно выделить следующие:

- получатель произвольного АКД-документа должен иметь определенный способ оценки правильности его содержания;

- для обеспечения человекочитаемости отправитель не должен передавать особую таблицу стилей вместе с АКД-документом. Должна существовать возможность визуализации АКД-документов с помощью единой таблицы стилей и общедоступных инструментов отображения;

- требование человекочитаемости относится к заверенному содержанию. В документе может передаваться дополнительная информация, предназначенная, в первую очередь, для машинной обработки, которая не включается в заверенное содержание и не нуждается в визуализации;

- если структурированное содержание является производным от содержания повествовательного блока, то должен существовать способ описания процесса, по которому машинно-обрабатываемые части были получены из повествовательного блока (например, они произведены автором, специальным сотрудником, кодирующим данные, с помощью алгоритмов обработки предложений естественного языка, с помощью специального программного обеспечения);

- если содержание повествовательного блока является производным от структурированного содержания, то должен существовать способ идентификации процесса, с помощью которого содержание повествовательного блока было получено из структурированных данных.

Эти принципы и требования легли в основу текущего подхода, при котором материал, предназначенный для визуализации, помещается в поле Section.text (см. Блок свободного текста секции (см. 5.4.3.5)). Модель содержания этого поля специально сконструирована таким образом, чтобы выполнялись указанные выше требования. Она аналогична модели содержания разделов, использованной в первом выпуске АКД. Структурированные результаты исследований могут содержать ссылки на повествовательное содержание, помещенное в поле Section.text. Мультимедийные результаты исследований передаются вне поля Section.text, а тег <renderMultiMedia>, включаемый в поле Section.text, показывает, где должны отображаться мультимедийные результаты, на которые в нем дается ссылка.

5.1.2.4 XML-разметка АКД-документов

Настоящая спецификация описывает XML-разметку АКД-документов. Экземпляры АКД-документов проверяются на соответствие XML-схеме АКД и могут быть объектами дополнительных проверок (см. 5.1.3). Не существует запретов на использование различных языков описания схем (например, W3C, DTD, RELAXING), коль скоро экземпляры документов, соответствующие схеме, остаются совместимыми.

При конструировании схемы АКД-документов использовались следующие принципы:

- использование общих требований: схема АКД-документов соответствует более общим требованиям к АКД (см. 5.1.1.3);

- согласование схемы АКД-документов со Спецификацией технологии реализации (ITS) V3: схема АКД-документов соответствует общей спецификации технологии реализации стандарта HL7 V3 на языке XML (V3 XML ITS);

- отображение на модель RIM: схема АКД-документов описывает стиль XML-разметки экземпляров АКД-документов для цели обмена. Ее нельзя понять вне контекста этой определяющей спецификации. В то же время схема АКД-документов самостоятельно применима для целей реализации, хотя и не предназначена для повторения или замещения уточненной информационной модели сообщений (R-MIM) и иерархического описания (HD). Таким образом, схема АКД-документов сама по себе не обеспечивает адекватное соответствие экземпляров документов модели HL7 RIM. Для обеспечения семантической интероперабельности экземпляров АКД-документов необходимо знать и использовать схему АКД-документов, модель R-MIM и иерархическое описание HD, а также соответствующую модель RIM;

- анализ документов: схема АКД-документов и соответствующие ей документы должны соответствовать требованиям к анализу документов, вытекающим из модели содержания;

Примечание — Анализ документа представляет собой процесс, который может рассматриваться как эквивалент анализа варианта использования. В этом процессе рассматривается документ или класс документов и анализируется их структура и содержание, нередко представляя его в виде дерева, записанного в нотации elm. При анализе документа также выявляются бизнес-правила жизненного цикла этого документа или класса документов. Традиционно анализ документа определяет модель содержания, общую структуру и стиль XML.

Анализ документа является итеративным шагом процесса конструирования модели его содержания — подход «снизу-вверх», дополняющий конструирование «сверху-вниз» из модели RIM. Тем самым обеспечивается не только соответствие схемы и экземпляров модели RIM, но и простота представления выявленных артефактов;

- прямая и обратная совместимость: схема АКД-документов должна удовлетворять требованиям прямой и обратной совместимости. (см. 5.1.5);

- присвоение имен: хотя по определению XML-разметка предназначена для машинной обработки, она должна быть оптимизирована для просмотра, отладки и конструирования человеком. Схема АКД-документов не является «самодокументируемой», но смысл содержания должен быть ясен из имени тега и документации (например, из отображения на модель RIM). В то же время человекочитаемое название тега не должно вводить в заблуждение;

- словари данных: словарные значения могут быть включены в схему АКД-документов или содержаться во внешнем источнике, на который дается ссылка. Предпочтительнее включать их в схему, когда их количество ограничено (не слишком велико) и они устойчивы (не пересматриваются между циклами голосования). В тех же случаях, когда словарь или слишком велик, или подвержен изменениям, предпочтительнее хранить его вне схемы АКД-документов и использовать по ссылке. В этом случае проверка документа на соответствие XML-схеме будет недостаточной для обеспечения его соответствия стандарту.

5.1.2.5 Безопасность, конфиденциальность и целостность данных

Прикладные системы, отправляющие и получающие АКД-документы, несут ответственность за соответствие всем нормативным требованиям, предъявляемым к аутентификации документов, их конфиденциальности и сохранности. В процессе взаимодействия по каналам передачи данных общего пользования могут потребоваться криптографические средства аутентификации источника/получателя и безопасной передачи вложенных документов. Для этих целей могут использоваться доступные коммерческие аппаратно-программные средства, что не входит в область применения настоящего стандарта.

В схеме АКД-документов предусмотрен статус конфиденциальности, используя который, прикладные системы могут управлять доступом к чувствительным данным. Статус конфиденциальности может применяться ко всему документу или к определенным фрагментам документа.

5.1.2.6 Связь АКД со стандартами передачи сообщений HL7

АКД-документ является определенным и законченным информационным объектом, который может существовать вне контекста сообщений и/или может быть помещен в сообщение HL7 (см. 5.3 Обмен документами АКД в сообщениях HL7). Таким образом, АКД дополняет спецификации сообщений HL7.

Клинические документы могут пересматриваться и добавляться к уже существующим документам. В идеале измененный документ должен быть признан устаревшим и содержать явное указание на более новую версию. Это должно снизить вероятность того, что поставщик медицинской помощи примет решение о лечении на основе ошибочных или неполных данных.

На практике, однако, невозможно гарантировать, что в устаревшую версию будет включена явная прямая ссылка на более новую версию. Без процесса отслеживания цепочки ответственности за содержание клинических документов и всех их копий, нет никакого способа гарантировать, что просматриваемый клинический документ не был впоследствии редактирован.

Для минимизации риска просмотра устаревшей информации необходимо использовать системы управления клиническими документами. Если АКД-документы просматриваются вне контекста системы управления документами, то нельзя знать наверняка, был ли просматриваемый документ впоследствии отредактирован. Сообщения HL7, в которых передаются АКД-документы (например, сообщения MDM в стандарте HL7 V2.x и Medical Records в стандарте HL7 V3), содержат критичную информацию о контексте, обеспечивающую точный просмотр клинических данных.

5.1.3 Соответствие стандарту АКД

Примечание — Полное обсуждение соответствия стандарту HL7 V3 см. в документе HL7 V3 Refinement and Localization.

Документом, соответствующим стандарту АКД, считается тот документ, который как минимум соответствует схеме АКД-документов и в котором использование кодированных данных ограничено значениями, разрешенными для соответствующих словарных доменов. Однако компьютерная программа не проверить каждый аспект соответствия. Цель данного раздела — выделить те аспекты соответствия стандарту АКД, которые не могут быть проверены автоматически, в особенности те, что связаны с требованиями человекочитаемости АКД-документов.

Создатель документа — роль приложения, создающего АКД-документ. Такой документ может быть создан с помощью преобразования из какого-либо другого формата, как непосредственный результат программы ввода документов и т. д. Создатель документа нередко ответственен за обмен документами с местом их постоянного хранения, для чего во многих случаях он может использовать сообщения MDM стандарта HL7 V2 или сообщения Medical Records стандарта HL7 V3. Создатель документа отвечает за полное соответствие созданных им АКД-документов настоящей спецификации.

Получатель документа — роль приложения, получающего информацию об измерении состояния документов и сами документы от создателя документа или системы управления документами. Получатель отвечает за то, что полученные документы АКД были визуализированы в полном соответствии с настоящей спецификацией.

Поскольку АКД представляет собой стандарт передачи и АКД-документ может не являться исходной формой документа, то настоящий стандарт не предъявляет никаких требований к постоянному хранению АКД-документов. Однако, как отмечалось выше в 5.1.2.6 «Связь АКД со стандартами передачи сообщений HL7», спецификация АКД существенно увязана с управлением документами. За управление исходным документом, который может храниться в отличной от АКД форме, отвечает владелец, идентифицированный в заголовке АКД-документа (см. 5.4.2.2.3).

5.1.3.1 Обязанности получателя

Получатель обязан:

1) использовать значения по умолчанию там, где они определены этой спецификацией, и где документ не содержит никакого значения: в случае, если в экземпляре АКД-документа отсутствует значение элемента или атрибута, для которого в спецификации АКД определено значение по умолчанию, получатель должен использовать это значение. (Примечание: значения по умолчанию обозначены в основной части этого документа как [по умолчанию]);

2) разбирать и интерпретировать полный заголовок АКД-документа: получатель АКД-документа должен быть способен разбирать и интерпретировать полный заголовок АКД-документа. Поскольку в приложениях демографические или другие данные заголовка АКД-документа могут извлекаться из центрального нормативно-справочного файла, то способ визуализации заголовка оставляется на усмотрение получателя. Кроме того, визуализация заголовка АКД-документа может зависеть от местной практики и контекста его использования (например, электронная медицинская карта, сценарий обезличивания). Если создатель документа хочет предложить свой вариант визуализации, то в дополнение к АКД-документу он может передать одну или несколько таблиц стилей XML. Использование этих таблиц стилей оставляется на усмотрение получателя;

3) разбирать и интерпретировать тело АКД-документа в объеме, достаточном для его визуализации: получатель АКД-документа должен быть способен разбирать и интерпретировать тело АКД-документа в объеме, достаточном для его визуализации, используя следующие правила:

- если тело документа не имеет XML-структуру, то для его визуализации должно использоваться программное обеспечение, которое способно обработать соответствующий тип среды MIME;

- если тело документа структурировано, то должно быть отображено название раздела, указанное в элементе Section.title. Отсутствие этого элемента означает, что раздел не имеет названия;

- если тело документа структурировано, то содержание элемента Section.text должно быть отображено в соответствии с правилами, указанными в 5.4.3.5.

Получатель АКД-документа не обязан:

1) разбирать и интерпретировать всю совокупность подразделов АКД-документа, содержащихся в его теле. По местным соглашениям взаимодействующие стороны могут назначать получателю дополнительные обязанности разбора и интерпретации различных подразделов;

2) проверять соответствие АКД-документа шаблонам, указанным в этом документе. По местным соглашениям взаимодействующие стороны могут назначать получателю дополнительные обязанности проверки соответствия шаблонам.

5.1.3.2 Обязанности создателя

Создатель обязан:

1) правильно конструировать повествовательные блоки АКД-документов: создатель обязан удостовериться, что заверяемая часть тела документа структурирована таким образом, что получатель, действуя в соответствии с описанными выше обязанностями, правильно отобразит документ. Сюда относятся следующие правила:

2) если тело документа структурировано, то название раздела должно передаваться в элементе Section.title. Отсутствие этого элемента означает, что раздел не имеет названия;

3) если тело документа структурировано, то заверяемое повествовательное содержание раздела должно быть помещено в элемент Section.text независимо от того, передается ли это содержание еще и в подразделах АКД-документа. Заверяемые ссылки на мультимедийное содержание должны быть включены в подразделы типа ObservationMedia или RegionOfInterest АКД-документа;

4) если тело документа структурировано, то содержание элемента Section.text должно соответствовать правилам, изложенным в подразделе 5.4.3.5.

Создатель АКД-документа не обязан:

1) полностью кодировать все повествовательные блоки тела АКД-документа в виде его подразделов. По местным соглашениям взаимодействующие стороны могут назначать создателю дополнительные обязанности создания различных подразделов.

5.1.4 Расширяемость АКД

Примечание — Полное обсуждение правил расширения XML-спецификаций стандарта HL7 V3 приведено в документе XML ITS — Informal Extensions.

Если местная семантика не имеет соответствующего представления в спецификации АКД, то может быть использована местная разметка. АКД стандартизует верхний уровень содержания общего пользования и в то же время обеспечивает ясный и стандартный механизм для разметки местного содержания. Для выполнения местных требований по расширяемости разрешается включать в схему АКД-документов дополнительные XML-элементы и атрибуты. Эти расширения не должны менять содержания ни одного из стандартных объектов данных, а получатели должны быть способны игнорировать дополнительные объекты. Получатели документа должны быть способны точно отображать содержание АКД-документа, игнорируя эти расширения.

Расширения могут быть включены в документ, используя пространство имен, отличающееся от пространства имен HL7v3, но при этом они не должны помещаться в элементы типа ED (напр., <text> внутри <procedure>), поскольку внутри документа, соответствующего стандарту, содержимое значения типа данных ED может быть в другом пространстве имен. Так как все содержание, соответствующее стандарту (внешнее по отношению к элементам с типом данных ED) находится в пространстве имен HL7, то отправитель может включить любое расширенное содержимое в чужое пространство имен (любое пространство имен, отличающееся от пространства имен HL7). При обнаружении таких расширений системы-получатели не должны сообщать об ошибке.

Когда эти механизмы расширения используются для разметки общезначимого содержания, то комитет HL7 предлагает пользователям формализовывать свои требования для включения в последующие версии стандарта с тем, чтобы максимизировать пользу от использования общей семантики.

5.1.5 Обратная и прямая совместимость

Примечание — Подробный список всех изменений, сделанных в АКД, выпуск 2 по сравнению с АКД, выпуск 1, можно найти в приложении (см. (п. В.4)).

Во втором выпуске АКД базовая модель по существу не изменилась. АКД-документ имеет заголовок и тело. Тело содержит вложенные разделы. Эти разделы могут быть закодированы с использованием стандартных словарей и содержать вложенные подразделы. Главными шагами эволюции ко второму выпуску АКД являются значительное расширение видов подразделов, которые могут включаться в разделы АКД-документов, а также приведение заголовка и тела в соответствие с моделью RIM. АКД, выпуск 2, обеспечивает возможность формально представления клинического содержания в том объеме, что смоделирован в RIM.

В настоящем подразделе описаны виды изменений, которые могут быть внесены в новый выпуск АКД, а также принципы прямой и обратной совместимости АКД. В общем случае в новый выпуск могут быть добавлены новые компоненты; существующие компоненты (включая имена XML-элементов и атрибутов в схеме АКД-документов) могут быть переименованы; часть компонентов, определенных в предыдущих выпусках, может быть запрещена; могут быть внесены изменения во множественность компонента (как в сторону сужения, так и расширения) или в словарь его домена значений (добавлены или изменены значения, квалификатор расширяемости CWE может быть заменен на CNE и наоборот). Изложенный ниже набор руководящих принципов определяет возможный путь развития АКД, по возможности, защищая усилия, приложенные разработчиками при использовании более ранних выпусков:

- документирование: в новом выпуске АКД будут перечислять все существенные отличия от предыдущего выпуска;
- заверяемое содержание: в заверяемом, человекочитаемом содержании не должно быть никаких потерь при переходе к новому выпуску. Обратная и прямая совместимость заверяемого содержания будет обеспечиваться таким образом, чтобы оставалась возможность автоматического преобразования человекочитаемого содержания в обоих направлениях;
- новые компоненты: в очередной выпуск АКД могут быть включены новые компоненты. Для сохранения возможности двустороннего отображения преобразование нового выпуска в предшествующий должно трактовать новые компоненты как расширения (например, как местную разметку или местное пространство имен);
- переименование: в новом выпуске АКД компоненты (включая имена XML-элементов и атрибутов) могут быть переименованы. В этом случае будет предоставлена таблица соответствия со списком всех изменений. При переименовании будут соблюдаться сформулированные выше соглашения об именовании. (см. 5.1.2.4);
- запрещенные компоненты: в новом выпуске АКД могут быть запрещены компоненты, определенные в предыдущем выпуске. Запрещенные компоненты будут удалены из следующего выпуска стандарта, поэтому их использование не рекомендуется;
- множественность: в новом выпуске АКД может быть изменена множественность компонента. Если необязательный компонент становится обязательным, то для него должно быть предусмотрено фиктивное или пустое значение;
- изменение словарного домена: в новом выпуске АКД может быть изменен словарный домен компонента. В этом случае будет предоставлена таблица преобразования со списком всех изменений;
- изменения словарного домена компонента с квалификатором расширяемости CNE: при удалении или переименовании значения компонента с квалификатором расширяемости CNE, будет предоставлена таблица преобразования, указывающая, каким должно быть текущее значение;
- изменения словарного домена компонента с квалификатором расширяемости CWE: при расширении словарного домена компонента с квалификатором расширяемости CWE пользователь должен начать использовать новые коды в дополнение к любым эквивалентным местным кодам, если таковые существуют;
- переход от квалификатора расширяемости CWE к квалификатору расширяемости CNE: чтобы обеспечить возможность двустороннего преобразования, в очередном выпуске нераспознанные компоненты должны быть представлены как расширения (например, используя местную разметку или местное пространство имен). В идеале эта ситуация должна разрешаться в процессе голосования по очередному выпуску, позволяя домену с квалификатором расширяемости CNE охватывать достаточное число значений.

Следование этим руководящим принципам привело к текущему подходу обеспечения совместимости, определенному в настоящем, втором выпуске стандарта АКД. Цель — обеспечение возможности преобразования экземпляров документов, созданных с использованием первого выпуска, в экземпляры, минимально соответствующие второму выпуску стандарта, а также возможности обратного преобразования полученных экземпляров, соответствующих второму выпуску, в экземпляры, соответствую-

щие первому выпуску, с помощью автоматических средств (преобразований) без потери заверенного человекочитаемого содержания и с известными ограничениями на потерю универсальной семантики при обработке.

5.2 Введение в технические артефакты АКД

Для полного понимания АКД необходимо ознакомиться с нормативными артефактами, используемыми в определениях спецификации. Иерархическое описание АКД-документов (Hierarchical Description) является определяющим источником правил соответствия стандарту АКД и служит источником для конструирования схемы АКД-документов. Помимо того, что экземпляр АКД-документа должен соответствовать этой схеме, он должен также удовлетворять правилам соответствия, приведенным в Иерархическом описании АКД, которое выводится из модели R-MIM, построенной для АКД, а та, в свою очередь, выводится из Эталонной информационной модели (RIM) HL7, являющейся наиболее полным источником определений классов и их атрибутов.

В следующих подразделах приводится краткое резюме по артефактам, используемым в АКД, и их использованию. Они рассчитаны на тех, кто желает реализовать или понять спецификацию АКД.

5.2.1 Эталонная информационная модель HL7

Полное описание Эталонной информационной модели HL7 (Reference Information Model; RIM) можно найти в ГОСТ ИСО/HL7 21731-2013.

Модель HL7 RIM является наиболее полным справочным источником, определяющим классы и атрибуты. Спецификация АКД не копирует исчерпывающим образом определения модели RIM, а вместо этого отсылает читателя к модели RIM для получения полных определений. Хотя АКД может дополнительно ограничивать определения модели RIM, определения АКД никогда не конфликтуют с определениями модели RIM.

Стандарт АКД, выпуск 2, получен из версии 2.07 модели HL7 RIM.

При необходимости посмотреть полное определение атрибута или класса модели RIM, следует обратиться к ее спецификации.

5.2.2 Типы данных HL7 V3

Комитет HL7 разработал как абстрактную спецификацию типов данных, являющуюся наиболее полной справочной информацией, так и специфическое представление типов данных на языке XML.

Типы данных определяют структурный формат данных, которые могут быть значениями атрибута модели RIM, и описывают набор разрешенных значений атрибута. Некоторые типы данных имеют очень небольшое внутреннее семантическое содержание. Однако HL7 также определяет более существенные типы данных, например, тип данных имени сущности Entity.name. Каждый атрибут модели RIM ассоциируется с одним и только одним типом данных.

В АКД, выпуск 2, используется абстрактная спецификация типов данных и их представление на языке XML, описанные в документе HL7 V3 Data Types, Release One.

В большинстве случаев разработчикам будет вполне достаточно иметь описания типов данных на языке XML, но иногда им может потребоваться более детальная информация, приведенная в абстрактной спецификации.

5.2.3 Словарные домены HL7

Наиболее полное описание словарных доменов приведено в документе HL7 Vocabulary.

Словарные домены представляют собой наборы допустимых значений кодированных компонентов АКД. Эти домены могут включать в себя как понятия, определенные в стандарте HL7, так и понятия, взятые из систем кодирования, рекомендованных стандартом HL7, например, LOINC или SNOMED. Документ HL7 Vocabulary является наиболее полным справочным источником понятий, определенных в стандарте HL7. Хотя АКД может дополнительно ограничивать словарные домены, определения АКД никогда не будут конфликтовать с определениями документа HL7 Vocabulary.

Словарные домены обладают квалификатором расширяемости, который может принимать значение «Кодировано, без расширений» (CNE), означающий, что для компонента АКД доступны только те значения, которые входят в заданном множестве, или «Кодировано, с расширениями» (CWE), означающий, что при необходимости могут использоваться значения, не входящие в заданное множество. Каждый словарный домен имеет уникальный идентификатор, определенный в стандарте HL7. Аналогично, каждому понятию словарного домена присвоен уникальный код.

Если кодированный компонент АКД ассоциирован с набором значений, имеющим квалификатор расширения CNE, то допустимые значения компонента фиксированы стандартом и перечислены, как показано в таблице 2.

Таблица 2 — Набор значений атрибута relatedDocument.typeCode (CNE)

Код	Определение
APND	Текущий документ является дополнением родительского документа (append)
RPLC	Текущий документ является заменой родительского документа (replace)
XFRM	Текущий документ является преобразованием родительского документа (transform)

5.2.4 Модель АКД R-MIM

Наиболее полное описание процесса уточнения моделей, предложенных в стандарте HL7, можно найти в документе HL7 V3 Guide.

Модель АКД R-MIM описана в 5.4.

При разработке спецификаций стандартов HL7, производных от модели RIM, используется процесс, называемый «клонированием». С его помощью осуществляется уточнение модели RIM в целях построения моделей конкретных предметных областей. Когда в уточненной модели используется специализация класса, определенного в модели RIM, то новый класс, включенный в уточненную модель, называется клоном этого класса. При специализации базовый класс может быть дополнительно ограничен, например, могут быть наложены более сильные ограничения на множественность атрибута или на допустимые словарные значения. В уточненной модели может появиться несколько клонов конкретного класса из модели RIM, представляющих разные специализации.

Модель АКД R-MIM является графическим представлением спецификации АКД. Ее форма соответствует нотации и соглашениям по построению диаграмм, разработанным комитетом HL7 для представления специфичных семантических конструкций содержания критичных, «основных» классов модели RIM. Как ее, так и модель RIM, можно представить в нотации Унифицированного языка моделирования (UML), однако та нотация, что предложена комитетом HL7, предоставляет более детальную информацию о конкретных ограничениях и клонах представляемых классов. Соглашения по представлению диаграмм, принятые комитетом HL7, сокращают некоторые описания отношений, что позволяет сократить диаграммы, сделать их более точными и обеспечить большую визуальную информативность.

Модель АКД R-MIM представляет собой графическое средство, предназначенное для облегчения понимания спецификации. Поскольку иерархическое описание АКД и, соответственно, схема АКД-документов являются производными от модели АКД R-MIM, то эта модель является хорошей основой для описания стандарта. Она дополняется повествовательным описанием специфичных клонов, используемых в АКД.

5.2.5 Иерархическое описание АКД

Наиболее полное описание процесса разработки и интерпретации иерархических описаний в стандартах HL7 можно найти в документе HL7 V3 Guide.

Иерархическое описание АКД приведено в 5.5.

Иерархическое описание является табличным представлением последовательности элементов (а именно, классов, атрибутов и ассоциаций), определенных в модели R-MIM и задающих структуру документа безотносительно к XML или иной технологии реализации.

Иерархическое описание АКД является наиболее полным источником правил соответствия стандарту АКД. Оно используется при конструировании XML-схемы АКД-документов. Экземпляр АКД-документа должен не только соответствовать XML-схеме, но еще и удовлетворять правилам соответствия, объявленным в иерархическом описании АКД. Иерархическое описание второго выпуска АКД однозначно идентифицируется строкой «POCD_HD000040». Как описано в 5.4.1, это значение должно быть включено в экземпляр АКД-документа для однозначного указания его соответствия спецификации АКД, выпуск 2.

5.2.6 Реализация АКД на языке XML

При конструировании XML-схемы АКД-документов использовалась спецификация технологии реализации на языке XML (HL7 XML ITS). Наиболее полное описание этой технологии и процесса перехода от иерархического описания к схеме приведено в документе XML Implementable Technology Specification for V3 Structures.

Схема АКД-документов описана ниже в 5.6.

Стандарт АКД, выпуск 2 основан на спецификации HL7 V3 XML Implementable Technology Specification for V3 Structures, Release One.

Специальные расширения схемы АКД-документов сверх тех, что определены в документе HL7 V3 XML ITS, описаны ниже в 5.6.

При просмотре модели АКД R-MIM читатель, знакомый с моделью RIM, документом HL7 Development Framework и правилами реализации на языке XML, может идентифицировать соответствующие элементы XML и их атрибуты. Но из-за того, что некоторые имена элементов генерируются автоматически, это соответствие может оказаться не очевидным, и в этом случае читателю следует обратиться к документу HL7 V3 XML ITS.

5.3 Передача АКД-документов в сообщениях HL7

Примечание — Точный метод упаковки и передачи экземпляра АКД-документа в сообщении не входит в область применения настоящего стандарта. Описанный в этом подразделе метод кодирования по стандарту MIME не является нормативным, он приведен в качестве иллюстрации одного из способов передачи документов, удовлетворяющих описанным ниже требованиям.

Любая стратегия передачи АКД-документов должна включать в себя следующие требования:

- должна обеспечиваться возможность включения всех компонентов АКД-документа, являющихся неотъемлемыми частями его состояния целостности (например, заверенные мультимедийные данные), в один пакет передачи;
- если ссылки перестают быть доступными после передачи сообщения через межсетевой экран, то должна обеспечиваться возможность включения отображаемого содержания в один пакет передачи;
- должна обеспечиваться возможность включения всех вспомогательных файлов, ассоциированных с АКД-документом и обеспечивающих получателю возможность отображения содержания в соответствии с пожеланиями отправителя (например, одной или нескольких таблиц стилей), в один пакет передачи;
- при создании пакета передачи, включающего базовый АКД-документ, не должна возникать необходимость изменения любых ссылок, содержащихся в этом документе (например, ссылок на заверенное мультимедийное содержание, передаваемое в отдельном файле);
- при извлечении содержания из полученного пакета, включающего базовый АКД-документ, не должна возникать необходимость изменения любых содержащихся в нем ссылок;
- при создании пакета передачи не должна возникать необходимость изменения каких-либо значений XML-атрибутов ID;
- не должно быть никаких ограничений на структуру папок файлов, используемую получателем. Получатели АКД-документа должны иметь возможность размещения его компонентов в папках по своему усмотрению;
- в пакет передачи должны быть включены ключевые метаданные АКД-документа, необходимые для управления документами (например, состояние документа, статус архивирования документа). (Полную информацию о метаданных клинического документа, управлении документами, а также о состояниях документа и переходах состояний см. в спецификации HL7 V3 Medical Records.

В контексте сообщений, определенных в стандартах HL7 2.x и V3, АКД-документ можно рассматривать как мультимедийный объект, который может передаваться в поле с типом данных ED (инкапсулированные данные) закодированным в соответствии со спецификацией многоцелевых расширений интернет-почты MIME (Multipurpose Internet Mail Extensions, RFC 2046).

В текущей спецификации MIME рекомендуется следовать подходу, изложенному в Интернет-стандарте RFC 2557 «MIME Encapsulation of Aggregate Documents, such as HTML (MHTML)» (инкапсулирование агрегированных документов, например, HTML (MHTML), в пакет MIME) и предусматривающему инкапсуляцию в пакетах MIME агрегированных документов, передаваемых в соответствии с протоколами ebXML и DICOM.

В любых сообщениях стандарта HL7 V2.x, позволяющих передавать документы (например, в сообщении MDM), АКД-документы должны передаваться в сегментах OBX. При этом пакет MIME передается в поле OBX-5 (00573 «Результат исследования») как значение типа данных ED. Полю OBX-2 (00570 «Тип значения») должно быть присвоено значение «ED». Значения поля OBX.3 должно быть тем же самым, что и значение компонента ClinicalDocument.code.

Значения многих полей сообщения перекрываются со значениями компонентов АКД-документа. В таблице 3 показано соответствие между полями сегмента TXA, передаваемого в сообщении HL7 V2 MDM, на компоненты АКД-документа.

Таблица 3 — Отображение полей сегмента HL7 V2 TXA на компоненты АКД-документа

Поле сегмента TXA	Компонент АКД-документа
2 Тип документа	ClinicalDocument.code
4 Дата и время действия	ServiceEvent.effectiveTime
5 Код/ФИО основного лица, ответственного за выполнение действия	ServiceEvent performer
6 Дата и время создания документа	ClinicalDocument.effectiveTime
7 Дата и время ввода документа	dataEnterer.time
9 Код/ФИО автора документа	author
11 Код/ФИО лица, которое ввело документ	dataEnterer
12 Уникальный номер документа	ClinicalDocument.id
13 Номер документа-родителя	ParentDocument.id
14 Номер заказа у заказчика	Order.id
18 Статус конфиденциальности документа	ClinicalDocument.confidentialityCode
19 Статус доступности документа	authenticator, legalAuthenticator
22 Штамп исполнителя, отвечающего за аутентичность документа	informationRecipient
23 Разосланные копии (коды и ФИО получателей)	ServiceEvent.effectiveTime

В следующем примере показано не нормативное, но допустимое использование спецификации RFC 2557 в сообщении стандарта HL7 V2. Возможны и другие допустимые представления.

Пример 3

MSH |...
EVN |...
PID |...
PV1 |...
TXA |...
OBX |1|ED|11492-6^History and Physical^LN||
^multipart^related^A^
MIME-Version: 1.0
Content-Type: multipart/related; boundary="HL7-CDA-boundary";
type="text/xml"; start="10.12.45567.43"
Content-Transfer-Encoding: BASE64

--HL7-CDA-boundary
Content-Type: text/xml; charset="US-ASCII"
Content-ID: <10.12.45567.43>

... Представление базового АКД-документа в кодировке Base64

```
...
<observationMedia classCode="OBS" moodCode="EVN">
  <id root="10.23.4567.345"/>
  <value mediaType="image/jpeg">
    <reference value="left_hand_image.jpeg"/>
  </value>
</observationMedia>
```

...

--HL7-CDA-boundary
Content-ID: <10.23.4567.345>
Content-Location: canned_left_hand_image.jpeg
Content-Type: image/JPEG

... Изображение в кодировке Base64 ...

--HL7-CDA-boundary—

АКД-документы могут передаваться в любых сообщениях стандарта HL7 V3, позволяющих передавать документы (например, в сообщении HL7 V3 Medical Records). При этом атрибут Act.text модели RIM должен содержать пакет MIME, закодированный как инкапсулированный тип данных.

Как и в случае сообщений стандарта HL7 V2, значения многих полей сообщения HL7 V3 перекрываются со значениями компонентов АКД-документа. Поскольку спецификация АКД и сообщения HL7 V3 Medical Records произведены от общей информационной модели, соответствие между ними достаточно очевидное (см. таблицу 4).

Таблица 4 — Отображение полей сообщения HL7 V3 Medical Records на компоненты АКД-документа

Компонент сообщения HL7 V3 Medical Records	Компонент АКД-документа	Примечание
ClinicalDocument	ClinicalDocument	Сообщение Medical Records содержит атрибуты, не представленные в АКД-документе (text, statusCode, availabilityTime, reasonCode, completioncode, storageCode, copyTime); АКД-документ содержит атрибуты, не представленные в сообщении Medical Records (title)
authenticator	authenticator	
legalAuthenticator	legalAuthenticator	
dataEnterer	dataEnterer	
EncounterEvent / encounterPerformer	encompassingEncounter / encounterParticipant; serviceEvent / performer	Атрибут encounterPerformer сообщения Medical Records расщепляется на два участника в АКД-документе
responsibleParty	responsibleParty	
custodian	custodian	
participant	participant	
informationRecipient	informationRecipient	
recordTarget	recordTarget	
author	author	
subject	subject	Атрибут subject сообщения Medical Records представляет собой каталог всех субъектов, указанных в документе
relatedDocument / ParentDocument	relatedDocument / parentDocument	
documentationOf / Event	documentationOf / serviceEvent	
inFulfillmentOf / Order	inFulfillmentOf / order	
componentOf / EncounterEvent	componentOf / encompassingEncounter	

В следующем примере показано не нормативное, но допустимое использование спецификации RFC 2557 в сообщении стандарта HL7 V3. Возможны и другие допустимые представления.

Пример 4

```
<someMessage>
 <Act.Code code="11488-4"
  codeSystem="2.16.840.1.113883.6.1"
  displayName="Заключение консультанта"/>
 <Act.text type="multipart/related">
MIME-Version: 1.0
Content-Type: multipart/related; boundary="HL7-CDA-boundary";
type="text/xml"; start="10.12.45567.43"
Content-Transfer-Encoding: BASE64

--HL7-CDA-boundary
Content-Type: text/xml; charset="US-ASCII"
Content-ID: &lt;10.12.45567.43>

... АКД-документ в кодировке Base 64:
 ...
 <observationMedia classCode="OBS" moodcode="EVN">
  <id root="10.23.4567.345"/>
  <value mediaType="image/jpeg">
   <reference value="left_hand_image.jpeg"/>
  </value>
 </observationMedia>
 ...

--HL7-CDA-boundary
Content-ID: &lt;10.23.4567.345>
Content-Location: canned_left_hand_image.jpeg
Content-Type: image/JPEG

... Base64 image ...

--HL7-CDA-boundary--

   </Act.text>
  </someMessage>
```

5.4 Модель АКД R-MIM

Примечание — Наиболее полное описание уточнения модели HL7 V3, разработки и интерпретации модели R-MIM можно найти в документе HL7 V3 Guide.

Графическое представление модели АКД R-MIM (POCD_RM000040) приведено в подразделе 6.2.

АКД-документ состоит из заголовка и тела. Заголовок идентифицирует и классифицирует документ, обеспечивает информацию о его заверении, о посещении пациента, о пациенте и поставщике медицинской помощи. Кроме того, он задает контекст документа в целом. Тело содержит клиническую информацию. Концептуально оно подразделяется на разделы, которые могут быть вложенными. Каждый из них содержит повествовательный блок, предназначенный для визуализации, и может также содержать подразделы и внешние ссылки.

5.4.1 Класс ClinicalDocument

Класс ClinicalDocument является входной точкой в модель АКД R-MIM и соответствует корневому элементу <ClinicalDocument> АКД-документа.

АКД-документ логически разделен на заголовок и тело. Заголовок АКД-документа содержит атрибуты класса ClinicalDocument, информацию об участниках и связях с действиями. Тело АКД-документа является целевым окончанием связи с действием для компонента ClinicalDocument.

Класс ClinicalDocument наследует различные атрибуты класса InfrastructureRoot модели RIM, в том числе ClinicalDocument.templateId и ClinicalDocument.typeId. Значение атрибута ClinicalDocument. templateId, передаваемое в экземпляре АКД-документа, служит признаком наложения ряда ограничений, определяемых в шаблоне документа в целом. Кроме того, этот атрибут присутствует во всех других классах модели АКД. Если он задан, то служит признаком наложения ряда ограничений, определяемых в шаблоне соответствующего уровня детализации. Значение этого атрибута является уникальным идентификатором конкретного шаблона.

Атрибут ClinicalDocument.typeId является технологически нейтральной ссылкой на данную спецификацию второго выпуска АКД. Его компонентам должны быть присвоены следующие значения: ClinicalDocument.typeId.root = «2.16.840.1.113883.1.3» (объектный идентификатор (ОИД) зарегистрированных моделей HL7); ClinicalDocument.typeId.extension = «POCD_HD000040» (уникальный идентификатор иерархического описания АКД, выпуск 2).

5.4.2 Заголовок

Цель заголовка АКД — сделать возможным обмен клиническими документами внутри и между организациями, способствовать управлению клиническими документами, а также встраиванию клинических документов отдельного пациента в его интегрированную электронную медицинскую карту.

5.4.2.1 Атрибуты заголовка

В этом разделе описаны атрибуты корневого класса ClinicalDocument.

Таблица 5 — Допустимые значения атрибута ClinicalDocument.classCode (CNE)

Код	Описание
DOCCLIN (клинический документ) [по умолчанию]	Согласно приведенному выше определению, клинический документ представляет собой совокупность сведений о состоянии здоровья, запланированной и оказанной медицинской помощи

Таблица 6 — Допустимые значения атрибута ClinicalDocument.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие (event), то есть состоявшееся действие документирования того, что уже произошло, а не требование, намерение, планирование или обещание подготовки документа

5.4.2.1.1 ClinicalDocument.id (идентификатор)

Содержит уникальный идентификатор экземпляра клинического документа.

5.4.2.1.2 ClinicalDocument.Code (тип документа)

Содержит код, указывающий конкретный тип документа (например, «Результат осмотра», «Выписной эпикриз», «Дневниковая запись»). Набор допустимых значений взят из классификации LOINC и имеет квалификатор расширяемости CWE.

Начиная с версии базы данных LOINC 2.09 (май 2003 года), коды типов документов описаны в тех записях, которые имеют значение “DOC” в компоненте Scale. Это подмножество LOINC включено в приложение (см. 5.8.2).

Примечание — Построение иерархии кодов документов в классификации LOINC все еще развивается. В руководстве по классификации LOINC, версия 2.14 (декабрь 2004) указано, что в максимально короткие сроки термины компонентов, используемые для создания имен кодов типов документов, будут отображены либо в термины метатезауруса UMLS Metathesaurus, либо в термины номенклатуры SNOMED CT. Это отображение поможет установить значения терминов и позволит объединить и классифицировать коды типов документов на основе определений, вычисляемых отношений и иерархий, уже существующих в справочной терминологии.

5.4.2.1.3 ClinicalDocument.title (название)

Содержит название документа. Клинические документы часто не имеют отдельного названия. В этом случае в качестве общего названия используется отображаемое значение свойства display атрибута ClinicalDocument.code (например, «Заключение консультанта» или «Дневниковая запись»). Если

предполагается отображать название клиницисту, или же если документ имеет уникальное название, то следует использовать компонент ClinicalDocument.title. В примере документа, приведенном в 5.7.1, атрибут ClinicalDocument.title имеет значение «Good Health Clinic Consultation Note».

5.4.2.1.4 ClinicalDocument.effectiveTime (дата и время создания)

Содержит время создания документа, то есть момент появления документа на свет. Если АКД-документ является преобразованием исходного документа в какой-либо иной формате, то ClinicalDocument.effectiveTime должен содержать время создания исходного документа. Время преобразования в настоящий момент не представлено в модели АКД.

5.4.2.1.5 ClinicalDocument.confidentialityCode (код конфиденциальности)

Содержит код конфиденциальности, являющийся обязательным элементом контекста АКД-документа. Значение этого кода, указанное в заголовке, применяется к документу в целом, но может быть переопределено для его части (см. 5.4.4).

Таблица 7 — Допустимые значения атрибута ClinicalDocument.confidentialityCode (CWE)

Код[1)]	Описание
N (обычный) (codeSystem = 2.16.840.1.11388.5.25)	Применяются обычные правила конфиденциальности (в соответствии с устоявшейся практикой в сфере здравоохранения), то есть получить доступ к этому содержанию могут только авторизованные лица, которым этот доступ необходим для выполнения юридически разрешенных медицинских или деловых действий
R (средний) (codeSystem = 2.16.840.1.11388.5.25)	Средние ограничения доступа, например, документ могут читать только медицинские работники, имеющие непосредственное отношение к текущей медицинской помощи пациенту
V (высокий) (codeSystem = 2.16.840.1.11388.5.25)	Высокие ограничения доступа, заданные политикой конфиденциальности

5.4.2.1.6 ClinicalDocument.languageCode (код языка)

Указывает код языка, на котором представлены строковые данные клинического документа (применяется как к содержанию, так и к значениям атрибутов). Значениями атрибута являются идентификаторы языка, определенные в документе IETF (Internet Engineering Task Force) RFC 3066 (по ред. H. Alvestrand. 1995), заменяющем устаревший документ RFC 1766. В стандартах HL7 для этих кодов используется система кодирования 2.16.840.1.113883.6.121. Значение кода языка, указанное в заголовке, применяется к документу в целом, но может быть переопределено для его части (см. 5.4.4).

5.4.2.1.7 ClinicalDocument.setId (идентификатор группы)

Содержит идентификатор группы документов, связанных между собой отношениями замены (версионности), дополнения и преобразования.

5.4.2.1.8 ClinicalDocument.versionNumber (номер версии)

Содержит целочисленный порядковый номер версии в цепочке замены документов.

5.4.2.1.9 ClinicalDocument.copyTime (запрещен)

Содержит время раскрытия документа (например, копирования или отправки на принтер), осуществленного системой управления документами, осуществляющей контроль версий данного документа. Будучи присвоенным, это значение не может быть изменено. Этот атрибут предназначался для использования в качестве информации о том, сколько времени документ пробыл вне безопасного контекста системы управления документами.

Оставлен для обратной совместимости с первым выпуском АКД. Он был запрещен, поскольку не является частью документа при его заверении, а представляет собой метаданные документа, относящиеся к некоторому произвольному моменту времени после того, как он был заверен. Его дальнейшее использование не рекомендовано.

5.4.2.2 Участники, указанные в заголовке АКД-документа

В этом подразделе описаны классы, связанные с корневым классом ClinicalDocument с помощью класса Participation.

[1)] Значение свойства codeSystem указано здесь в связи с тем, что атрибут confidentialityCode имеет тип данных CE и, следовательно, в нем должны быть указаны и код (code), и система кодирования (codeSystem).

5.4.2.2.1 Участник authenticator (контролер)

Содержит сведения о контролере документа, то есть о лице, которое заверяет точность документа, но не несет за него юридическую ответственность. Примером может служить врач-ординатор, который осматривает пациента и диктует запись, а затем подписывает ее. (см. 5.4.2.2.8).

Клинический документ может не иметь контролера, иметь одного или нескольких контролеров. Хотя в АКД-документе не предусмотрена электронная подпись, как контролер, так и лицо, юридически ответственное за содержание документа, должны подписать документ вручную или электронным способом. Для контролера обязательно должны быть указаны время аутентификации (authenticator.time) и код подписи (authenticator.signatureCode), указывающий, что подпись получена и связана с документом.

Таблица 8 — Допустимые значения атрибута authenticator.typeCode (CNE)

Код	Описание
AUTHEN (контролер) [по умолчанию]	Лицо, которое заверяет точность документа, но не несет за него юридическую ответственность

Таблица 9 — Допустимые значения атрибута authenticator.signatureCode (CNE)

Код	Описание
S (подписан)	Документ подписан и сохранен
X (подпись требуется) (запрещен)	В первом выпуске АКД контролер мог быть фактическим (S) или планируемым (X). Во втором выпуске АКД предусмотрен только фактический контролер, поэтому значение X запрещено

Контролером является лицо в роли представителя организации (класс AssignedEntity), то есть лицо, которому контролирующая организация назначила определенную роль. Сущностью, выполняющей роль, является физическое лицо (класс Person). Сущностью, контролирующей роль, является организация (класс Organization), (Описание отношений «роль исполнителя» (player role) и «роль контролера» (scoper role) приведено в модели HL7 RIM).

Таблица 10 — Допустимые значения атрибута AssignedEntity.classCode (CNE)

Код	Описание
ASSIGNED (уполномочен) [по умолчанию]	Роль агента, в которой работник выступает как представитель организации. Акцент делается на функциональной роли представительства

Таблица 11 — Допустимые значения атрибута Person.classCode (CNE)

Код	Описание
PSN (физическое лицо) [по умолчанию]	Живой организм вида homo sapiens

Таблица 12 — Допустимые значения атрибута Person.determinerCode (CNE)

Код	Описание
INSTANCE (экземпляр) [по умолчанию]	Определитель INSTANCE указывает фактический экземпляр сущности, в отличие от определителя KIND, который относится к типу сущностей. Например, можно говорить о конкретном автомобиле (об экземпляре автомобиля) или об автомобилях вообще (о типе транспортного средства)

Таблица 13 — Допустимые значения атрибута Organization.classCode (CNE)

Код	Описание
ORG (организация) [по умолчанию]	Социальная структура или юридическое лицо, созданное людьми

Таблица 14 — Допустимые значения атрибута Organization.determinerCode (CNE)

Код	Описание
INSTANCE (экземпляр) [по умолчанию]	Определитель INSTANCE указывает фактический экземпляр сущности, в отличие от определителя KIND, который относится к типу сущностей. Например, можно говорить о конкретном автомобиле (об экземпляре автомобиля) или об автомобилях вообще (о типе транспортного средства)

Контролирующая организация может быть частью более крупной организации. Если необходимо указать отношение «целое-часть», то можно использовать роль OrganizationPartOf. Атрибут OrganizationPartOf.statusCode содержит код состояния этого отношения (например, «активное», «завершенное»). В атрибуте OrganizationPartOf.effectiveTime указан интервал времени, в течение которого действует отношение «целое-часть», если он применим и известен.

Таблица 15 — Допустимые значения атрибута OrganizationPartOf.classCode (CNE)

Код	Описание
PART (часть) [по умолчанию]	Отношение двух сущностей, при котором исполняющая сущность является частью контролирующей сущности

Таблица 16 — Допустимые значения атрибута OrganizationPartOf.statusCode (CNE)

Код	Описание
normal (нормальная)	Типичное состояние. Исключает состояние nullified, которое указывает, что экземпляр роли был создан по ошибке
active (активна)	Это состояние указывает, что сущность в настоящий момент активна в данной роли
cancelled (отмененное)	Терминальное состояние, возникающее при отмене роли до того, как она стала активной
pending (в ожидании)	Это состояние указывает, что роль еще не стала активной
suspended (приостановлена)	Это состояние указывает, что выполнение роли сущностью, исполняющей эту роль, временно приостановлено. Переход в это состояние возможен из состояния active (активная)
terminated (завершена)	Это состояние указывает, что выполнение роли успешно завершено
nullified (аннулирована)	Это состояние является терминальным состоянием экземпляра роли, созданного по ошибке

5.4.2.2.2 Участник author (автор)

Содержит сведения о физическом лице или устройстве (программном обеспечении), создавшем содержание документа.

В некоторых случаях роль или функция автора вытекает из типа документа, содержащегося в атрибуте ClinicalDocument.code, например, «Дневниковая запись студента-медика». Роль автора может быть также указана в атрибутах Author.functionCode или AssignedAuthor.code. Если какой-либо из них присутствует, то его значение должно быть эквивалентно роли, вытекающей из типа документа, или служить ее дальнейшим уточнением (например, атрибут ClinicalDocument.code имеет значение «Врачебная дневниковая запись», а атрибут Author.functionCode — значение «Лечащий врач»), но отнюдь не конфликтовать с ней, поскольку это может создать двусмысленную ситуацию.

Таблица 17 — Допустимые значения атрибута author.typeCode (CNE)

Код	Описание
AUT (автор) [по умолчанию]	Сторона, которая инициирует выполнение действия и, следовательно, несет ответственность за информацию об этом действии

Таблица 18 — Допустимые значения атрибута author.contextControlCode (CNE)

Код	Описание
OP (замещающая, распространяемая) [по умолчанию]	Ассоциация участника, замещающая ассоциацию с тем же значением атрибута typeCode. Эта замещающая ассоциация будет распространяться на действия-потомки, которые связаны с данным экземпляром класса Act с помощью экземпляра класса ActRelationship (см. ниже CDA Context)

Автором является лицо, исполняющее роль назначенного автора (класс AssignedAuthor). Сущностью, исполняющей эту роль, является физическое лицо (класс Person) или устройство (класс AuthoringDevice). Сущностью, контролирующей эту роль, является та организация (класс Organization), в которой был создан документ.

Таблица 19 — Допустимые значения атрибута AssignedAuthor.classCode (CNE)

Код	Описание
ASSIGNED (уполномочен) [по умолчанию]	Роль агента, в которой исполняющая сущность выступает как служащий или как представитель контролирующей организации

Таблица 20 — Допустимые значения атрибута AuthoringDevice.classCode (CNE)

Код	Описание
DEV (устройство) [по умолчанию]	Сущность, используемая при выполнении действия и существенно не изменяющаяся в процессе его выполнения

Таблица 21 — Допустимые значения атрибута AuthoringDevice.determinerCode (CNE)

Код	Описание
INSTANCE (экземпляр) [по умолчанию]	Определитель INSTANCE указывает фактический экземпляр сущности, в отличие от определителя KIND, который относится к типу сущностей. Например, можно говорить о конкретном автомобиле (об экземпляре автомобиля) или об автомобилях вообще (о типе транспортного средства)

Примечание — В первом выпуске АКД можно было указать лица, ответственные за эксплуатацию устройства. Во втором выпуске такая функциональность запрещена. Класс MaintainedEntity оставлен только для обратной совместимости и его использование допускается только в том случае, если необходимо обеспечить преобразование документа, соответствующему первому выпуску АКД.

Таблица 22 — Допустимые значения атрибута MaintainedEntity.classCode (CNE)

Код	Описание
MNT (эксплуатируемая сущность) [по умолчанию]	Сущность, управляемая другой сущностью. Это типичная роль прочного оборудования. На контролера возложена ответственность за правильное функционирование, качество и безопасность

5.4.2.2.3 Участник custodian (обладатель документа)

Обладателем является организация, которая несет юридическую ответственность за управление документом. Ей вменена обязанность заботы о документе. Каждый АКД-документ имеет одного обладателя.

Участие обладателя удовлетворяет определению ответственности, приведенному в данном стандарте (см. 5.1.1 «Что такое АКД»). Поскольку АКД является стандартом передачи документов, а форма передаваемого документа может отличаться от исходной формы заверенного документа, то под обладателем понимается организация, несущая ответственность за исходный документ.

Таблица 23 — Допустимые значения атрибута custodian.typeCode (CNE)

Код	Описание
CST (обладатель) [по умолчанию]	Организация, которая несет юридическую ответственность за управление данным документом

Обладателем является организация, исполняющая роль назначенного обладателя (класс AssignedCustodian). Ответственной организацией (класс CustodianOrganization) является сущность, контролирующая роль назначенного обладателя. Атрибут CustodianOrganization.id обязателен.

Таблица 24 — Допустимые значения атрибута AssignedCustodian.classCode (CNE)

Код	Описание
ASSIGNED (уполномоченная) [по умолчанию]	Роль, в которой исполняющая сущность выступает как служащий или как представитель контролирующей организации

Таблица 25 — Допустимые значения атрибута CustodianOrganization.classCode (CNE)

Код	Описание
ORG (организация) [по умолчанию]	Социальная структура или юридическое лицо, созданное людьми

Таблица 26 — Допустимые значения атрибута CustodianOrganization.determinerCode (CNE)

Код	Описание
INSTANCE (экземпляр) [по умолчанию]	Определитель INSTANCE указывает фактический экземпляр сущности, в отличие от определителя KIND, который относится к типу сущностей. Например, можно говорить о конкретном автомобиле (об экземпляре автомобиля) или об автомобилях вообще (о типе транспортного средства)

5.4.2.2.4 Участник dataEnterer (оператор по вводу данных)

Содержит информацию об операторе, который преобразовал надиктованное содержание в текст.

Таблица 27 — Допустимые значения атрибута dataEnterer.typeCode (CNE)

Код	Описание
ENT (оператор по вводу данных) [по умолчанию]	Лицо, обеспечившее ввод данных в исходную систему. Регистрация оператора по вводу данных не обязательна и осуществляется для внутренних целей контроля качества информации. Примером служит оператор, который ввел данные в информационную систему с диктофонной записи

Таблица 28 — Допустимые значения атрибута dataEnterer.contextControlCode (CNE)

Код	Описание
OP (замещающая, распространяемая) [по умолчанию]	Ассоциация участника, замещающая ассоциацию с тем же значением атрибута typeCode. Эта замещающая ассоциация будет распространяться на действия-потомки, которые связаны с данным экземпляром класса Act с помощью экземпляра класса ActRelationship (см. ниже CDA Context)

5.4.2.2.5 Участник encounterParticipant (участник посещения)

Описание класса encounterParticipant см. в 5.4.2.3.5.

5.4.2.2.6 Участник informant (информатор)

Информатором (или источником информации) является лицо, которое предоставило соответствующую информацию, например, родитель коматозного пациента, сообщивший о поведении пациента до наступления комы.

Таблица 29 — Допустимые значения атрибута informant.typeCode (CNE)

Код	Описание
INF (информатор) [по умолчанию]	Источник информации, включенной в содержание документа (например, близкое лицо, отвечающее на вопросы об истории заболевания пациента). Если не указано иное, то при сборе сведений об истории заболевания информатором неявно является пациент

Таблица 30 — Допустимые значения атрибута informant.contextControlCode (CNE)

Код	Описание
OP (замещающая, распространяемая) [по умолчанию]	Ассоциация участника, замещающая ассоциацию с тем же значением атрибута typeCode. Эта замещающая ассоциация будет распространяться на действия-потомки, которые связаны с данным экземпляром класса Act с помощью экземпляра класса ActRelationship (см. ниже CDA Context)

Источником информации может быть лицо в одной из двух ролей. Роль RelatedEntity используется для того, чтобы представить информатора, которому не присвоен идентификатор role.id (например, родителя или человека с улицы). В этом случае информатор имеет некоторое формальное или персональное отношение к пациенту. Для этой роли нет контролирующей сущности, поскольку подразумевается, что пациент всегда является неявным контролером. Для описания характера отношения информатора к пациенту может быть использован атрибут RelatedEntity.code. Роль AssignedEntity используется для идентифицированного информатора, и должна контролироваться какой-либо организацией.

Таблица 31 — Допустимые значения атрибута RelatedEntity.classCode (CNE)

Код	Описание
Любой подтип домена RoleClassMutualRelationship	Роль сущности, основанная на некотором взаимоотношении с пациентом. Основой такого отношения могут быть соглашения (например, брачный союз, двусторонний договор), устоявшиеся отношения (например, с друзьями), а также побочные отношения (например, участие сторон в диспуте, братья или сестры, дети). Допустимые значения составляют домен RoleClassMutualRelationship

5.4.2.2.7 Участник informationRecipient (получатель информации)

Содержит информацию о получателе документа.

Примечание — Получателем информации считается лицо, которое автор документа назвал получателем копии документа в момент его создания. Это не то же самое, что перечень лиц, которым документ впоследствии был предоставлен на протяжении жизни пациента. Подобный список фактов предоставления документа не хранится в документе и находится вне области применения АКД.

Таблица 32 — Допустимые значения атрибута informationRecipient.typeCode (CNE)

Код	Описание
PRCP (основной получатель) [по умолчанию]	Основной получатель документа
TRC (вторичный получатель)	Вторичный получатель документа

Если получателем является лицо, исполняющее роль назначенного получателя (класс IntendedRecipient), то сущностью, исполняющей эту роль, является физическое лицо (класс Person), которое может контролироваться организацией (класс Organization). Если получателем является организация, то атрибуту IntendedRecipient.classCode присваивается значение «ASSIGNED», и признаком такого получателя является указание контролирующей организации без указания исполнителя роли. Если в качестве получателя должна быть указана медицинская карта, то атрибуту IntendedRecipient. classCode присваивается значение «HLTHCHRT». В этом случае исполнитель роли не указывается, а указание контролирующей организации не является обязательным.

Таблица 33 — Допустимые значения атрибута informationRecipient.typeCode (CNE)

Код	Описание
ASSIGNED (уполномоченная) [по умолчанию]	Роль, в которой исполняющая сущность выступает как служащий или как представитель контролирующей организации
HLTHCHRT (медицинская карта)	Роль, в которой исполняющей сущностью является медицинская карта, принадлежащая контролирующей организации

5.4.2.2.8 Участник legalAuthenticator (юридически ответственное лицо)

Содержит информацию об участнике, на которого возлагается юридическая ответственность за документ.

Стандарт АКД определяет структуру передаваемых клинических документов. Если в целях передачи исходный документ был преобразован в АКД-документ, то заверяется исходный документ и этот факт отражается в передаваемом АКД-документе. АКД-документ может быть не заверенным, заверенным или юридически заверенным. Не заверенное состояние имеет тот документ, в котором отсутствует информация о заверении (то есть отсутствует информация о контролере и юридически ответственном лице).

Хотя электронные подписи не передаются в АКД-документе, для заверения и юридического заверения требуется, чтобы документ было подписан ответственным лицом вручную или электронным способом. У атрибута legalAuthenticator есть обязательное свойство legalAuthenticator.time, означающее дату и время заверения, а также обязательное свойство legalAuthenticator.signatureCode, которое служит признаком наличия подписи и сохранения подписанного документа.

Таблица 34 — Допустимые значения атрибута legalAuthenticator.typeCode (CNE)

Код	Описание
LA (юридически ответственное лицо) [по умолчанию]	Контролер, который юридически заверяет точность действия. Примером может служить врач, осматривающий пациента, диктующий результат осмотра, а затем подписывающий введенный текст. Тем самым он принимает на себя юридическую ответственность за содержание документа

Таблица 35 — Допустимые значения атрибута legalAuthenticator.signatureCode (CNE)

Код	Описание
S (подписан)	Документ подписан и сохранен
X (подпись требуется) (запрещен)	В первом выпуске АКД контролер мог быть фактическим («S») или планируемым («X»). Во втором выпуске АКД предусмотрен только фактический контролер, поэтому значение «X» запрещено

Таблица 36 — Допустимые значения атрибута legalAuthenticator.contextControlCode (CNE)

Код	Описание
OP (замещающая, распространяемая) [по умолчанию]	Ассоциация участника, замещающая ассоциацию с тем же значением атрибута typeCode. Эта замещающая ассоциация будет распространяться на действия-потомки, которые связаны с данным экземпляром класса Act с помощью экземпляра класса ActRelationship (см. 5.4.4)

Юридически ответственным является лицо в роли представителя организации (класс AssignedEntity), то есть лицо, которому контролирующая организация назначила определенную роль. Сущностью, выполняющей роль, является физическое лицо (класс Person). Сущностью, контролирующей роль, является организация (класс Organization).

5.4.2.2.9 Участник participant (другой участник)

Содержит информацию об ином участнике, не вошедшем ни в один из других классов, но каким-либо образом вовлеченном в документируемые действия.

Таблица 37 — Допустимые значения атрибута participant.typeCode (CNE)

Код	Описание
Любой подтип домена ParticipationType	Допустимые значения составляют домен ParticipationType

Таблица 38 — Допустимые значения атрибута participant.contextControlCode (CNE)

Код	Описание
OP (замещающая, распространяемая) [по умолчанию]	Ассоциация участника, замещающая ассоциацию с тем же значением атрибута typeCode. Эта замещающая ассоциация будет распространяться на действия-потомки, которые связаны с данным экземпляром класса Act с помощью экземпляра класса ActRelationship (см. 5.4.4)

Участником является физическое лицо или организация в роли участвующей сущности (класс AssociatedEntity). Сущностью, выполняющей роль, является физическое лицо (класс Person). Сущностью, контролирующей роль, является организация (класс Organization).

Таблица 39 — Допустимые значения атрибута AssociatedEntity.classCode (CNE)

Код	Описание
Любой подтип домена RoleClassAssociative	Допустимые значения составляют домен RoleClassAssociative

Если участником является организация, то признаком этого является указание контролирующей организации без указания исполнителя роли.

5.4.2.2.10 Участник performer (исполнитель)

Описание участника-исполнителя см. в 5.4.2.3.2.

5.4.2.2.11 Участник recordTarget (целевая медицинская карта)

Содержит идентификацию медицинской карты, в которую вкладывается данный документ.

Обычно клинический документ связан только с одной картой, идентифицированной атрибутом recordTarget. В тех редких случаях, когда клинический документ (например, дневник группового посещения) помещается более чем в одну медицинскую карту, можно указать несколько экземпляров атрибута recordTarget.

Значение атрибута recordTarget задается в заголовке документа, распространяется на вложенное содержание и замещению не подлежит (см. 5.4.4).

Таблица 40 — Допустимые значения атрибута recordTarget.typeCode (CNE)

Код	Описание
RCT (целевая медицинская карта) [по умолчанию]	Целевая медицинская карта, в которую помещается информация о данном действии

Таблица 41 — Допустимые значения атрибута recordTarget.contextControlCode (CNE)

Код	Описание
OP (замещающая, распространяемая) [по умолчанию]	Ассоциация участника, замещающая ассоциацию с тем же значением атрибута typeCode. Эта замещающая ассоциация будет распространяться на действия-потомки, которые связаны с данным экземпляром класса Act с помощью экземпляра класса ActRelationships (см. 5.4.4)

Целевая медицинская карта (recordTarget) представлена как отношение между физическим лицом и организацией, в котором лицо выполняет роль пациента. Сущностью, выполняющей роль, является пациент (класс Patient). Сущностью, контролирующей роль, является организация (класс Organization). Уникальным идентификатором пациента является значение атрибута PatientRole.id.

В первом выпуске АКД можно было использовать различные идентификаторы лица, соответствующие значению атрибута Patient.id в АКД, выпуск 2. Этот атрибут включен для обратной совместимости и запрещен, поскольку два разных способа идентификации пациента могут привести к несовместимому применению. Дальнейшее использование атрибута Patient.id не рекомендовано.

Таблица 42 — Допустимые значения атрибута PatientRole.classCode (CNE)

Код	Описание
PAT (пациент) [по умолчанию]	Лицо, получающее медицинскую помощь у медицинской организации

Таблица 43 — Допустимые значения атрибута Patient.classCode (CNE)

Код	Описание
PSN (физическое лицо) [по умолчанию]	Живой организм вида homo sapiens

Таблица 44 — Допустимые значения атрибута Patient.determinerCode (CNE)

Код	Описание
INSTANCE (экземпляр) [по умолчанию]	Определитель INSTANCE указывает фактический экземпляр сущности, в отличие от определителя KIND, который относится к типу сущностей. Например, можно говорить о конкретном автомобиле (об экземпляре автомобиля) или об автомобилях вообще (о типе транспортного средства)

Способности пациента к общению на определенном языке могут быть представлены ассоциированным классом LanguageCommunication. Место рождения пациента представляется как отношение пациента к определенному месту. Класс Birthplace (место рождения) является ролью места (класс Place), контролируемой пациентом (класс Patient).

Таблица 45 — Допустимые значения атрибута Birthplace.classCode (CNE)

Код	Описание
BIRTHPL (место рождения) [по умолчанию]	Связывает экземпляр класса Place, в котором указано место рождения, с экземпляром класса LivingSubject, описывающим живой организм

Таблица 46 — Допустимые значения атрибута Place.classCode (CNE)

Код	Описание
PLC (место) [по умолчанию]	Физическое место или участок с содержащимися в них структурами

Таблица 47 — Допустимые значения атрибута Place.determinerCode (CNE)

Код	Описание
INSTANCE (экземпляр) [по умолчанию]	Определитель INSTANCE указывает фактический экземпляр сущности, в отличие от определителя KIND, который относится к типу сущностей. Например, можно говорить о конкретном автомобиле (об экземпляре автомобиля) или об автомобилях вообще (о типе транспортного средства)

Опекун пациента — это физическое лицо или организация в роли опекуна (класс Guardian). Сущностью, выполняющей роль опекуна, является лицо (класс Person) или организация (класс Organization). Контролирующей сущностью является пациент (класс Patient).

Если опекун явно не указан, то его значение по умолчанию должно быть определено в соответствии с местной практикой (например, пациент самостоятельно соглашается на предлагаемую медицинскую помощь, кроме случаев его физической неспособности, и тогда решение принимает его супруг(а)).

Таблица 48 — Допустимые значения атрибута Guardian.classCode (CNE)

Код	Описание
GUARD (опекун) [по умолчанию]	Сущность (исполнитель), действующая или уполномоченная действовать в качестве опекуна пациента

5.4.2.2.12 Участник responsibleParty (ответственная сторона)

Описание участника responsibleParty (ответственная сторона) см. в 5.4.2.3.5.

5.4.2.2.13 Сценарии участия

Одно и то же лицо может одновременно выступать в ролях нескольких участников, указанных в заголовке АКД-документа. В этих случаях оно должно быть идентифицировано как исполнитель соответствующей роли. Например, если лицо является и автором, и контролером документа, то в заголовке АКД-документа оно должно быть указано и как автор, и как контролер.

В других случаях участниками, указанными в заголовке АКД-документа, являются разные лица. В таблице 49 приведены примеры разных сценариев с разными участниками.

Таблица 49 — Сценарии выполнения ролей, идентифицированных в заголовке АКД-документа

1 Лечащий врач осматривает пациента как консультант, диктует результат осмотра, затем подписывает напечатанный текст	
Автор	Лечащий врач
Участник посещения	Лечащий врач (typeCode = «CONS»)
Юридически ответственное лицо	Лечащий врач
2 Лечащий врач осматривает пациента, диктует результат осмотра. Заведующий отделением подписывает напечатанный текст	
Автор	Лечащий врач
Юридически ответственное лицо	Заведующий отделением
3 Ординатор осматривает пациента вместе с лечащим врачом. Ординатор диктует результат осмотра и подписывает напечатанный текст. Лечащий врач ставит на документе вторую подпись	
Автор	Ординатор
Контролер	Контролер
Участник посещения	Лечащий врач (typeCode = «ATND»)
Юридически ответственное лицо	Лечащий врач
4 Ординатор осматривает пациента вместе с лечащим врачом. Лечащий врач диктует результат осмотра и подписывает напечатанный текст. Заведующий отделением ставит вторую подпись	
Автор	Ординатор
Контролер	Ординатор
Участник посещения	Лечащий врач (typeCode = «ATND»)
Юридически ответственное лицо	Заведующий отделением
5. Ординатор осматривает пациента вместе с лечащим врачом. Ординатор диктует результат осмотра и уходит в отпуск. Заменяющий его ординатор подписывает напечатанный документ. Лечащий врач ставит вторую подпись	
Автор	Ординатор
Контролер	Заменяющий ординатор
Участник посещения	Лечащий врач (typeCode = «ATND»)
Юридически ответственное лицо	Лечащий врач

Окончание таблицы 49

6 Ординатор осматривает пациента вместе с лечащим врачом. Ординатор диктует результат осмотра, который подписывают другой ординатор и заведующий отделением	
Автор	Ординатор
Контролер	Другой ординатор
Участник посещения	Лечащий врач (typeCode = «ATND»)
Юридически ответственное лицо	Заведующий отделением
Лечащий врач получает из лаборатории патологический результат анализа, безуспешно пытается связаться с пациентом, делает дневниковую запись и подписывает ее	
Автор	Лечащий врач
Юридически ответственное лицо	Лечащий врач
Ассистент оперирует пациента вместе с хирургом. Хирург диктует дневник операции и подписывает напечатанный текст	
Автор	Хирург
Контролер	Отсутствует (не должен быть указан)
Юридически ответственное лицо	Хирург
Исполнитель	Хирург (typeCode = «PPRF»)
Исполнитель	Ассистент (typeCode = «SPRF»)

Примечание — Возможность постановки второй подписи или подписи по поручению другого клинициста определяется действующими нормами и местной практикой.

5.4.2.3 Отношения, указанные в заголовке

В этом подразделе описаны классы, связанные с корневым классом ClinicalDocument с помощью ассоциации ActRelationship.

5.4.2.3.1 ParentDocument (родительский документ)

Атрибут ParentDocument используется для указания документа-источника, к которому были применены действия пересмотра, дополнения или преобразования. Свойство ParentDocument.text имеет тип данных ED. С его помощью можно указать тип MIME родительского документа, но содержание этого документа не должно в него вкладываться, поэтому использование свойства ParentDocument.text.BIN запрещено.

Допустимые значения кода отношения текущего документа к родительскому (relatedDocument. typeCode) приведены в таблице 50.

Таблица 50 — Допустимые значения атрибута relatedDocument.typeCode (CNE)

Код	Описание
APND	Текущий документ является дополнением родительского документа (append)
RPLC	Текущий документ является заменой родительского документа (replace)
XFRM	Текущий документ является преобразованием родительского документа (transform)

К родительскому документу, совместимому со спецификацией АКД, может идти от других документов единственная связь типа «APND», единственная связь типа «RPLC», единственная связь типа «XFRM», пара связей типа «XFRM» и «RPLC» либо пара связей «XFRM» и «APND». Никакие другие сочетания связей не допускаются.

Таблица 51 — Допустимые значения атрибута ParentDocument.classCode (CNE)

Код	Описание
DOCCLIN (клинический документ) [по умолчанию]	Клинический документ

Таблица 52 — Допустимые значения атрибута ParentDocument.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие

Идентификация документа, его пересмотры и дополнения

Клинический документ может быть заменен новым документом или дополнен им.

Заменяющий документ представляет собой новую версию родительского документа. Родительский документ считается устаревшим, но система может сохранить его для истории или для отчетности. На замененный родительский документ делается ссылка relatedDocument, у которой свойство relatedDocument.typeCode имеет значение «RPLC» (замена). Примером может служить протокол исследования, содержащий ошибку и затем замененный на исправленный протокол.

Дополнение представляет собой отдельный документ, который ссылается на родительский документ и может дополнять или модифицировать содержание предшествующего документа. Родительский документ остается текущим компонентом медицинской карты пациента. Получатели документов должны читать как дополнение, так и его родительский документ. На дополненный родительский документ делается ссылка relatedDocument, у которой свойство relatedDocument.typeCode имеет значение «APND» (дополнение).

Каждый АКД-документ должен иметь уникальный идентификатор, передаваемый в атрибуте ClinicalDocument.id. Поэтому и у заменяющего, так и у дополняющего документа должен быть свой идентификатор ClinicalDocument.id, отличающийся от идентификатора родительского документа.

АКД-документы могут содержать также атрибуты идентификатора группы документов (ClinicalDocument.setId) и номера версии (ClinicalDocument.versionNumber), которые в совокупности обеспечивают поддержку схемы идентификации документа и его версионности, используемой в некоторых системах управления документами. В соответствии с этой схемой все документы в цепочке замен имеют одинаковый идентификатор группы документов ClinicalDocument.setId и отличаются последовательно возрастающими номерами версий ClinicalDocument.versionNumber. В исходном документе кроме уникального идентификатора самого документа, ClinicalDocument.id, должен быть указан новый уникальный идентификатор группы документов ClinicalDocument.setId, а номер версии ClinicalDocument.versionNumber должен иметь значение 1. (Необходимо иметь в виду, что при замене документа номер версии должен увеличиться на 1, но по местным требованиям приращение номера может быть иным.)

Примеры этих отношений между документами показаны на рисунке 1. Типичным сценарием являются простая замена (например, документ с идентификатором ClinicalDocument.id «1.2.345.6789.266» заменяет документ с идентификатором ClinicalDocument.id «1.2.345.6789.123») и простое дополнение (например, документ с идентификатором ClinicalDocument.id «1.2.345.6789.456» дополняет документ с идентификатором ClinicalDocument.id «1.2.345.6789.123»). Можно предвидеть и более сложные сценарии, например, замену дополнения (документ с идентификатором ClinicalDocument.id «1.2.345.6789.224» заменяет документ с идентификатором ClinicalDocument.id «1.2.345.6789.456», который в свою очередь является дополнением документа с идентификатором ClinicalDocument.id «1.2.345.6789.123»). В этом случае для чтения заменяющий документ с идентификатором ClinicalDocument.id «1.2.345.6789.224» должен быть показан как дополнение документа с идентификатором ClinicalDocument.id «1.2.345.6789.123». Другим примером служит дополнение замененного документа (документ с идентификатором ClinicalDocument.id «1.2.345.6789.456» дополняет документ с идентификатором ClinicalDocument.id «1.2.345.6789.123», который заменен на документ с идентификатором ClinicalDocument.id «1.2.345.6789.266». В этом случае для чтения дополняющий документ должен быть показан вместе с заменяющим документом (например, документ с идентификатором ClinicalDocument.id «1.2.345.6789.456» должен быть показан как дополнение документа с идентификатором ClinicalDocument.id «1.2.345.6789.266».

Преобразования документа

АКД-документ может быть получен в результате преобразования другого документа, то есть был получен в результате машинного отображения из какого-либо иного формата (например, из формата структурированного протокола лучевого исследования DICOM SR). В этом случае атрибут relatedDocument.typeCode должен иметь значение «XFRM».

При правильном преобразовании человеко-читаемое содержание документа не должно претерпеть никаких изменений. По местным правилам преобразованный протокол может заменить исходный, но обычно это не практикуется. Если это все же произошло, то должна использоваться дополнительная связь с типом «RPLC». Отношение «XFRM» может использоваться и в том случае, если документ в местном формате преобразуется в АКД-документ для передачи последнего. В этом случае целевым значением связи «XFRM» будет идентификатор местного документа.

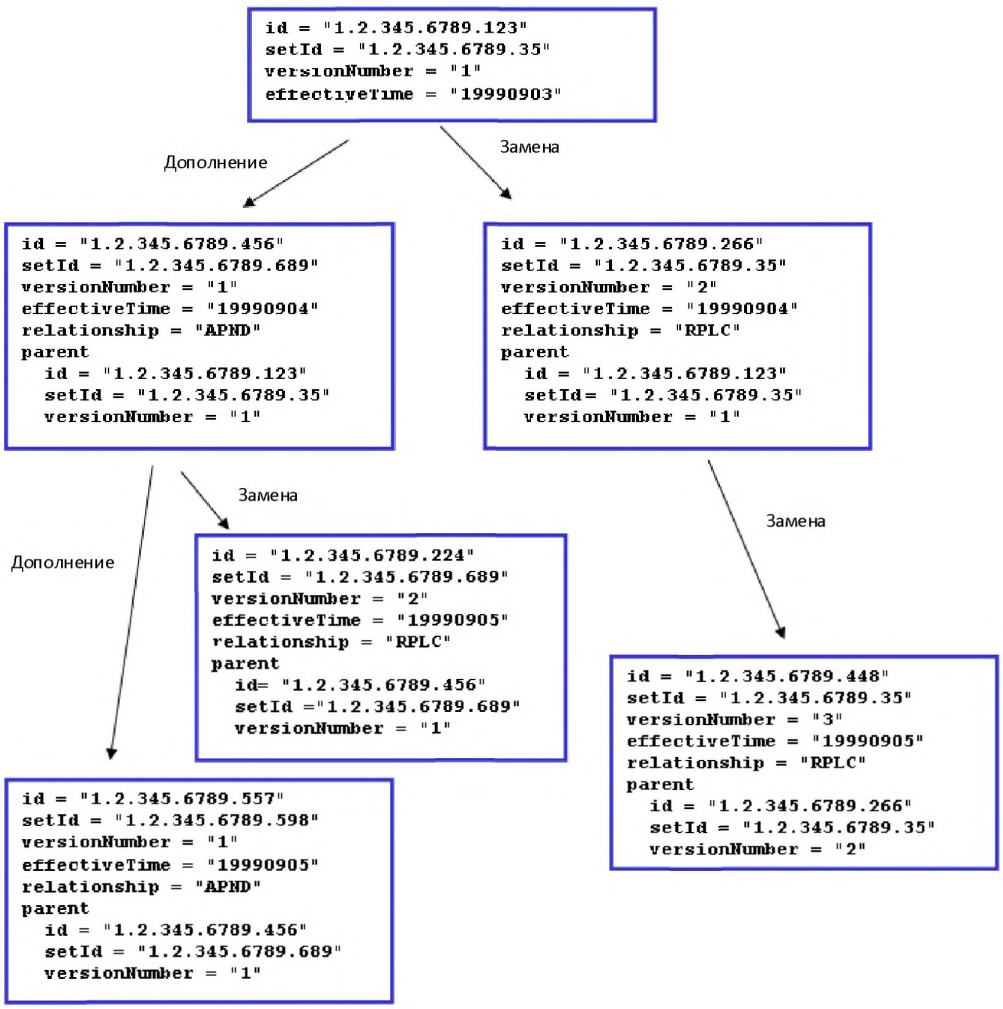


Рисунок 1 — Отношения дополнения и замены

5.4.2.3.2 ServiceEvent (событие медицинской помощи)

Этот класс представляет основное документируемое действие, например, колоноскопию или аппендэктомию.

В некоторых случаях содержание класса ServiceEvent производится от типа клинического документа, указанного в атрибуте ClinicalDocument.code, например, этот атрибут имеет значение «Сбор анамнеза и физикальный осмотр» и документируемой процедурой является сбор анамнеза и физикальный осмотр. Содержание класса ServiceEvent может представлять собой дальнейшую специали-

зацию действия, подразумеваемого типом документа, например, атрибут ClinicalDocument.code имеет значение «Протокол процедуры», а документируемой процедурой является колоноскопия. Если класс ServiceEvent присутствует, то его значение должно быть эквивалентно типу документа, или служить его дальнейшим уточнением, но отнюдь не конфликтовать с ним, поскольку это может создать двусмысленную ситуацию.

Свойство ServiceEvent.effectiveTime может использоваться для указания даты и времени документируемого события, если его надо отличать от даты и времени более протяженного посещения, в ходе которого это событие произошло.

Таблица 53 — Допустимые значения атрибута documentationOf.typeCode (CNE)

Код	Описание
DOC (документ) [по умолчанию]	Текущий документ является документированием события, идентифицированного в классе ServiceEvent

Таблица 54 — Допустимые значения атрибута ServiceEvent.classCode (CNE)

Код	Описание
ACT (действие) [по умолчанию]	Медицинская помощь
Любой подтип типа ACT	Допустимые значения берутся из словарного домена ActClassRoot

Таблица 55 — Допустимые значения атрибута ServiceEvent.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие (event), то есть состоявшееся действие документирования того, что уже произошло, а не требование, намерение, планирование или обещание подготовки документа

Участником performer (исполнитель) является медицинский работник, который фактически оказал медицинскую помощь, указанную в классе ServiceEvent. Свойство performer.time может использоваться для указания того времени, в течение которого исполнитель принимал непосредственное участие в документируемом действии. Свойство performer.functionCode может использоваться для указания дополнительных деталей о функции исполнителя (например, медсестра, которая берет мазок, третий ассистент и т. д.). Значения этого свойства берутся из словарного домена ParticipationFunction, имеющего квалификатор расширяемости CWE.

Таблица 56 — Допустимые значения атрибута performer.typeCode (CNE)

Код	Описание
PRF (исполнитель)	Лицо, фактически оказавшее медицинскую помощь
PPRF (основной исполнитель)	Основной исполнитель медицинской помощи, указанной в классе ServiceEvent
SPRF (другой исполнитель)	Лицо, непосредственно присутствующее при оказании медицинской помощи, указанной в классе ServiceEvent, и существенно участвующее в ней, например, ассистент, инженер, помощник или иной исполнитель

Исполнителем является сущность в роли ассоциированной сущности (класс AssignedEntity). Ассоциированной сущностью является лицо, которому эта роль назначена контролирующей сущностью. Сущностью, исполняющей роль, является физическое лицо (класс Person), а контролирующей сущностью — организация (класс Organization).

5.4.2.3.3 Order (заказ)

В экземплярах этого класса указаны те заказы, результат исполнения которых указан в данном документе. Например, врач заказывает рентгенологическое исследование. Исследование выполнено.

Рентгенолог анализирует рентгеновский снимок и составляет протокол исследования. Идентификатор заказа исследования передается в классе Order, код выполненного рентгенологическое исследования передается в классе ServiceEvent, а код документа ClinicalDocument.code должен иметь значение «Протокол рентгенологического исследования».

Таблица 57 — Допустимые значения атрибута inFulfillmentOf.typeCode (CNE)

Код	Описание
FLS (выполнение) [по умолчанию]	Текущий документ документирует результат выполнения заказа, указанного в классе ActOrder

Таблица 58 — Допустимые значения атрибута Order.classCode (CNE)

Код	Описание
ACT (действие) [по умолчанию]	Медицинская помощь
Любой подтип типа ACT	Допустимые значения берутся из словарного домена ActClassRoot

Таблица 59 — Допустимые значения атрибута Order.moodCode (CNE)

Код	Описание
RQO (заказ) [по умолчанию]	Требование или заказ документированного действия

5.4.2.3.4 Consent (согласие)

В экземплярах этого класса передается информация о согласиях, ассоциируемых с данным документом. Тип согласия (например, согласие на выполнение медицинской помощи, указанной в связанном классе ServiceEvent, согласие на передачу третьей стороне информации, содержащейся в данном документе) передается в атрибуте Consent.code. Согласия, информация о которых передается в заголовке документа, должны быть свершившимся фактом (атрибут Consent.statusCode должен иметь значение «completed») и должны быть сохранены.

Таблица 60 — Допустимые значения атрибута authorization.typeCode (CNE)

Код	Описание
AUTH (разрешает) [по умолчанию]	Согласие разрешает или сертифицирует действия, указанные в текущем документе

Таблица 61 — Допустимые значения атрибута Consent.classCode (CNE)

Код	Описание
CONS (согласие) [по умолчанию]	Класс Consent представляет информированные согласия и законные медицинские соглашения

Таблица 62 — Допустимые значения атрибута Consent.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие

Таблица 63 — Допустимые значения атрибута Consent.statusCode (CNE)

Код	Описание
completed	Согласие, информация о котором передается в АКД-документе, является свершившимся фактом и сохранено

5.4.2.3.5 EncompassingEncounter (посещение)

В этом необязательном классе передается информация о месте оказания медицинской помощи при посещении пациента, в процессе которого произошли документируемые действия или была оказана медицинская помощь, указанная в классе ServiceEvent. Документы не обязательно составляются во время посещения. Например, клиницист получает патологический результат лабораторного теста, безуспешно пытается связаться с пациентом и пишет дневниковую запись.

В некоторых случаях идентификация места оказания медицинской помощи вытекает из типа документа, передаваемого в атрибуте ClinicalDocument.code, например «Дневниковая запись Клиники диабета». Идентификация места оказания медицинской помощи может быть также передана в атрибуте HealthCareFacility.code. Если этот атрибут присутствует, то его значение должно быть эквивалентно значению, вытекающему из типа документа, или служить его дальнейшим уточнением (например, атрибут ClinicalDocument.code имеет значение «Дневниковая запись», а атрибут HealthCareFacility.code — значение «Клиника кардиологии»), но отнюдь не конфликтовать с ним, поскольку это может создать двусмысленную ситуацию.

Атрибут EncompassingEncounter.dischargeDispositionCode может использоваться для указания типа выписки пациента из больницы (например, выписан домой, скончался, отказался от медицинской помощи и т. д.).

Таблица 64 — Допустимые значения атрибута componentOf.typeCode (CNE)

Код	Описание
COMP (компонент) [по умолчанию]	В текущем документе документируются события, произошедшие при посещении, идентифицируемом классом EncompassingEncounter

Таблица 65 — Допустимые значения атрибута EncompassingEncounter.classCode (CNE)

Код	Описание
ENC (посещение) [по умолчанию]	Контакт пациента с одним или несколькими медицинскими работниками, в процессе которого пациенту оказана медицинская помощь или проведена оценка состояния его здоровья

Таблица 66 — Допустимые значения атрибута EncompassingEncounter.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие

Участник location (местонахождение) связывает медицинскую организацию (класс HealthCareFacility) и посещение для указания, где оно имело место. Сущностью, исполняющей роль медицинской организации, является местонахождение (класс Place). Сущностью, контролирующей эту роль, является организация (класс Organization).

Место посещения (например, клиника кардиологии, поликлиника, реабилитационный центр, дом престарелых) может быть передано в атрибуте HealthCareFacility.code. Место посещения и физическое местонахождение — не одно и то же. Между местом посещения и физическим местонахождением существует отношение многие-ко-многим. Например, конкретный кабинет может оказаться местом посещения клиники кардиологии в один день и местом посещения поликлиники — в другой. А клиника кардиологии сегодня может иметь одно физическое местонахождение, а завтра — другое.

Если местом посещения является организация в целом, то контролирующий класс Organization должен присутствовать без указания местонахождения (класс Place).

Таблица 67 — Допустимые значения атрибута location.typeCode (CNE)

Код	Описание
LOC (местонахождение) [по умолчанию]	Физическое местонахождение, где была оказана медицинская помощь. Может быть статическим (здание или кабинет в нем) или мобильным (например, машина скорой помощи, вертолет, самолет, поезд, грузовик, корабль и т. д.)

Таблица 68 — Допустимые значения атрибута HealthCareFacility.classCode (CNE)

Код	Описание
SDLOC (место оказания медицинской помощи) [по умолчанию]	Роль места, в котором может быть оказана медицинская помощь
Любой подтип домена RoleClassServiceDeliveryLocation	Допустимые значения берутся из словарного домена RoleClassServiceDeliveryLocation

Участником responsibleParty (ответственная сторона) является участник, на которого возложена основная юридическая ответственность за посещение. Он отличается от юридически ответственного лица (legalAuthenticator) тем, что последнее может и не быть ответственной стороной, поскольку его функция состоит в придании документу юридической силы и переводу его в состояние завершенности посредством подписи.

Таблица 69 — Допустимые значения атрибута responsibleParty.typeCode (CNE)

Код	Описание
RESP Ответственная сторона) [по умолчанию]	Поставщик медицинской помощи (физическое лицо или организация), на которого возложена основная ответственность за посещение. Ответственный поставщик не обязательно непосредственно участвует в посещении, но несет ответственность за делегированное действие и за контроль действий непосредственного исполнителя

Ответственной стороной (класс responsibleParty) является лицо или организация в роли ассоциированной сущности (класс AssignedEntity). Ассоциированной сущностью является лицо, которому эта роль назначена контролирующей сущностью. Сущностью, исполняющей роль, является физическое лицо (класс Person), а контролирующей сущностью — организация (класс Organization).

Если ответственной стороной является организация, то признаком этого является значение атрибута AssignedEntity.classCode, равное «ASSIGNED», и указание контролирующей организации без указания исполнителя роли.

Экземпляры участников encounterParticipant представляют медицинских работников, непосредственно участвующих в посещении (например, начиная его, завершая или надзирая за ним).

Таблица 70 — Допустимые значения атрибута encounterParticipant.typeCode (CNE)

Код	Описание
АДМ (принимающий врач)	Врач, принимающий пациента при поступлении в стационар
ATND (лечащий врач)	Основной врач, координирующий лечение пациента во время посещения
CONS (консультант)	Врач, который в процессе посещения оценивает состояние пациента и дает рекомендации лечащему врачу
DIS (выписывающий врач)	Врач, отвечающий за выписку пациента из стационара
REF (направляющий врач)	Лицо, которое по итогам посещения направляет пациента к другому врачу для оказания медицинской помощи

Участником encounterParticipant (участник посещения) является сущность в роли уполномоченной сущности (класс AssignedEntity). Уполномоченной сущностью является лицо, которому контролирующая организация назначила эту роль. Сущностью, исполняющей роль, является физическое лицо (класс Person). Сущностью, контролирующей роль, является организация (класс Organization).

5.4.3 Тело документа

5.4.3.1 Варианты структуры тела документа

Тело АКД-документа может представлять собой неструктурированный блок двоичных данных или содержать структурирующую разметку. Каждый АКД-документ имеет ровно одно тело, ассоциируемое с классом ClinicalDocument с помощью связи component.

Таблица 71 — Допустимые значения атрибута responsibleParty.typeCode (CNE)

Код	Описание
COMP (компонент) [по умолчанию]	Ассоциированное тело документа является компонентом документа

5.4.3.1.1 NonXMLBody (тело, не являющееся XML-фрагментом)

Класс NonXMLBody представляет тело документа, имеющее формат, отличный от XML. Он используется для ссылки на данные, хранящиеся вне АКД-документа, или для кодирования данных, вложенных в документ.

Для визуализации тела, не являющегося XML-фрагментом, необходимо программное обеспечение, способное обработать конкретный тип среды MIME, присвоенный блоку двоичных данных.

Таблица 72 — Допустимые значения атрибута NonXMLBody.classCode (CNE)

Код	Описание
DOCBODY (тело документа) [по умолчанию]	Контекст, отличающий тело документа от заголовка документа

Таблица 73 — Допустимые значения атрибута NonXMLBody.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие

Таблица 74 — Допустимые значения атрибута NonXMLBody.confidentialityCode (CWE)

Код[2)]	Описание
N (обычный) (codeSystem = 2.16.840.1.11388.5.25)	Применяются обычные правила конфиденциальности (в соответствии с устоявшейся практикой в сфере здравоохранения), то есть получить доступ к этому содержанию могут только авторизованные лица, которым этот доступ необходим для выполнения юридически разрешенных медицинских или деловых действий
R (средний) (codeSystem = 2.16.840.1.11388.5.25)	Средние ограничения доступа, например, документ могут читать только медицинские работники, имеющие непосредственное отношение к текущей медицинской помощи пациенту
V (высокий) (codeSystem = 2.16.840.1.11388.5.25)	Высокие ограничения доступа, заданные политикой конфиденциальности

5.4.3.1.2 StructuredBody (структурированное тело)

Класс StructuredBody представляет тело АКД-документа, состоящее из одного или нескольких разделов (класс Section).

Таблица 75 — Допустимые значения атрибута StructuredBody.classCode (CNE)

Код	Описание
DOCBODY (тело документа) [по умолчанию]	Контекст, отличающий тело документа от заголовка документа

Таблица 76 — Допустимые значения атрибута StructuredBody.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие

[2)] Значение свойства codeSystem указано здесь в связи с тем, что атрибут confidentialityCode имеет тип данных CE и, следовательно, в нем должны быть указаны и код (code), и система кодирования (codeSystem).

Таблица 77 — Допустимые значения атрибута StructuredBody.confidentialityCode (CWE)

Код[3)]	Описание
N (обычный) (codeSystem = 2.16.840.1.11388.5.25)	Применяются обычные правила конфиденциальности (в соответствии с устоявшейся практикой в сфере здравоохранения), то есть получить доступ к этому содержанию могут только авторизованные лица, которым этот доступ необходим для выполнения юридически разрешенных медицинских или деловых действий
R (средний) (codeSystem = 2.16.840.1.11388.5.25)	Средние ограничения доступа, например, документ могут читать только медицинские работники, имеющие непосредственное отношение к текущей медицинской помощи пациенту
V (высокий) (codeSystem = 2.16.840.1.11388.5.25)	Высокие ограничения доступа, заданные политикой конфиденциальности

Класс StructuredBody ассоциируется с одним или несколькими экземплярами класса Section с помощью связи component.

Таблица 78 — Допустимые значения атрибута component.typeCode (CNE)

Код	Описание
COMP (компонент) [по умолчанию]	Ассоциированное тело документа является компонентом документа

5.4.3.2 Атрибуты класса Section

Разделы документа могут быть вложенными. В разделе может быть переопределен контекст, наследуемый от заголовка (см. 5.4.4). Раздел может содержать повествовательный блок и подразделы.

В схеме АКД-документов к элементу section добавлен атрибут ID, имеющий тип данных XML ID. Этот атрибут является целевым в ссылках <linkHtml> (см. 5.4.3.5.2). Все значения атрибута типа XML ID должны быть уникальны в пределах документа (в соответствии со спецификацией XML организации W3C).

Таблица 79 — Допустимые значения атрибута Section.classCode (CNE)

Код	Описание
DOCSECT (раздел документа) [по умолчанию]	Контекст, обеспечивающий структурирование тела документа. С помощью деления документа на разделы читателю дается представление о том, какое можно ожидать содержание. Разделы используются для организации содержания тела документа и согласованности его частей

Таблица 80 — Допустимые значения атрибута Section.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие

5.4.3.2.1 Атрибут Section.id

Содержит уникальный идентификатор конкретного экземпляра раздела документа.

5.4.3.2.2 Атрибут Section.code

Содержит код, указывающий вид раздела (например, «Жалобы», «Осмотр», «Оценка состояния»). Допустимый набор значений берется из классификатора LOINC и имеет квалификатор расширяемости CWE.

5.4.3.2.3 Атрибут Section.title

Содержит название раздела. Если ему присвоено значение, то при визуализации оно должно выводиться вместе с повествовательным содержанием тела клинического документа.

[3)] Значение свойства codeSystem указано здесь в связи с тем, что атрибут confidentialityCode имеет тип данных CE и, следовательно, в нем должны быть указаны и код (code), и система кодирования (codeSystem).

5.4.3.2.4 Атрибут Section.text

Содержит повествовательное содержание, предназначенное для визуализации. Его называют также повествовательным блоком АКД-документа. Детальные сведения см. в 4.3.5.

5.4.3.2.5 Атрибут Section.confidentialityCode

Значение атрибута Section.confidentialityCode переопределяет код конфиденциальности, наследуемый от класса StructuredBody (структурированное тело). Детальные сведения см. в 5.4.4.

Таблица 81 — StructuredBody.confidentialityCode (CWE)

Код[4)]	Описание
N (обычный) (codeSystem = 2.16.840.1.11388.5.25)	Применяются обычные правила конфиденциальности (в соответствии с устоявшейся практикой в сфере здравоохранения), то есть получить доступ к этому содержанию могут только авторизованные лица, которым этот доступ необходим для выполнения юридически разрешенных медицинских или деловых действий
R (средний) (codeSystem = 2.16.840.1.11388.5.25)	Средние ограничения доступа, например, документ могут читать только медицинские работники, имеющие непосредственное отношение к текущей медицинской помощи пациенту
V (высокий) (codeSystem = 2.16.840.1.11388.5.25)	Высокие ограничения доступа, заданные политикой конфиденциальности

5.4.3.2.6 Атрибут Section.languageCode

Указывает код языка, на котором представлены строковые данные раздела (применяется как к содержанию, так и к значениям атрибутов). Значениями атрибута являются идентификаторы языка, определенные в документе IETF (Internet Engineering Task Force) RFC 3066 (по ред. H. Alvestrand. 1995), заменяющем устаревший документ RFC 1766. В стандартах HL7 для этих кодов используется система кодирования 2.16.840.1.113883.6.121.

Значение атрибута Section.languageCode переопределяет код языка, наследуемый от класса StructuredBody (структурированное тело). Детальные сведения см. в 5.4.4.

5.4.3.3 Участники раздела

5.4.3.3.1 Участник author (автор)

Участник author, описанный выше в подразделе 0

«Участник author (автор)» может быть приписан разделу. В этом случае он переопределяет авторство, наследуемое от заголовка АКД-документа.

5.4.3.3.2 Участник informant (информатор)

Участник informant, описанный выше в подразделе 5.4.2.2.6 «Участник informant (информатор)», может быть приписан разделу. В этом случае он переопределяет источники информации, наследуемые от заголовка АКД-документа.

5.4.3.3.3 Участник subject (субъект)

Участник subject представляет основной субъект содержания подразделов документа. В большинстве случаев им будет то лицо, чья медицинская карта указана в атрибуте recordTarget (см. подраздел 5.4.2.2.11 «Участник recordTarget (целевая медицинская карта)»), но он может и отличаться, как, например, при акушерском ультразвуковом исследовании, когда субъектом является плод.

Участник subject может быть приписан как разделу, так и подразделу АКД-документа. Он наследуется вложенными компонентами, если только не переопределен в них. Субъектом документа считается пациент.

Субъектом является лицо, выполняющее одно из нескольких возможных ролей (класс RelatedSubject). Сущностью, выполняющей роль, является лицо (класс SubjectPerson).

Таблица 82 — Допустимые значения атрибута subject.typeCode (CNE)

Код	Описание
SBJ (субъект) [по умолчанию]	Основной субъект документированных действий

[4)] Значение свойства codeSystem указано здесь в связи с тем, что атрибут confidentialityCode имеет тип данных CE и, следовательно, в нем должны быть указаны и код (code), и система кодирования (codeSystem).

Таблица 83 — Допустимые значения атрибута subject.contextControlCode (CNE)

Код	Описание
OP (замещающая, распространяемая) [по умолчанию]	Ассоциация участника, замещающая ассоциацию с тем же значением атрибута typeCode. Эта замещающая ассоциация будет распространяться на действия-потомки, которые связаны с данным экземпляром класса Act с помощью экземпляра класса ActRelationships (см. 5.4.4)

Таблица 84 — Допустимые значения атрибута RelatedSubject.classCode (CNE)

Код	Описание
PRS (персональное отношение) [по умолчанию]	Субъект имеет персональные отношения с пациентом. Тип персональных отношений передается в атрибуте SubjectRole.code и берется из расширяемого словарного домена PersonalRelationshipRoleType (с квалификатором CWE). Контролирующей сущностью всегда предполагается пациент
PAT (пациент)	Субъектом раздела является пациент, идентифицированный в атрибуте recordTarget заголовка АКД-документа

Таблица 85 — Допустимые значения атрибута SubjectPerson.classCode (CNE)

Код	Описание
PSN (физическое лицо) [по умолчанию]	Живой организм вида homo sapiens

Таблица 86 — Допустимые значения атрибута SubjectPerson.determinerCode (CNE)

Код	Описание
INSTANCE (экземпляр) [по умолчанию]	Определитель INSTANCE указывает фактический экземпляр сущности, в отличие от определителя KIND, который относится к типу сущностей. Например, можно говорить о конкретном автомобиле (об экземпляре автомобиля) или об автомобилях вообще (о типе транспортного средства)

5.4.3.4 Отношения, указанные в разделе Section

5.4.3.4.1 Отношение component (компонент)

Отношение «компонент» используется для вложения одного раздела в другой. Контекст распространяется на вложенный раздел (см. 5.4.4).

Таблица 87 — Допустимые значения атрибута component.typeCode (CNE)

Код	Описание
COMP (компонент) [по умолчанию]	Вложенный раздел является компонентом внешнего раздела.

5.4.3.4.2 Отношение entry (подраздел)

Отношения между разделами и его подразделами кодируются с помощью промежуточного отношения entry.

Примечание — Обсуждение ссылок внутрь повествовательного блока и ссылок из повествовательного блока приведены в 5.4.3.5.12.

Повествовательный блок каждого раздела, включая мультимедийное содержание, на которое есть ссылки в этом блоке, образует полное заверяемое содержание раздела. Мультимедийное содержание образуется подразделами ObservationMedia и RegionOfInterest, на которые есть ссылки в тегах <renderMultimedia>, включенные в повествовательный блок Section.text. Это единственная ситуация, когда подразделы содержат заверяемое содержание, которое должно выводиться вместе с содержанием повествовательного блока.

В терминах отношений между разделом и его подразделами стандарт АКД определяет общий вариант по умолчанию и более специфичный вариант, который может быть использован при необходимости.

По умолчанию отношение между разделом и подразделом имеет код «COMP» (компонент). Этот вариант рассчитан на общий случай, когда единственное утверждение об отношении состоит в том, что связанные подразделы содержатся в подразделе и никакая другая семантика не подразумевается. В этом случае исходным заверяемым содержанием является повествовательный блок раздела. Подразделы АКД-документа создаются разными способами (например, автоматизированная обработка текста на естественном языке, ручная кодировка, структурированный ввод данных, при котором одновременно создаются подразделы и блок текста). Метод создания подраздела может быть указан в атрибуте участника подраздела (содержащего, например, идентификацию алгоритма обработки или лица, которое обеспечило создание подраздела). Отношения между различными подразделами (например, между двумя подразделами Observation (результат исследования) или подразделом Observation и подразделом ObservationMedia (мультимедийный результат исследования)) могут быть представлены с помощью типов отношений, определенных в 4.3.8.4.

Если подразделы содержат информацию, не являющуюся частью клинического документа, то раздел, в который они вложены, может не иметь повествовательного блока. Протокол исследования может содержать вложенную информацию, идентифицирующую подтверждающие данные, реагенты, калибраторы, а также другие сведения, которые могут использоваться для дальнейшей обработки, но не являются частью клинического содержания. Такие подразделы также связаны с разделом, используя код типа отношения «COMP».

Код отношения «DRIV» (является производным) может использоваться в том особом случае, когда повествовательный блок полностью произведен из содержания подразделов entry. Когда протокол исследования, полностью состоящий из структурированных подразделов, преобразуется в АКД-документ, то приложение, выполняющее это преобразование, должно гарантировать, что заверяемое содержание (повествовательный блок и мультимедийные данные) является правильным и полностью обеспечивает отображение клинического содержания структурированных данных первоисточника. Тем самым, как и во всех АКД-документах, обеспечивается полнота представления повествовательного блока и мультимедийных данных как заверяемого содержания раздела. В этом случае повествовательный блок и мультимедийные данные не содержат никакой клинической информации, которой не было бы в подразделах. Примером может служить структурированный протокол акушерского ультразвукового исследования, соответствующий стандарту DICOM Structured Reporting, предоставляемый результаты в табличной форме с помощью программы, преобразующей формат DICOM Structured Reporting в повествовательный блок АКД-документа. Если атрибут typeCode отношения, связывающего такие подразделы с разделом, имеет значение «DRIV», то приложение-получатель должно исходить из того, что: 1) источником повествовательного блока являются подразделы; 2) содержание подразделов и повествовательного блока эквивалентно.

У подразделов, вложенных в один раздел, могут быть разные типы отношения с разделом. В этом случае объединение подразделов, являющихся целевыми в отношении типа «DRIV», используется для генерирования содержания повествовательного блока и эквивалентно ему по содержанию. Дополнительные подразделы, имеющие тип отношения «COMP» и содержащиеся в том же самом разделе, не добавляют к содержанию раздела отдельную семантику.

Таблица 88 — Допустимые значения атрибута entry.typeCode (CNE)

Код	Описание
COMP (компонент) [по умолчанию]	Соответствующий подраздел является компонентом раздела. Никакие семантические отношения не подразумеваются
DRIV (является производным)	Повествовательный блок производится от содержания подразделов и не включает в себя никакую клиническую информацию, которая не вытекает из этого содержания

5.4.3.5 Повествовательный блок раздела

Как указано в 5.1.3 «Соответствие стандарту АКД», для хранения отображаемого повествовательного содержания используется атрибут Section.text. Поэтому он называется повествовательным блоком АКД-документа.

XML-схема повествовательного блока приведена в приложении В.

Модель содержания этой схемы специально доработана вручную, чтобы обеспечить изложенные выше требования к визуализации повествовательного блока (см. подраздел 5.1.2.3 «Человекочитаемость и визуализация документов АКД»). Схема зарегистрирована в качестве типа MIME с идентификатором «text/x-hl7-text+xml», являющегося фиксированным типом среды атрибута Section.text. Компоненты этой схемы описаны в следующих подразделах.

5.4.3.5.1 Элемент <content>

Элемент <content> используется для включения в повествовательный блок строки или текста, доступный для внешних ссылок или отображаемый специальным образом. Элемент <content> может быть рекурсивно вложенным, что позволяет разбивать строку или неформатированный текст на любые желаемые порции.

Элемент <content> содержит необязательный идентификатор, который может служить целью ссылки. Все значения атрибута типа XML ID должны быть уникальны в пределах документа (в соответствии со спецификацией XML организации W3C). В компоненте originalText (исходный текст) атрибута модели RIM, входящем в любой подраздел АКД-документа, может быть указана явная ссылка на идентификатор элемента <content>, тем самым обозначающая ту часть повествовательного блока, которая ассоциируется с этим атрибутом подраздела.

Пример 5 —

```
<section>
  <code code="10153-2"
   codeSystem="2.16.840.1.113883.6.1"
   codeSystemName="LOINC"/>
  <title>Анамнез заболевания</title>
  <text>
   Имеющееся заболевание: <content ID="a1">Астма</content>
  </text>
  <entry>
    <observation classCode="OBS" moodCode="EVN">
      <code code="195967001"
       codeSystem="2.16.840.1.113883.6.96"
       codeSystemName="SNOMED CT"
       displayName="Asthma">
        <originalText>
          <reference value="#a1"/>
        </originalText>
      </code>
      <statusCode code="completed"/>
    </observation>
  </entry>
</section>
```

Использование ссылок из подразделов в повествовательный блок раздела не является обязательным. Механизм ссылок может использоваться в тех случаях, когда важно связать компонент исходного текста с кодом, указанным в подразделе.

Элемент <content> содержит необязательный атрибут revised, который может содержать значения «insert» (вставлен) или «delete» (удален). Его можно использовать для указания изменений повествовательного содержания, сделанных в последней версии АКД-документа. Назначение этого атрибута ограничено отражением изменений предыдущей версии документа. Его применение, если оно имеет место, должно сочетаться со стандартным механизмом отслеживания изменений документа, предложенным в настоящем стандарте. Изменения в АКД-документе, переданном в использование при оказании медицинской помощи пациенту, требуют формального пересмотра и отслеживания версий. В повествовательные блоки пересмотренного документа могут включаться необязательные атрибуты revised, помечающие внесенные изменения. Получатели документа должны интерпретировать атрибут revised

при отображении повествовательного блока и визуально отличать удаленную часть блока или скрывать его содержание.

5.4.3.5.2 Элемент <linkHtml>

Элемент <linkHtml>, используемый в повествовательном блоке раздела, аналогичен, но не идентичен тегу <a> в HTML. Он используется для ссылок на внутренние части документа или на внешнее содержание.

Для описания мультимедийных данных, являющихся неотъемлемой частью документа и заверяемого содержания, необходимо использовать подраздел, содержащий класс ObservationMedia. В повествовательный блок должен включаться элемент <renderMultiMedia> со ссылкой на этот подраздел (см. 5.4.3.5.6). Для ссылки на мультимедийные данные, не являющиеся неотъемлемой частью документа, можно использовать тег <linkHtml>.

Идентификатор связанного содержания указывается в атрибуте linkHtml.href. При внутренних ссылках его значением является идентификатор типа XML ID, уже существующий в другом элементе того же или иного повествовательного блока или добавленный в элементы <section>, <ObservationMedia> или <renderMultiMedia> XML-схемы. Атрибут linkHtml.name запрещен, поскольку атрибуты типа XML ID представляют собой альтернативные и более согласованные цели ссылок. Следуя соглашениям, принятым в HTML, значению внутренней ссылки должен предшествовать знак фунта, как это показано в следующем примере.

Пример 6 —

```
<section ID="SECT001">
  <code code="10164-2" codeSystem="2.16.840.1.113883.6.1"
   codeSystemName="LOINC"/>
  <title>Анамнез заболевания</title>
  <text>Г-н Смит, 57 лет, мужчина, жалуется на боль в груди
   3 года назад он перенес инфаркт миокарда, ...
  </text>
</section>

...

<section ID="SECT003">
  <code code="10153-2" codeSystem="2.16.840.1.113883.6.1"
   codeSystemName="LOINC"/>
  <title>Предшествующие заболевания</title>
  <text>Заболевание коронарных артерий в анамнезе, см.
   <linkHtml href="#SECT001">выше</linkHtml>.</text>
</section>
```

Такие ссылки в АКД-документах не типизированы. Для установления связей определенных типов следует включать в АКД-документ подразделы с указанием соответствующих типов отношений. От получателя не требуется визуализация внешней или внутренней ссылки, равно как и цели внешней ссылки.

5.4.3.5.3 Элементы <sub> и <sup>

Элементы <sub> и <sup> используются для выделения в повествовательном блоке нижних и верхних индексов соответственно.

При отображении получатели должны интерпретировать эти элементы, обеспечивая визуальное отличие нижних и верхних индексов.

5.4.3.5.4 Элемент

Элемент
 используется в АКД-документе для указания принудительного перехода к началу следующей строки. Он отличается от элемента <paragraph> тем, что не имеет содержания. При отображении получатели должны интерпретировать этот элемент как переход к началу следующей строки.

5.4.3.5.5 Элементы <footnote> и <footnoteRef>

Элемент <footnote> используется в АКД-документе для указания сноски. Он содержит текст сноски, включаемый внутрь потока текста, к которому она применяется.

С помощью элемента <footnoteRef> можно сослаться на уже существующую сноску, включенную в тот же или иной повествовательный блок данного документа. Он нужен в том случае, если одна и та же сноска используется несколько раз. Атрибуту footnoteRef.IDREF должно быть присвоено значение атрибута footnote.ID сноски, существующей в данном документе.

При отображении получатели документа должны интерпретировать их, визуально отличая текст сноски. Точный способ отображения оставляется на усмотрение получателя, например, пометка текста в месте сноски с гиперссылкой на содержание сноски, простое выделение, к примеру, «Это текст со сноской [это текст сноски]» и т. д.

5.4.3.5.6 Элемент <renderMultiMedia>

Элемент <renderMultiMedia> используется в АКД-документе для ссылки на внешние мультимедийные данные, являющиеся неотъемлемой частью документа и его заверяемого содержания. Он показывает, в какое место отображения повествовательного блока должны быть включены эти мультимедийные данные.

В элемент <renderMultiMedia> может быть вложен необязательный элемент <caption> (заголовок). Элемент <renderMultiMedia> имеет обязательный атрибут referencedObject (типа XML IDREFS), значение которого должно быть равно значению XML-атрибута ID подраздела типа ObservationMedia или RegionOfInterest, содержащегося в том же самом документе.

Пример 7 —

```
<section>
  <code code="8709-8" codeSystem="2.16.840.1.113883.6.1"
   codeSystemName="LOINC"/>
  <title>Skin exam</title>
   <text>Эритемная сыпь на внутренней поверхности левого указательного пальца.<renderMultiMedia
referencedObject=»MM1»/>
  </text>
  <entry>
    <observationMedia classCode="OBS" moodCode="EVN" ID="MM1">
      <id root="2.16.840.1.113883.19.2.1"/>
      <value xsi:type="ED" mediaType="image/jpeg">
        <reference value="left_hand_image.jpeg"/>
      </value>
    </observationMedia>
  </entry>
</section>
```

Для ссылки на те мультимедийные данные, которые не являются неотъемлемой частью документа, должен использоваться элемент <linkHtml>.

Ожидается, что мультимедийные данные, на которые есть ссылка в повествовательном блоке, будут отображены или обозначены на месте ссылки. Элемент <renderMultiMedia> может содержать ссылку на один класс ObservationMedia (среда результата исследования) или на один либо несколько классов RegionOfInterest (область интереса). Если этот элемент ссылается на класс ObservationMedia, то содержание этого класса должно быть отображено или обозначено на месте ссылки. Если элемент <renderMultiMedia> содержит ссылку на один либо несколько классов RegionOfInterest, то все области интереса должны быть отображены или обозначены на месте ссылки поверх тех мультимедийных данных, к которым они относятся. Если этот элемент ссылается на несколько областей интереса, то каждая из них должна относиться к тем же самым мультимедийным данным.

5.4.3.5.7 Элемент <paragraph>

Элемент <paragraph> аналогичен тегу <p> в HTML. Он позволяет разбивать повествовательные блоки на логически связанные структуры. В элемент <paragraph> может быть вложен необязательный элемент <caption> (заголовок). Если он присутствует, то должен предшествовать всем остальным строковым данным.

5.4.3.5.8 Элемент <list>

Элемент <list> аналогичен списку в HTML. В элемент <list> может быть вложен необязательный элемент <caption> (заголовок), а также один или несколько элементов строк списка <item>. В эле-

мент <item> может быть вложен необязательный элемент <caption>. Если он присутствует, то должен предшествовать всем остальным строковым данным. В обязательном атрибуте listType должно быть указано, является ли список упорядоченным или неупорядоченным (по умолчанию он считается неупорядоченным). Строки неупорядоченных списков обычно маркируются кружочками, а упорядоченных — номерами, хотя это и не обязательно.

5.4.3.5.9 Элемент <table>

Элемент <table> аналогичен таблице в HTML. Таблица размечается только для целей отображения и, в отличие от таблицы базы данных, не содержит значащие имена столбцов.

В настоящем стандарте модель таблицы XHTML модифицирована следующим образом: теги форматирования удалены и модель содержания ячеек таблицы принята аналогичной модели содержания других элементов повествовательного блока АКД-документа.

Атрибуты рамок таблицы table.border, отступов в ячейке table.cellpadding и промежутков между ячейками table.cellspacing запрещены, поскольку с помощью атрибута styleCode (см. 5.4.3.5.11) отправители документа могут более подходящим способом задать характеристики отображения таблицы.

5.4.3.5.10 Элемент <caption>

Элемент <caption> задает заголовок абзаца <paragraph>, списка <list>, строки списка <item>, таблицы <table> или ячейки таблицы. Он может также использоваться в элементе <renderMultiMedia> в качестве заголовка изображения (класс ObservationMedia) или области интереса (класс RegionOfInterest). Элемент caption содержит неформатированный текст и может содержать ссылки и сноски.

5.4.3.5.11 Атрибут styleCode

С помощью атрибута styleCode (код стиля) автор повествовательного блока АКД-документа может задать желаемые характеристики визуализации вложенных строковых данных. Получатели документа не обязаны отображать документы в соответствии с предложенными стилями и могут форматировать стилизованный текст в соответствии с собственными соглашениями о стилях.

Значения кодов стилей берутся из словарного домена HL7 styleType и допускают расширение (квалификатор расширяемости CWE).

Таблица 89 — Допустимые значения атрибута styleCode (CWE)

Код	Описание
Стиль шрифта (определяет характеристики отображения шрифта)	
Bold	Полужирный шрифт
Underline	Подчеркнутый шрифт
Italics	Курсивный шрифт
Emphasis	Некоторый способ выделения текста
Стиль обрамления таблицы (определяет характеристики отображения ячеек)	
Lrule	Выравнивание содержания ячейки по левому краю
Rrule	Выравнивание содержания ячейки по правому краю
Toprule	Выравнивание содержания ячейки по верхнему краю
Botrule	Выравнивание содержания ячейки по нижнему краю
Стиль упорядоченного списка (определяет характеристики отображения упорядоченных списков)	
Arabic	Нумерация списка арабскими цифрами (1, 2, 3, …)
LittleRoman	Нумерация списка строчными римскими цифрами (i, ii, iii, …)
BigRoman	Нумерация списка прописными римскими цифрами (I, II, III, …)
LittleAlpha	Нумерация списка строчными латинскими буквами (a,b, c, …)
BigAlpha	Нумерация списка прописными латинскими буквами (A, B, C, …)
Стиль неупорядоченного списка (определяет характеристики отображения неупорядоченных списков)	

Окончание таблицы 89

Код	Описание
Disc	Маркировка списка маленькими сплошными кружками
Circle	Маркировка списка маленькими окружностями
Square	Маркировка списка маленькими сплошными квадратами

Местные расширения словарного домена styleType должны придерживаться следующего шаблона в формате кода: [x][A-Za-z][A-Za-z0-9]* (первый символ «x», второй прописная или строчная буква из диапазона A-Z, остальные символы могут быть любой комбинацией прописных или строчных букв из диапазона A-Z и цифр).

Атрибут styleCode может содержать несколько кодов стиля, разделенных пробельным символом. Если элемент, содержащий непустой атрибут styleCode, вложен в другой элемент, который также содержит этот атрибут, то, как показано в следующем примере, стили должны объединиться.

Пример 8 —

```
<section>
  <text><content styleCode=»Bold»>Отображается полужирным шрифтом,
   <content styleCode=»Italics»>отображается полужирным
   курсивом,</content>отображается полужирным шрифтом.</content>
   <content styleCode="Bold Italics">Также отображается
   полужирным курсивом.</content>
  </text>
</section>
```

5.4.3.5.12 Ссылки внутрь повествовательного блока и из него

Примечание — Обсуждение связей между разделом и входящими в него подразделами приведено в подразделе 0 «

Отношение entry (подраздел)».

Ниже перечислены способы ссылок внутрь повествовательного блока АКД-документа и из него:

- в подразделе entry может быть указана ссылка на элемент <content> повествовательного блока (см. подраздел 5.4.3.5.1 «Элемент <content>»);
- в элементе <linkHtml> повествовательного блока АКД-документа могут быть указаны ссылки на внешние или внутренние источники (см. подраздел 5.4.3.5.2 «Элемент <linkHtml>»);
- элемент <footnoteRef> повествовательного блока АКД-документа может содержать ссылку на элемент <footnote>, содержащийся в тот же самом или другом повествовательном блоке данного документа (см. подраздел 5.4.3.5.5 «Элементы <footnote> и <footnoteRef>»);
- элемент <renderMultiMedia> повествовательного блока АКД-документа может содержать ссылку на подраздел типа ObservationMedia или RegionOfInterest, содержащийся в данном документе (см. подраздел 5.4.3.5.6 «Элемент <renderMultiMedia>»).

5.4.3.6 Типы подразделов entry

Подразделы entry используются для передачи структурированных компонентов раздела section, предназначенных для машинной обработки. Каждый раздел section может содержать любое число подразделов entry или ни одного.

Содержание клинических документов включает в себя широкий спектр информации, для полного и завершенного кодирования которой необходимо использовать значительную часть информационных объектов, описанных в модели RIM. Текущие типы подразделов АКД-документа разработаны в соответствии с заданными требованиями и сценариями, входящими в область применения АКД. В стандарте принят подход, согласно которому с помощью анализа похожих требований вместо большого числа узкоспециализированных подразделов entry, разработанных для каждого сценария, определены достаточно универсальные структуры, которые могут быть специализированы с учетом национальных или местных условий. Аналогичные подходы использованы в стандартах организации CEN и стандарте DICOM, а также в проекте OpenEHR.

Модель подразделов entry АКД-документа произведена от общей модели HL7 Clinical Statement, для разработки которой были объединены усилия нескольких комитетов, предпринятые в целях обеспечения согласованного представления результатов клинических исследований и действий в различных спецификациях стандартов V3.

5.4.3.6.1 Класс Act (действие)

Структура класса Act произведена от одноименного класса модели RIM. Подразделы этого типа используются в том случае, если более специфичные классы оказываются недостаточно подходящими.

Если атрибут Act.negationInd (признак отрицания) имеет значение «true», то это означает отрицание всего содержания класса Act, за исключением таких атрибутов, как Act.id и Act.moodCode, а также участников действия. Эти атрибуты сохраняют свое значение, к примеру, автор остается автором и у отрицательных сведений. Содержание класса Act, для которого указан атрибут negationInd, по-прежнему остается утверждением о конкретном факте, описываемом этим классом. Например, отрицание утверждения «1 июля наблюдалась одышка» означает, что автор определенно отрицает, что одышка наблюдалась 1 июля, и берет на себя ту же ответственность за это утверждение и придерживается тех же требований к его доказательству, как если бы он не использовал отрицание.

Таблица 90 — Допустимые значения атрибута Act.classCode (CNE)

Код	Описание
ACT (действие)	Медицинская помощь.
ACCM (размещение)	Размещение является услугой лицу или другому живому организму, состоящей в предоставлении места, в котором этот субъект может находиться в течение некоторого времени
CONS (согласие)	Информированные согласия и все аналогичные медико-юридические процедуры между пациентом (или его законным представителем) и поставщиком медицинской помощи
CTTEVENT (событие клинического испытания)	Момент времени при проведении клинического испытания, когда планируется выполнение одного или нескольких действий (moodCode = «INT») или когда такие действия фактически выполнены (moodCode = «EVN»)
INC (инцидент)	Событие вне контроля одной или нескольких участвующих сторон, например, несчастный случай
INFRM (информирование)	Акт передачи субъекту определенной информации или знаний в некоторой предметной области (или запрос на ее передачу)
PCPR (оказание медицинской помощи)	Оказание медицинской помощи пациенту или группе пациентов. Ответственность возлагается на исполнителя
REG (регистрация)	Регистрация информации о сущности или ее роли
SPCTRT (обработка образца)	Процедура или обработка, необходимая для подготовки образца к выполнению анализа

Таблица 91 — Допустимые значения атрибута Act.moodCode (CNE)

Код	Описание
EVN (событие)	В подразделе описано фактически произошедшее событие
INT (намерение)	В подразделе описано намерение или план
APT (действие, запланированное в расписании)	Действие, описанное в подразделе, запланировано на определенное время в определенном месте
ARQ (запрос на планирование в расписании)	Содержание подраздела представляет собой запрос на планирование в расписании
PRMS (подтверждение)	Подтверждение возможности выполнения действия, описанного в подразделе
PRP (предложение)	Предложение выполнения действия, описанного в подразделе

Окончание таблицы 91

Код	Описание
RQO (требование)	Требование или заказ выполнения действия, описанного в подразделе
DEF (описание)	Подраздел представляет собой описание услуги в нормативно-справочной информации

5.4.3.6.2 Класс Encounter (посещение)

Структура класса Encounter произведена от класса PatientEncounter модели RIM. Подразделы этого типа используются для идентификации связанных посещений, например, ранее состоявшихся или повторных посещений.

Примечание — Класс EncompassingEncounter, используемый в заголовке АКД-документа (см. подраздел 5.4.2.3 «Отношения, указанные в заголовке»), предназначен для указания места посещения, во время которого произошло документируемое действие. Класс Encounter в теле документа используется для представления информации о других связанных посещениях.

Таблица 92 — Допустимые значения атрибута Encounter.classCode (CNE)

Код	Описание
ENC (посещение)	Контакт пациента с одним или несколькими медицинскими работниками, в процессе которого пациенту оказана медицинская помощь или проведена оценка состояния его здоровья

Таблица 93 — Допустимые значения атрибута EncompassingEncounter.moodCode (CNE)

Код	Описание
INT (намерение)	В подразделе описано намерение или план
EVN (событие)	В подразделе описано фактически произошедшее событие
APT (действие, запланированное в расписании)	Действие, описанное в подразделе, запланировано на определенное время в определенном месте
ARQ (запрос на планирование в расписании)	Содержание подраздела представляет собой запрос на планирование в расписании
PRMS (подтверждение)	Подтверждение возможности выполнения действия, описанного в подразделе
PRP (предложение)	Предложение выполнения действия, описанного в подразделе
RQO (требование)	Требование или заказ выполнения действия, описанного в подразделе

5.4.3.6.3 Класс Observation (исследование)

Структура класса Observation произведена от одноименного класса модели RIM. Подразделы этого типа используются для представления кодируемых и других исследований.

Если атрибут Observation.negationInd (признак отрицания) имеет значение «true», то это означает отрицание всего содержания класса Observation, за исключением таких атрибутов, как Observation. id и Observation.moodCode, а также участников действия. Эти атрибуты сохраняют свое значение, к примеру, автор остается автором и у отрицательных сведений об исследовании. Содержание класса Observation, для которого указан атрибут negationInd, по-прежнему остается утверждением о конкретном факте, описываемом этим классом. Например, отрицание утверждения «1 июля наблюдалась одышка» означает, что автор определенно отрицает, что одышка наблюдалась 1 июля, и берет на себя ту же ответственность за это утверждение и придерживается тех же требований к его доказательству, как если бы он не использовал отрицание.

Таблица 94 — Допустимые значения атрибута Observation.classCode (CNE)

Код	Описание
OBS (исследование)	Исследованиями являются действия, выполняемые для получения ответа или результата
Любой подтип типа OBS	Допустимые значения берутся из словарного домена ActClassObservation

Таблица 95 — Допустимые значения атрибута EncompassingEncounter.moodCode (CNE)

Код	Описание
EVN (событие)	В подразделе описано фактически произошедшее событие
DEF (описание)	Подраздел представляет собой описание исследования в нормативно-справочной информации
GOAL (цель)	В подразделе описана цель или намерение исследования
INT (намерение)	В подразделе описано намерение или план
PRMS (подтверждение)	Подтверждение возможности выполнения действия, описанного в подразделе
PRP (предложение)	Предложение выполнения действия, описанного в подразделе
RQO (требование)	Требование или заказ выполнения действия, описанного в подразделе

Класс Observation (исследование) может иметь несколько отношений типа referenceRange (справочный диапазон), связывающих его с классом ObservationRange, в экземплярах которого могут быть указаны диапазоны ожидаемых результатов конкретного исследования.

Таблица 96 — Допустимые значения атрибута referenceRange.classCode (CNE)

Код	Описание
REFV (имеет справочные значения) [по умолчанию]	Справочные диапазоны являются по своей сути описателями класса значений результатов исследований, которые объявляются «нормой», «патологией» или «критичными значениями». Этот тип связи может использоваться для описания триггеров, инициирующих тревожные сигналы при появлении критичных значений

Таблица 97 — Допустимые значения атрибута ObservationRange.classCode (CNE)

Код	Описание
OBS (исследование) [по умолчанию]	Исследованиями являются действия, выполняемые для получения ответа или результата
Любой подтип типа OBS	Допустимые значения берутся из словарного домена ActClassObservation

Таблица 98 — Допустимые значения атрибута ObservationRange.moodCode (CNE)

Код	Описание
EVN.CRT (критерий события) [по умолчанию]	Критерий или условие, применяемые к исследованиям, которые должны быть удовлетворены до выполнения соответствующей услуги

5.4.3.6.4 Класс ObservationMedia (мультимедийный результат исследования)

Класс ObservationMedia является производным от класса Observation модели RIM. Он представляет мультимедийные данные, которые логически являются частью текущего документа. Он используется только для тех мультимедийных данных, которые логически являются частью заверяемого содержания документа. Для визуализации данных, ссылка на которые содержится в классе ObservationMedia, необходимо программное обеспечение, способное обработать конкретный тип среды MIME.

В схеме АКД-документов к элементу ObservationMedia добавлен атрибут ID, имеющий тип данных XML ID. Этот атрибут является целевым в ссылках <renderMultiMedia> (см. подраздел 5.4.3.5.6 «Элемент <renderMultiMedia>»). Все значения атрибута типа XML ID должны быть уникальны в пределах документа (в соответствии со спецификацией XML организации W3C).

Различие между классами ObservationMedia и ExternalObservation состоит в том, что подразделы entry типа ObservationMedia являются частью заверяемого содержания, а подразделы entry типа ExternalObservation — нет. Например, если клиницист начертит рисунок как часть дневниковой записи, этот рисунок должен быть представлен с помощью класса ObservationMedia. А если он составляет описание рентгеновского снимка грудной клетки, то ссылка на рентгеновский снимок представляется с помощью класса ExternalObservation.

Таблица 99 — Допустимые значения атрибута ObservationRange.classCode (CNE)

Код	Описание
OBS (исследование)	Мультимедийный результат исследования

Таблица 100 — Допустимые значения атрибута ObservationRange.moodCode (CNE)

Код	Описание
EVN (событие)	В подразделе описано фактически произошедшее событие

5.4.3.6.5 Класс Organizer (группировка)

Класс Organizer является производным от класса Act модели RIM. Он может использоваться для группировки других подразделов АКД-документа, объединенных общим контекстом. С помощью отношений component (компонент) в подраздел типа Organizer могут быть включены другие подразделы типа Organizer или иных типов. С помощью отношений reference (ссылка) в подраздел типа Organizer может быть включена информация о внешних действиях. Подраздел типа Organizer не может быть источником отношения entryRelationship (связь между подразделами).

Примечание — Подразделы entry иных типов, например Observation, тоже могут включать в себя другие подразделы, но только с помощью отношений entryRelationship. Для группировки подразделов АКД-документа вовсе не требуется использовать подраздел типа Organizer.

Таблица 101 — Допустимые значения атрибута Organizer.classCode (CNE)

Код	Описание
BATTERY (панель)	Панель представляет собой комплекс исследований, которые обычно логически или практически группируются для широкоупотребительных клинических или функциональных целей. Например, в комплекс группируются исследования, выполняемые на одном автоанализаторе
CLUSTER (кластер)	Группа подразделов entry, имеющих логическую общность. С помощью класса Cluster их можно агрегировать в комплексное утверждение

Таблица 102 — Допустимые значения атрибута Organizer.moodCode (CNE)

Код	Описание
EVN (событие)	В подразделе описано фактически произошедшее событие

5.4.3.6.6 Класс Procedure (процедура)

Класс Procedure является производным от одноименного класса модели RIM. Он используется для представления информации о процедурах.

Если атрибут Procedure.negationInd (признак отрицания) имеет значение «true», то это означает отрицание всего содержания класса Procedure, за исключением таких атрибутов, как Procedure.id и Procedure.moodCode, а также участников действия. Эти атрибуты сохраняют свое значение; к примеру, автор остается автором и отрицательных сведений о процедуре. Содержание класса Procedure, для которого указан атрибут negationInd, по-прежнему остается утверждением о конкретном факте, описы-

ваемом этим классом. Например, отрицание утверждения «выполнена аппендэктомия» означает, что автор определенно отрицает, что аппендэктомия выполнялась, и берет на себя ту же ответственность за это утверждение и придерживается тех же требований к его доказательству, как если бы он не использовал отрицание.

Таблица 103 — Допустимые значения атрибута Procedure.classCode (CNE)

Код	Описание
PROC (процедура)	Непосредственным и основным результатом процедуры (постусловие) является измененное физическое состояние субъекта

Таблица 104 — Допустимые значения атрибута Procedure.moodCode (CNE)

Код	Описание
EVN (событие)	В подразделе описано фактически произошедшее событие
INT (намерение)	В подразделе описано намерение или план
APT (действие, запланированное в расписании)	Действие, описанное в подразделе, запланировано на определенное время в определенном месте
ARQ (запрос на планирование в расписании)	Содержание подраздела представляет собой запрос на планирование в расписании
PRMS (подтверждение)	Подтверждение возможности выполнения действия, описанного в подразделе
PRP (предложение)	Предложение выполнения действия, описанного в подразделе
RQO (требование)	Требование или заказ выполнения действия, описанного в подразделе
DEF (описание)	Подраздел представляет собой описание услуги в нормативно-справочной информации

5.4.3.6.7 Класс RegionOfInterest (область интереса)

Класс RegionOfInterest является производным от класса Observation модели RIM. Он представляет область интереса, выделенную на изображении с помощью наложения некоторой формы. Она используется для указания некоторых частей изображения, например для указания места физикального исследования, заключая в «окружность» часть схематического образа тела человека. Координаты, передаваемые в атрибуте RegionOfInterest.value, задаются в пикселах в виде массива целых чисел. Началом координат служит левый верхний угол, положительная часть оси X идет направо, а положительная часть оси Y — вниз. С помощью отношений entryRelationship может быть установлена связь между подразделом типа RegionOfInterest и подразделом типа ObservationMedia, а с помощью отношений reference — между подразделом типа RegionOfInterest и подразделом типа ExternalObservation (внешний результат исследования). При этом атрибут typeCode должен иметь значение «SUBJ». Подраздел типа RegionOfInterest может ассоциироваться ровно с одним подразделом типа ObservationMedia или с одним подразделом типа ExternalObservation. Если подраздел типа RegionOfInterest является целью ссылки, указанной в элементе <renderMultimedia>, то он может ассоциироваться только с подразделом типа ObservationMedia, а не ExternalObservation.

В схеме АКД-документов к элементу RegionOfInterest добавлен атрибут ID, имеющий тип данных XML ID. Этот атрибут является целевым в ссылках <renderMultiMedia> (см. подраздел 5.4.3.5.6 «Элемент <renderMultiMedia>»). Все значения атрибута типа XML ID должны быть уникальны в пределах документа (в соответствии со спецификацией XML организации W3C).

Таблица 105 — Допустимые значения атрибута RegionOfInterest.classCode (CNE)

Код	Описание
ROIOVL (наложение области интереса)	Область интереса выделяется на изображении с помощью наложения формы

Таблица 106 — Допустимые значения атрибута RegionOfInterest.moodCode (CNE)

Код	Описание
EVN (событие)	В подразделе описано фактически произошедшее событие

Таблица 107 — Допустимые значения атрибута RegionOfInterest.code (CNE)

Код	Описание
CIRCLE (круг)	Круг задается двумя парами чисел (столбец, строка). Первая пара задает центр круга, а вторая — точку на его окружности
ELLIPSE (эллипс)	Эллипс задается четырьмя парами чисел (столбец, строка). Первые две пары задают конечные точки большой оси, а следующие две пары — конечные точки малой оси
POINT (точка)	Одна точка, задаваемая парой чисел (столбец, строка), или несколько точек, задаваемых такими парами
POLY (полилиния)	Последовательность соединенных отрезков с упорядоченными концами, задаваемыми парами чисел (столбец, строка). Если первый конец совпадает с последним, то образуется замкнутый полигон

В следующем примере иллюстрируется схема использования области интереса. В этом примере клиницист обнаруживает сыпь при физикальном исследовании кожи и указывает ее расположение, рисуя область интереса поверх схематического изображения ладони, взятого из библиотеки изображений. В повествовательный блок включается элемент <renderMultiMedia> со ссылкой на подраздел entry типа RegionOfInterest, а этот подраздел, в свою очередь, содержит ссылку на образ ладони.

Пример 9 —

```
<section>
  <code code="8709-8" codeSystem="2.16.840.1.113883.6.1"
   codeSystemName="LOINC"/>
  <title>Skin Exam</title>
   <text> Эритемная сыпь на внутренней поверхности левого указательного пальца.<renderMultiMedia
referencedObject=»MM2»/>
  </text>
  <entry>
    <observation classCode="OBS" moodCode="EVN">
      <code code="271807003"
       codeSystem="2.16.840.1.113883.6.96"
       codeSystemName="SNOMED CT"
       displayName="Rash"/>
      <statusCode code="completed"/>
      <targetSiteCode code="48856004"
       codeSystem="2.16.840.1.113883.6.96"
       codeSystemName="SNOMED CT"
       displayName="Skin of palmer surface of index finger">
        <qualifier>
          <name code="78615007"
           codeSystem="2.16.840.1.113883.6.96"
           displayName="with laterality"/>
          <value code="7771000"
           codeSystem="2.16.840.1.113883.6.96"
           displayName="left"/>
        </qualifier>
```

```
      </targetSiteCode>
      <entryRelationship typeCode="SPRT">
        <regionOfInterest classCode="ROIOVL" moodCode="EVN" ID="MM2">
          <id root="2.16.840.1.113883.19.3.1"/>
          <code code="ELLIPSE"/>
          <value value="3"/>
          <value value="1"/>
          <value value="3"/>
          <value value="7"/>
          <value value="2"/>
          <value value="4"/>
          <value value="4"/>
          <value value="4"/>
          <entryRelationship typeCode="SUBJ">
            <observationMedia classCode="OBS" moodCode="EVN">
              <id root="2.16.840.1.113883.19.2.1"/>
              <value mediaType="image/jpeg">
                <reference value="lefthand.jpeg"/>
              </value>
            </observationMedia>
          </entryRelationship>
        </regionOfInterest>
      </entryRelationship>
    </observation>
  </entry>
</section>
```

5.4.3.6.8 Класс SubstanceAdministration (прием лекарства)

Класс SubstanceAdministration является производным от одноименного класса модели RIM. Он используется для представления информации о событиях, связанных с медикаментозной терапией, например об истории лекарственных назначений или о планируемом применении лекарственных средств.

Если атрибут SubstanceAdministration.negationInd (признак отрицания) имеет значение «true», то это означает отрицание всего содержания класса SubstanceAdministration, за исключением таких атрибутов, как SubstanceAdministration.id и SubstanceAdministration.moodCode, а также участников действия. Эти атрибуты сохраняют свое значение; к примеру, автор остается автором и отрицательных сведений о процедуре. Содержание класса SubstanceAdministration, для которого указан атрибут negationInd, по-прежнему остается утверждением о конкретном факте, описываемом этим классом. Например, отрицание утверждения «принимается аспирин» означает, что автор определенно отрицает, что аспирин принимается, и берет на себя ту же ответственность за это утверждение и придерживается тех же требований к его доказательству, как если бы он не использовал отрицание.

Таблица 108 — Допустимые значения атрибута SubstanceAdministration.classCode (CNE)

Код	Описание
SBADM (применение субстанции)	Действие введения или иного применения субстанции к субъекту

Таблица 109 — Допустимые значения атрибута SubstanceAdministration.moodCode (CNE)

Код	Описание
EVN (событие)	В подразделе описано фактически произошедшее событие
INT (намерение)	В подразделе описано намерение или план

Окончание таблицы 108

Код	Описание
PRMS (подтверждение)	Подтверждение возможности выполнения действия, описанного в подразделе
PRP (предложение)	Предложение выполнения действия, описанного в подразделе
RQO (требование)	Требование или заказ выполнения действия, описанного в подразделе

Атрибут SubstanceAdministration.priorityCode характеризует срочность применения субстанции. В атрибуте SubstanceAdministration.doseQuantity указана разовая доза препарата, а в атрибуте SubstanceAdministration.rateQuantity может быть указана частота применения или скорость введения (для внутривенных инфузий).

Атрибут SubstanceAdministration.maxDoseQuantity используется для указания максимальной дозы препарата, который может быть применен в заданном интервале времени (например, максимальная суточная доза морфина, максимальная жизненная доза доксорубицина). Атрибут SubstanceAdministration.effectiveTime используется для указания времени применения. Его значение задается согласно Общей спецификации времени GTS (General Timing Specification), позволяющей описать разные сценарии дозировки, как показано в следующем примере.

Пример 10 —

```
<section>
  <text>Принимать каптоприл по 25 мг перорально каждые 12 часов, начиная с 1-го января 2002 года и заканчивая 1-м февраля 2002 года.
  </text>
  <entry>
    <substanceAdministration classCode="SBADM" moodCode="RQO">
      <effectiveTime xsi:type="IVL_TS">
        <low value="20020101"/>
        <high value="20020201"/>
      </effectiveTime>
      <effectiveTime xsi:type="PIVL_TS" operator="A">
        <period value="12" unit="h"/>
      </effectiveTime>
      <routeCode code="PO"
       codeSystem="2.16.840.1.113883.5.112"
       codeSystemName="RouteOfAdministration"/>
      <doseQuantity value="1"/>
      <consumable>
        <manufacturedProduct>
          <manufacturedLabeledDrug>
            <code code="318821008"
             codeSystem="2.16.840.1.113883.6.96"
             codeSystemName="SNOMED CT"
             displayName="Captopril 25mg tablet"/>
          </manufacturedLabeledDrug>
        </manufacturedProduct>
      </consumable>
    </substanceAdministration>
  </entry>
</section>
```

Для представления информации, относящейся к применению лекарственных препаратов, предусмотрены отношения класса SubstanceAdministration с несколькими другими классами. Отношение

consumable (расходуемый) используется для связи с разделами типа LabeledDrug (маркированное лекарство) или Material (вещество), описывающими применяемую субстанцию. Класс LabeledDrug, являющийся клоном класса Entity и исполняющий роль произведенного продукта (класс Manufactured Product), используется для идентификации лекарственного препарата, расходуемого при применении субстанции. Лекарственный препарат идентифицируется кодом, передаваемым в атрибуте LabeledDrug. code, или наименованием, передаваемым в атрибуте LabeledDrug.name. Класс Material используется для идентификации веществ, не являющихся лекарственными препаратами, например вакцин и продуктов крови.

Таблица 110 — Допустимые значения атрибута consumable.typeCode (CNE)

Код	Описание
CSM (расходуемый) [по умолчанию]	Субстанция, расходуемая или потребляемая при ее применении

Таблица 111 — Допустимые значения атрибута ManufacturedProduct.classCode (CNE)

Код	Описание
MANU (произведенный) [по умолчанию]	Произведенный продукт

Таблица 112 — Допустимые значения атрибута LabeledDrug.classCode (CNE)

Код	Описание
MMAT (произведенный) [по умолчанию]	Произведенное вещество

Таблица 113 — Допустимые значения атрибута LabeledDrug.determinerCode (CNE)

Код	Описание
KIND (вид) [по умолчанию]	Определитель KIND указывает, что данный экземпляр класса Entity представляет общее описание вида сущностей, которые могут рассматриваться в целом, по частям или по отдельным элементам

Таблица 114 — Допустимые значения атрибута Material.classCode (CNE)

Код	Описание
MMAT (произведенный) [по умолчанию]	Произведенное вещество

Таблица 115 — Допустимые значения атрибута Material.determinerCode (CNE)

Код	Описание
KIND (вид) [по умолчанию]	Определитель KIND указывает, что данный экземпляр класса Entity представляет общее описание вида сущностей, которые могут рассматриваться в целом, по частям или по отдельным элементам

5.4.3.6.9 Класс Supply (поставка)

Класс Supply является производным от одноименного класса модели RIM. Он используется для представления информации о поставке продукта, осуществляемой одной сущностью в интересах другой сущности.

Таблица 116 — Допустимые значения атрибута Supply.classCode (CNE)

Код	Описание
SPLY (поставка)	Действие отпуска или доставки продукта

Таблица 117 — Допустимые значения атрибута Supply.moodCode (CNE)

Код	Описание
EVN (событие)	В подразделе описано фактически произошедшее событие
INT (намерение)	В подразделе описано намерение или план
PRMS (подтверждение)	Подтверждение возможности выполнения действия, описанного в подразделе
PRP (предложение)	Предложение выполнения действия, описанного в подразделе.
RQO (требование)	Требование или заказ выполнения действия, описанного в подразделе

Отпущенный продукт связывается с действием поставки Supply с помощью отношения product, связанного с той же самой ролью ManufacturedProduct, что используется для класса SubstanceAdministration.

Таблица 118 — Допустимые значения атрибута product.typeCode (CNE)

Код	Описание
PRD (продукт) [по умолчанию]	Целевой материал, который предоставляется (производится) при выполнении услуги

Класс Supply представляет отпуск, а класс SubstanceAdministration — применение. Лекарственные назначения являются сложными видами деятельности, охватывающими требования применения к пациенту (например, «принимать дигоксин таблетки 0,125 мг перорально 1 раз в день») и требования отпуска из аптеки (например, «отпустить 30 таблеток с 5 повторениями). Подобное назначение представляется в АКД с помощью подраздела entry типа SubstanceAdministration (применение субстанции), в который с помощью отношения component вложен подраздел типа Supply (запас). Атрибут Supply. independentInd (признак независимости) у вложенного подраздела может иметь значение false, указывающее, что соответствующая поставка не должна осуществляться отдельно от того, что указано в содержащем их подразделе типа SubstanceAdministration.

Пример 11 —

```
<section>
  <text>Дигоксин таблетки дигоксин таблетки 0,125 мг перорально 1 раз в день, N30, 5 повторений.</
text>
  <entry>
    <substanceAdministration classCode="SBADM" moodCode="RQO">
      <effectiveTime xsi:type="PIVL_TS">
        <period value="24" unit="h"/>
      </effectiveTime>
      <routeCode code="PO"
       codeSystem="2.16.840.1.113883.5.112"
       codeSystemName="RouteOfAdministration"/>
      <doseQuantity value="1"/>
      <consumable>
        <manufacturedProduct>
          <manufacturedLabeledDrug>
            <code code="317896006"
             codeSystem="2.16.840.1.113883.6.96"
             codeSystemName="SNOMED CT"
             displayName="Digoxin 125micrograms tablet"/>
          </manufacturedLabeledDrug>
        </manufacturedProduct>
      </consumable>
```

```
      <entryRelationship typeCode="COMP">
        <supply classCode="SPLY" moodCode="RQO">
          <repeatNumber>
            <low value="0"/>
            <high value="5"/>
          </repeatNumber>
          <independentInd value="false"/>
          <quantity value="30"/>
        </supply>
      </entryRelationship>
    </substanceAdministration>
  </entry>
</section>
```

5.4.3.7 Участники подразделов entry

Структуры и подразделы АКД-документов могут включать в себя информацию о различных участниках, некоторые из которых указаны также в заголовке АКД-документа. Как указано в описании контекста АКД-документа (см. 5.4.4), участники, наследуемые от заголовка, могут быть переопределены в теле документа.

5.4.3.7.1 Участник author (автор) Описанный выше участник author (см. подраздел 0 «

Участник author (автор)») может быть приписан разделу section, чтобы переопределить значения, наследуемые от заголовка АКД-документа. Он может быть также приписан подразделу entry. В этом случае он переопределяет значения, наследуемые от раздела, и распространяется на вложенные подразделы.

5.4.3.7.2 Участник consumable (расходуемый)

Участник consumable описан выше в подразделе 5.4.3.6 «Типы подразделов entry».

5.4.3.7.3 Участник informant (информатор)

Участник informant описан выше в подразделе 5.4.2.2.6 «Участник informant (информатор)». Он может быть приписан разделу section, и в этом случае он переопределяет значения, наследуемые от заголовка АКД-документа. Он может быть приписан также подразделу entry. Тогда он переопределяет значения, наследуемые от раздела, и распространяется на все вложенные подразделы, пока не будет там переопределен.

5.4.3.7.4 Участник participant (другой участник)

Содержит информацию об ином участнике, который не может быть причислен ни к одному из более специфичных участников. Участник participant может быть приписан подразделу entry, и в этом случае он распространяется на все вложенные подразделы, пока не будет там переопределен.

Таблица 119 — Допустимые значения атрибута participant.typeCode (CNE)

Код	Описание
Любой подтип домена ParticipationType	Допустимые значения составляют домен ParticipationType

Таблица 120 — Допустимые значения атрибута participant.contextControlCode (CNE)

Код	Описание
OP (замещающая, распространяемая) [по умолчанию]	Ассоциация участника, замещающая ассоциацию с тем же значением атрибута typeCode. Эта замещающая ассоциация будет распространяться на действия-потомки, которые связаны с данным экземпляром класса Act с помощью экземпляра класса ActRelationship (см. 5.4.4)

Участником participant является сущность, выполняющая одну из нескольких возможных ролей (ParticipantRole). Сущностью, исполняющей роль, является устройство (класс Bevice) или другая сущность (класс PlayingEntity). Контролером может являться любая сущность (класс Entity).

Таблица 121 — Допустимые значения атрибута ParticipantRole.classCode (CNE)

Код	Описание
Любой подтип типа ROL (из домена RoleClassRoot)	Допустимые значения составляют домен RoleClassRoot

Таблица 122 — Допустимые значения атрибута Device.classCode (CNE)

Код	Описание
DEV (устройство) [по умолчанию]	Сущность, используемая при выполнении действия и существенно не изменяющаяся в процессе его выполнения
Любой подтип типа DEV	Допустимые значения берутся из словарного домена EntityClassDevice

Таблица 123 — Допустимые значения атрибута Device.determinerCode (CNE)

Код	Описание
INSTANCE (экземпляр) [по умолчанию]	Определитель INSTANCE указывает фактический экземпляр сущности, в отличие от определителя KIND, который относится к типу сущностей. Например, можно говорить о конкретном автомобиле (об экземпляре автомобиля) или об автомобилях вообще (о типе транспортного средства)

Таблица 124 — Допустимые значения атрибута PlayingEntity.classCode (CNE)

Код	Описание
ENT (сущность) [по умолчанию]	Физический объект, группа физических объектов или организация, способные выполнять определенную роль в действиях
Любой подтип типа ENT	Допустимые значения берутся из словарного домена EntityClassRoot

Таблица 125 — Допустимые значения атрибута PlayingEntity.determinerCode (CNE)

Код	Описание
INSTANCE (экземпляр) [по умолчанию]	Определитель INSTANCE указывает фактический экземпляр сущности, в отличие от определителя KIND, который относится к типу сущностей. Например, можно говорить о конкретном автомобиле (об экземпляре автомобиля) или об автомобилях вообще (о типе транспортного средства)

Таблица 126 — Допустимые значения атрибута Entity.classCode (CNE)

Код	Описание
ENT (сущность) [по умолчанию]	Физический объект, группа физических объектов или организация, способные выполнять определенную роль в действиях
Любой подтип типа ENT	Допустимые значения берутся из словарного домена EntityClassRoot

Таблица 127 — Допустимые значения Entity Device.determinerCode (CNE)

Код	Описание
INSTANCE (экземпляр) [по умолчанию]	Определитель INSTANCE указывает фактический экземпляр сущности, в отличие от определителя KIND, который относится к типу сущностей. Например, можно говорить о конкретном автомобиле (об экземпляре автомобиля) или об автомобилях вообще (о типе транспортного средства)

5.4.3.7.5 Участник performer (исполнитель)

Исполнителем (performer) является физическое лицо, которое выполняет или выполнит конкретное действие. Исполнитель не обязан быть основным ответственным участником действия; например, хирург-ординатор, оперирующий под надзором ведущего хирурга, является исполнителем.

Таблица 128 — Допустимые значения атрибута performer.typeCode (CNE)

Код	Описание
PRF (исполнитель) [по умолчанию]	Лицо, которое фактически выполняет или выполнит конкретное действие, в том числе традиционный исполнитель заказа

5.4.3.7.6 Участник product (продукт)

Участник product описан выше в подразделе 5.4.3.6 «Типы подразделов entry».

5.4.3.7.7 Участник specimen (образец)

Образцом (specimen) является часть некоторой сущности, обычно субъекта, которая служит предметом лабораторного, лучевого или иного исследования. Во многих клинических исследованиях, например при физикальном осмотре, предметом является пациент, и в этом случае образец отсутствует. Участник specimen используется только в том случае, когда исследуется некоторая субстанция или объект, взятый или произведенный от субъекта.

Таблица 129 — Допустимые значения атрибута specimen.typeCode (CNE)

Код	Описание
SPC (образец) [по умолчанию]	Предмет не клинического (например, лабораторного) исследования

Таблица 130 — Допустимые значения атрибута SpecimenRole.classCode (CNE)

Код	Описание
SPEC (образец) [по умолчанию]	Роль, выполняемая материальной сущностью, являющейся образцом в некотором действии

5.4.3.7.8 Участник subject (субъект)

Участник subject описан выше в подразделе 5.4.3.3.3 «Участник subject (субъект)». Он может быть приписан разделу section, а также подразделу entry. В последнем случае он переопределяет значения, наследуемые от раздела, и распространяется на все вложенные подразделы, пока не будет там переопределен.

5.4.3.8 Отношения между подразделами

5.4.3.8.1 Отношение component (компонент)

Источником отношения component является подраздел entry типа Organizer (см. подраздел 0 «Класс Organizer (группировка)»), а целью — другой подраздел АКД-документа. Это отношение используется для группировки подразделов АКД-документа путем вложения в подраздел типа Organizer.

Таблица 131 — Допустимые значения атрибута component.typeCode (CNE)

Код	Описание
COMP (компонент) [по умолчанию]	Вложенный подраздел является компонентом подраздела типа Organizer

5.4.3.8.2 Отношение precondition (предварительное условие)

Класс precondition является производным от класса ActRelationship (отношение между действиями) и используется вместе с классом Criterion (критерий) для описания условия, которое должно быть удовлетворено до начала некоторой деятельности.

Таблица 132 — Допустимые значения атрибута precondition.typeCode (CNE)

Код	Описание
PRCN (предварительное условие) [по умолчанию]	Условие, которое должно быть удовлетворено до начала предоставления услуги

Таблица 133 — Допустимые значения атрибута Criterion.classCode (CNE)

Код	Описание
OBS (исследование) [по умолчанию]	Исследованиями являются действия, выполняемые для получения ответа или результата
Любой подтип типа OBS	Допустимые значения берутся из словарного домена ActClassObservation

Таблица 134 — Допустимые значения атрибута Criterion.moodCode (CNE)

Код	Описание
EVN.CRT (критерий события) [по умолчанию]	Критерий или условие, применяемые к исследованиям, которые должны быть удовлетворены до выполнения соответствующей услуги

5.4.3.8.3 Отношение referenceRange (справочный диапазон)

У отношения referenceRange (описанного выше в подразделе 0 «Класс Observation (исследование)») источником является подраздел типа Observation, а целью — подраздел типа ObservationRange.

5.4.3.8.4 Отношение entryRelationship (связь между подразделами)

В настоящем стандарте идентифицированы и смоделированы различные сценарии связей и ссылок. Реализация этих сценариев обеспечивает семантическую связь подразделов entry АКД-документа с подразделами, содержащимися в том же самом документе (с помощью отношения entryRelationship), и с объектами, внешними по отношению к этому документу (с помощью отношения reference).

Примечание — Спецификация АКД позволяет установить отношения любого подраздела АКД-документа с любым другим его подразделом, задавая любой из типов отношений, указанных в таблице 135. Во многих случаях такие отношения окажутся лишенными смысла. Таблица 135 может служить руководством по разумным типам отношений между подразделами, но приведенная в ней информация не является нормативным требованием соответствия стандарту.

Таблица 135 — Типы отношений entryRelationship между подразделами entry

Тип отношения	Разумные типы источников и целей отношения	Примечания
CAUS (является этиологией)	[Act | Observation | Procedure | SubstanceAdministration] CAUS [Observation]	Указывает, что источник отношения вызвал целевой результат исследования (например, источник «сахарный диабет» вызвал «заболевание почек»)
COMP (содержит компонент)	[Act | Observation | Procedure | SubstanceAdministration | Supply] COMP [Act | Observation | Procedure | SubstanceAdministration | Supply]	Указывает, что цель отношения является компонентом источника (например, «анализ содержания гемоглобина» является компонентом «общего анализа крови»)
GEVL (оценка соответствия (цели))	[Observation] GEVL [Observation]	Задает связь между исследованием (состоявшимся или планируемым) и поставленной целью, означающую, что исследование предназначено для оценки соответствия цели (например, источник «фактически пройденная дистанция» является оценкой соответствия цели «адекватная пройденная дистанция»)
MFST (проявление)	[Observation] MFST [Observation]	Указывает, что источник является проявлением цели (например, источник «крапивница» является проявлением цели отношения «аллергия на пенициллин»)
REFR (ссылается на)	[Act | Observation | Procedure | SubstanceAdministration | Supply] REFR [Act | Observation | ObservationMedia | Procedure | RegionOfInterest | SubstanceAdministration | Supply]	Указывает общую зависимость между источником и целью отношения, когда более специфичная семантика отношения не известна

Окончание таблицы 135

Тип отношения	Разумные типы источников и целей отношения	Примечания
RSON (имеет причину)	[Act | Encounter | Observation | Procedure | SubstanceAdministration | Supply] RSON [Act | Encounter | Observation | Procedure | SubstanceAdministration | Supply]	Указывает причину или основание выполнения услуги (например, источник «тредмил-тест» обусловлен наличием «боли в груди»)
SAS (начинается после начала)	[Act | Encounter | Observation | Procedure | SubstanceAdministration | Supply] SAS [Act | Encounter | Observation | Procedure | SubstanceAdministration | Supply]	Действие-источник начинается после начала действия-цели отношения (например, источник «пото-отделение» начинается после начала «боли в груди»)
SPRT (подтверждается)	[Observation] SPRT [Observation | ObservationMedia | RegionOfInterest]	Указывает, что источник отношения подтверждается целью отношения (например, гипотеза «подозрение на опухоль легких» подтверждается целью «обнаружена масса на рентгеновском снимке грудной клетки»)
SUBJ (относится к предмету)	[Observation | RegionOfInterest] SUBJ [Observation | ObservationMedia]	Используется для привязки области интереса (источника отношения) к целевому изображению или для привязки результата исследования к его предмету (например, источник «умеренный характер» имеет предметом «боль в груди»). Тип отношения «относится к предмету» (класс ActRelationshipType) похож на тип отношения «субъект» (класс ParticipationType). В подразделах, содержание которых относится к физическим объектам, используется отношение Participation (участие), а в подразделах, которые в основном оперируют другими подразделами, — отношение ActRelationship (связь действий)
XCRPT (выписка из)	[Act | Observation] XCRPT [Act | Observation | Procedure | SubstanceAdministration | Supply]	Указывает, что источник является выпиской из цели отношения (например, источник «показатель гемоглобина = 12» является выпиской из цели отношения «общий анализ крови»). Различие между выпиской и участником informant (информатор) может быть не столь очевидным. Например, при добавлении записи в историю лекарственных назначений врач может получить сведения от информатора или сделать выписку из информации, хранящейся на другом компьютере. Информатором (или источником информации) является лицо, предоставляющее информацию. Информатор указывается в заголовке документа и может быть переопределен в теле. Выписка представляет собой часть документирования некоторого другого действия

Если атрибуту entryRelationship.inversionInd (признак обращения) присвоено значение «true», то это означает, что отношение должно быть обращено, как если бы источник и цель отношения поменялись ролями. Выше был приведен пример отношения «тредмил-тест» (RSON обусловлен наличием «боли в груди»). Будучи обращенным, оно примет вид «боль в груди» (обращение RSON) «тредмил-тест». Обращение отношения полезно в том случае, когда текущий контекст описывает цель отношения, которую надо проследить до ее источника.

Использование атрибута entryRelationship.contextConductionInd (признак распространения контекста) отличается от его обычного использования в других случаях (см. 5.4.4) тем, что во всех других слу-

чаях он имеет фиксированное значение «true», в то время как здесь ему по умолчанию присваивается значение «true», но при ссылке на подраздел того же документа оно может быть изменено на «false», если надо отразить тот факт, что объект, на который сделана ссылка, сохраняет свой исходный контекст.

5.4.3.8.5 Отношение reference (ссылка)

Подразделы АКД-документов могут содержать ссылки на внешние объекты, например изображения и предшествующие документы. Эти внешние объекты не являются частью заверяемого содержания данного документа. Если они обладают достаточно развитой структурой, то лучше сделать на них ссылку, чем дублировать их внутри документа. Для кодирования отдельных элементов описания внешних объектов, на которые необходимо сослаться в повествовательном блоке, можно использовать подраздел entry, содержащий эти элементы вместе со ссылкой.

Каждый объект должен иметь идентификатор и код типа объекта и содержать атрибут Act.text, описанный в модели RIM. Он может использоваться для хранения адреса URL и типа среды MIME этого объекта. Классы ссылок на внешние объекты должны иметь атрибут moodCode с фиксированным значением «EVN».

Класс отношения reference имеет атрибут reference.seperatableInd, который указывает, должен ли источник отношения интерпретироваться независимо от цели отношения. Наличие этого атрибута не может воспрепятствовать лицу или приложению отделить обработку источника от обработки цели, он указывает только пожелания автора и готовность заверить содержание источника, если оно отделено от цели. Обычно в тех случаях, когда атрибут seperatableInd имеет значение «false», в пакет передаваемых данных включается также целевой объект ссылки с тем, чтобы получатель мог обеспечить его визуализацию.

В таблице 136 приведено описание допустимых значений атрибута reference.typeCode. Как и таблица 135 (типы отношений entryRelationship), она служит руководством по разумным типам отношений между подразделами АКД-документа и внешними ссылками, но приведенная в ней информация не является нормативным требованием соответствия стандарту.

Таблица 136 — Типы отношений reference в АКД-документе

Тип отношения	Разумные типы источников и целей отношения	Примечания
ELNK (отождествляет эпизоды)	[Observation] ELNK [ExternalObservation]	Указывает, что источник и цель отношения относятся к одному и тому же эпизоду (например, диагноз «пневмония» может быть связан с записью «пневмония» в листе уточненных диагнозов, чтобы показать, что этот диагноз относится к текущему эпизоду пневмонии)
REFR (ссылается на)	[Act | Observation | Procedure | SubstanceAdministration | Supply] REFR [ExternalAct | ExternalDocument | ExternalObservation | ExternalProcedure]	Указывает общую зависимость между источником и целью отношения, когда более специфичная семантика отношения не известна
RPLC (заменяет)	[Act | Encounter | Observation | ObservationMedia | Organizer | Procedure | SubstanceAdministration | Supply] RPLC [ExternalAct | ExternalDocument | ExternalObservation | ExternalProcedure]	Указывает, что подраздел—источник отношения заменяет документирование целевого внешнего действия
SPRT (подтверждается)	[Observation] SPRT [ExternalDocument | ExternalObservation]	Указывает, что источник отношения подтверждается целью отношения
SUBJ (относится к предмету)	[Observation | RegionOfInterest] SUBJ [ExternalObservation]	Используется для привязки области интереса (источника отношения) к целевому изображению или для привязки результата исследования к его предмету
XCRPT (выписка из)	[Act | Observation] XCRPT [ExternalAct | ExternalDocument | ExternalObservation | ExternalProcedure]	Указывает, что источник является выпиской из цели отношения (например, источник «показатель гемоглобина = 10,7» является выпиской из цели отношения «общий анализ крови»)

Целевыми классами отношения ссылки reference могут быть ExternalAct (внешнее действие), ExternalDocument (внешний документ), ExternalObservation (внешний результат исследования) и External Procedure (внешняя процедура).

Класс ExternalAct является производным от класса Act модели RIM и предназначен для использования в тех случаях, когда другие более специфичные классы оказываются неподходящими.

Таблица 137 — Допустимые значения атрибута ExternalAct.classCode (CNE)

Код	Описание
ACT (действие) [по умолчанию]	Медицинская помощь
Любой подтип типа ACT	Допустимые значения берутся из словарного домена ActClassRoot

Таблица 138 — Допустимые значения атрибута ExternalAct.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие

Класс ExternalDocument является производным от класса Document модели RIM и предназначен для представления внешних документов. Атрибут ExternalDocument.text должен иметь значение с типом данных ED, позволяющее указать тип среды MIME внешнего документа.

Таблица 139 — Допустимые значения атрибута ExternalDocument.classCode (CNE)

Код	Описание
DOC (документ)[по умолчанию]	Понятие документа навеяно передачей информации на бумажных носителях, где ему соответствует содержание, записанное на отдельных листах бумаги. В электронном мире документ представляет собой информационный объект, структура которого напоминает структуру бумажного документа. Обычно электронные документы предполагаются человеко-читаемыми. Понятие документа в стандартах HL7 отличается от того, что определено в спецификации XML консорциума W3C, согласно которой документом считается содержание между начальным и конечным тегом корневого элемента. Не все XML-документы являются документами HL7
Любой подтип типа DOC	Допустимые значения берутся из словарного домена ActClassDocument

Таблица 140 — Допустимые значения атрибута ExternalDocument.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие

Класс ExternalObservation является производным от класса Observation модели RIM и предназначен для представления внешних кодируемых и других результатов исследований.

Таблица 141 — Допустимые значения атрибута ExternalObservation.classCode (CNE)

Код	Описание
OBS (исследование) [по умолчанию]	Исследованиями являются действия, выполняемые для получения ответа или результата
Любой подтип типа OBS	Допустимые значения берутся из словарного домена ActClassObservation

Таблица 142 — Допустимые значения атрибута ExternalObservation.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие

Класс ExternalProcedure является производным от класса Procedure модели RIM и предназначен для представления внешних процедур.

Таблица 143 — Допустимые значения атрибута ExternalProcedure.classCode (CNE)

Код	Описание
PROC (процедура) [по умолчанию]	Непосредственным и основным результатом процедуры (постусловие) является измененное физическое состояние субъекта

Таблица 144 — Допустимые значения атрибута ExternalProcedure.moodCode (CNE)

Код	Описание
EVN (событие) [по умолчанию]	Фактически произошедшее событие

5.4.4 Контекст АКД-документа

Контекст АКД-документа задается в его заголовке и распространяется на весь документ. Контекст может быть переопределен на уровне тела, раздела section или подраздела entry.

5.4.4.1 Общие сведения о контексте АКД-документа

В определенном смысле документ является контекстуальной оберткой его содержания. Атрибуты, приведенные в заголовке документа, обычно применимы к утверждениям, содержащимся в теле документа, если только не переопределены. Например, пациент, идентифицированный в заголовке документа, предполагается субъектом исследований, описанных в теле документа, если не будет явно задан другой субъект. Автор, идентифицированный в заголовке документа, предполагается автором всего документа, но в разделе section может быть указан другой автор. Правила контекста АКД-документа разработаны для формального представления подобной практики через связь с моделью RIM, чтобы компьютер воспринимал контекст части документа таким же образом, как этот контекст интерпретируется человеком.

В то же время нет никаких гарантий, что машинная обработка будет в состоянии идентифицировать ошибки применения правил контекста. Если врач сделает в повествовательном блоке утверждение «внешний диагноз», но при этом не обнулит атрибут контекста informant (информатор), машинная обработка не выявит переключение атрибутов. Этот особый случай иллюстрирует ограничения, присущие машинной проверке записей электронной медицинской карты, и существует вне зависимости от механизма наследования контекста. Другими словами, некоторые ошибки кодирования данных могут быть выявлены только человеком.

Подход к моделированию контекста, принятый в настоящем стандарте, и к распространению контекста на вложенные компоненты документа опирается на следующие принципы разработки:

1) в АКД используется механизм контекста, предложенный в модели RIM (управляется атрибутом contextControlCode классов, произведенных от класса Participation, и атрибутом contextConductionInd классов, произведенных от класса ActRelationship). Для достижения приведенных ниже целей разработки этим атрибутам присваиваются фиксированные значения, ограничивающие модель контекста, описанную в модели RIM. Для определенных атрибутов заголовка АКД-документа расширено свойство распространения контекста, которое также распространяется через отношения ActRelationship, у которых атрибут contextConductionInd имеет значение «true»;

2) заголовок АКД-документа задает контекст для всего документа. Значения признака распространения, указанные в заголовке документа, сохраняются во всем документе, если только явно не переопределяются. Этот принцип применяется как к отношениям участия (класс Participation), так и к определенным атрибутам заголовка АКД-документа. Контекстуальные компоненты заголовка (то есть те, чьи значения распространяются на компоненты тела) включают в себя следующие атрибуты:

- author (автор),
- confidentialityCode (код конфиденциальности),
- dataEnterer (оператор по вводу данных),
- languageCode (код языка),
- informant (информатор),
- legalAuthenticator (юридически ответственное лицо),
- participant (другой участник),
- recordTarget (целевая медицинская карта);

3) на уровне тела документа могут быть переопределены следующие атрибуты:
- confidentialityCode (код конфиденциальности),
- languageCode (код языка);

4) на уровне раздела документа section могут быть переопределены следующие атрибуты:
- author (автор),
- confidentialityCode (код конфиденциальности),
- languageCode (код языка),
- informant (информатор),
- subject (субъект);

5) на уровне подраздела АКД-документа entry могут быть переопределены следующие атрибуты:
- author (автор),
- languageCode (код языка),
- informant (информатор),
- participant (другой участник),
- subject (субъект);

6) контекст распространяется от внешних тегов к вложенным тегам. Контекст, специфичный для внешнего тега, распространяется на все вложенные теги, если только не переопределен во вложенном теге. Контекст, заданный для тега в теле АКД-документа, всегда переопределяет контекст, наследуемый от внешнего тега. Например, указание автора на уровне раздела документа section переопределяет всех авторов, наследуемых от внешнего тега;

7) контекст иногда известен точно, а иногда неизвестен, как, например, в том случае, когда документ состоит из большого не структурированного повествовательного блока, который потенциально может содержать утверждения, противоречащие внешнему контексту. Поскольку в АКД-документ контекст всегда распространяется, пока не переопределен, то для указания неопределенности контекста его атрибутам надо присвоить пустые значения.

5.4.4.2 Технические аспекты контекста АКД-документа

В модели RIM контекст действия определяется как совокупность участников этого действия, которая может распространяться на вложенные действия. Согласно модели RIM, возможность распространения участников контекста на вложенные действия зависит от того, разрешает ли отношение между родительским и дочерним действием распространение контекста. Явное представление контекста и его возможности распространения на вложенные действия выражается в модели RIM через атрибуты Participation.contextControlCode и ActRelationship.contextConductionInd. В настоящем стандарте общий механизм контекста, принятый в модели RIM, ограничивается таким образом, что контекст всегда переопределяется и распространяется (таблица 145).

Таблица 145 — Ограничения атрибутов контекста RIM, принятые в АКД

Атрибут модели RIM	Кратность	Соответствие	Фиксированное значение
Participation. contextControlCode	1..1	Обязателен (пустые значения не допускаются)	«OP» (замещающая, распространяемая)
ActRelationship. contextConductionInd	1..1	Обязателен (пустые значения не допускаются)	«true»[5)]

Если контекст вложенного компонента не известен, то распространяемый контекст должен быть переопределен пустыми значениями в соответствии с таблицей 146.

[5)] Единственное исключение из этого правила состоит в том, что атрибуту entryRelationship. contextConductionInd, который по умолчанию имеет значение «true», может быть присвоено значение «false» (см. подраздел 5.4.3.8.4 «Отношение entryRelationship (связь между подразделами)».

Таблица 146 — Блокировка распространения контекста пустыми значениями

Атрибут контекста	Представление пустого значения
author	AssignedAuthor.id = NULL; исполняющая сущность не задана, контролирующая сущность не задана
confidentialityCode	confidentialityCode = NULL
languageCode	languageCode = NULL
informant	AssignedEntity.id = NULL; исполняющая сущность не задана, контролирующая сущность не задана
participant	ParticipantRole.id = NULL; исполняющая сущность не задана, контролирующая сущность не задана

Модель контекста АКД-документа иллюстрируется следующим примером. Для экземпляра класса ClinicalDocument заданы атрибуты контекста author (автор), confidentialityCode (код конфиденциальности) и languageCode (код языка), которые будут распространяться на вложенные действия. Атрибут contextConductionInd отношения ActRelationship, связывающего класс ClinicalDocument с классом bodyChoice, имеет фиксированное значение «true», разрешающее распространение контекста. Подклассы NonXMLBody и StructuredBody класса bodyChoice имеют атрибуты confidentialityCode и languageCode, которые могут использоваться для переопределения значений, указанных в заголовке. Атрибут contextConductionInd отношения component, связывающего класс StructuredBody с классом Section, имеет фиксированное значение «true», разрешающее контексту класса StructuredBody распространиться на класс Section. Этот класс имеет атрибуты confidentialityCode, languageCode и author, которые могут быть переопределены. Пустое значение, присвоенное атрибуту author, означает, что автор данного конкретного раздела не известен.

Модель примера контекста показана на рисунке 2.

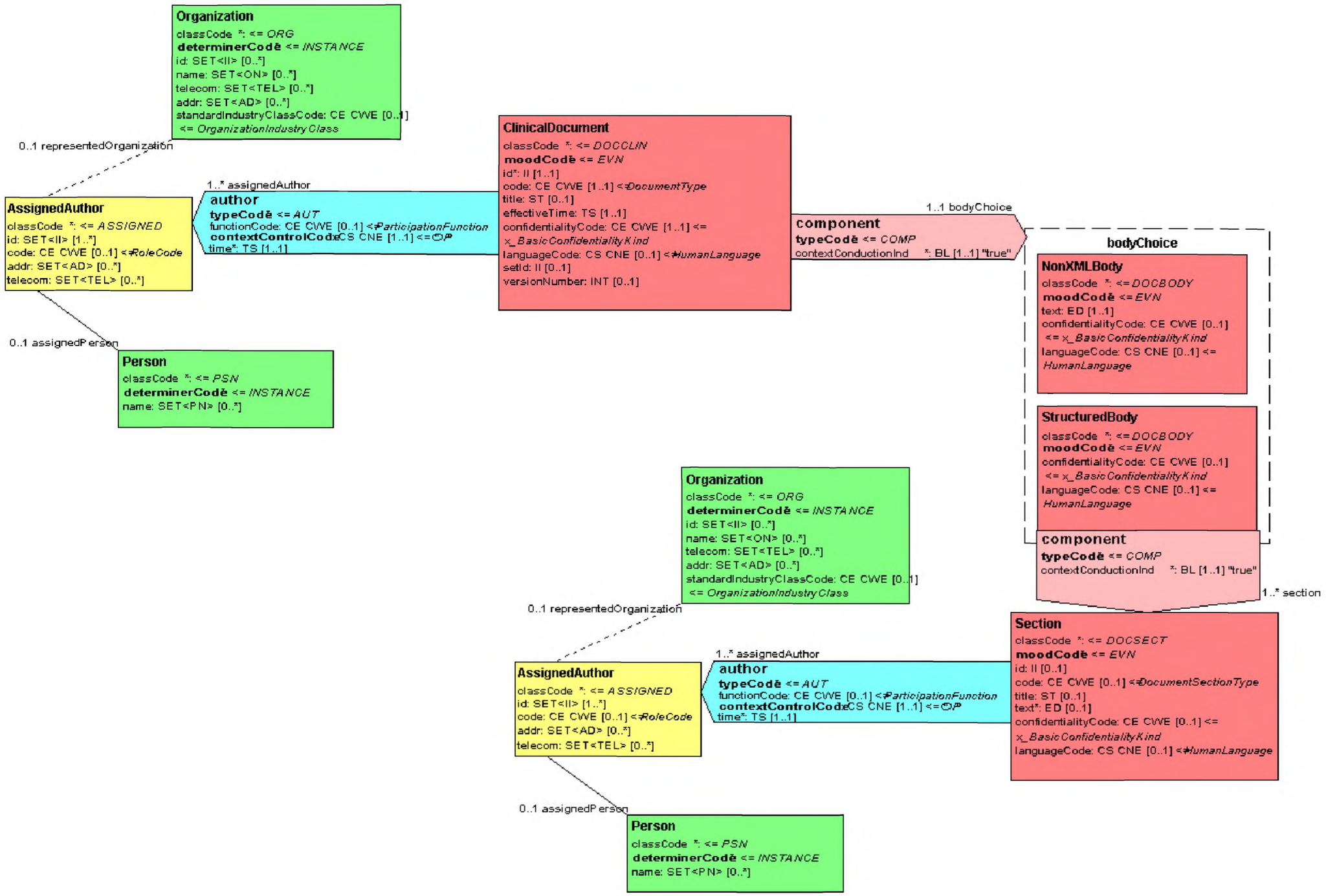


Рисунок 2 — Модель примера контекста АКД-документа

Поскольку контекст всегда переопределяется и распространяется, можно вычислить контекст данного узла XML-представления АКД-документа, найдя ближайшее к нему вхождение соответствующего элемента. В следующем примере показано выражение на языке XPath, которое может использоваться для идентификации автора в контексте раздела section или подраздела entry.

Пример 12 —

(ancestor-or-self::*/author)[position( )=last( )]

5.5 Иерархическое описание структуры АКД-документов

Примечание — Наиболее полное описание процесса разработки и интерпретации иерархических описаний в стандартах HL7 можно найти в документе HL7 V3 Guide.

Иерархическое описание структуры АКД-документов приведено в 6.1, таблица 151.

Иерархическое описание АКД является наиболее полным источником правил соответствия стандарту АКД. Оно используется при конструировании XML-схемы АКД-документов. Экземпляр АКД-документа должен не только соответствовать XML-схеме, но еще и удовлетворять правилам соответствия, объявленным в иерархическом описании АКД, а также правилам, описанным в тексте настоящей спецификации.

В настоящем стандарте спецификация соответствия предусматривается на уровне каждого атрибута модели RIM. Эти атрибуты могут быть объявлены «требуемыми», то есть им должны присваиваться значения, если они известны, даже в том случае, если кратность значений атрибута допускает его отсутствие, и могут быть объявлены «обязательными», то есть им во всех случаях должны присваиваться не пустые значения.

Во втором выпуске АКД признаки соответствия «требуемый» и «обязательный» применяются следующим образом:

а) требуемые атрибуты:
   1) Section.text,
   2) все атрибуты с ненулевым нижним пределом кратности;

б) обязательные атрибуты:
   1) ClinicalDocument.typeId,
   2) Структурные атрибуты модели RIM:
      - ClassCode,
      - MoodCode,
      - TypeCode,
      - DeterminerCode;
   3) атрибуты контекста:
      - contextControlCode,
      - ContextConductionInd.

Примечание — В тех случаях, когда обязательные атрибуты имеют значение по умолчанию или фиксированное значение, указанное в иерархическом описании АКД, экземплярам этих атрибутов значение может не присваиваться. Получатель документа должен подразумевать значение по умолчанию.

5.6 Реализация АКД на языке XML

Примечание — Наиболее полное описание технологии реализации стандартов HL7 на языке XML приведено в документе HL7 XML Implementation Technology Specification (XML ITS).

XML-схема АКД-документа cda.xsd приведена в приложении F.

XML-схема Datatypes.xsd приведена в припожении F.

XML-схема Datatypes-base.xsd приведена в приложении F.

XML-схема POCD_MT000040.xsd приведена в приложении F.

XML-схема повествовательного блока NarrativeBlock.xsd приведена в приложении F.

XML-схема voc.xsd приведена в приложении F.

Сама по себе XML-схема АКД не является нормативной. Проверка на соответствие этой схеме является суррогатом проверки соответствия нормативной спецификации XML ITS.

XML-схема повествовательного блока, представляющая собой модель содержания атрибута section.text на языке XML, разработана вручную, как описано в подразделе 0 «Повествовательный

блок раздела». В отличие от XML-схемы АКД-документа cda.xsd, XML-схема повествовательного блока NarrativeBlock.xsd является нормативной.

5.7 Примеры

5.7.1 Пример документа

Заключение консультанта клиники Good Health Clinic
Консультант: Роберт Долин, д.м.н.
Дата: апрель 7, 2000
Пациент: Генри Левин, 7-й; М/карта: 12345; Пол: Муж
Дата рождения: сентябрь 24, 1932

История текущего заболевания

Генри Левин, 7-й, 67 лет, обратился за дальнейшим лечением от астмы. Заболел, когда был подростком. В прошлом году был дважды госпитализирован и уже дважды в этом году. Последние несколько месяцев не может обходиться без стероидов.

История заболеваний

Астма
Гипертензия (детали см. в документе HTN.cda)
Остеоартрит правого колена

Медикаментозная терапия

Теодур, 200 мг два раза в день
Альбутерол ингалятор, 2 вдоха, 4 раза в день, по необходимости
Преднизон, 20 мг в день
Гидрохлортиазид, 25 мг в день

Аллергии и непереносимость

Пенициллин — крапивница
Аспирин — одышка
Кодеин — зуд и тошнота

Семейный анамнез

Отец умер от инфаркта миокарда чуть старше 50 лет.
Рака или диабета не было.

Социальный анамнез

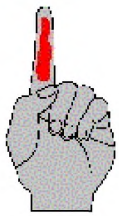
Курение: 1 пачка в день с 20 до 55 лет, затем бросил.
Алкоголь: изредка

Физикальный осмотр

Жизненно важные показатели

Дата / время	**7 апреля 2000 14:30**	**7 апреля 2000 15:30**
Рост	177 см	
Вес	88,0 кг	
ИМТ	28,1 кг/м^2	
ППТ	2.05 м^2	
Температура	36.9 по Цельсию	36.9 по Цельсию
Пульс	86 / мин	84 / мин
Ритм	Регулярный	Регулярный
Дыхание	16 / мин, незатрудненное	14 / мин
Систолическое АД	132 mmHg	135 mmHg
Диастолическое АД	86 mmHg	88 mmHg
Положение / манжета	Левая рука	Левая рука

Осмотр кожи: эритемная сыпь, поверхность ладони, левый указательный палец.



Легкие: чистые, без одышки; дыхание хорошее.

Аускультация сердца: нормальный ритм без шумов, нет S3, нет S4.

Лабораторные и диагностические исследования

Рентген грудной клетки 02/03/1999: легкие растянуты. Нормальные очертания сердца, легкие чистые.

Пикфлоу сегодня: 260 л/мин

Выполненные процедуры

Снятие швов на левом предплечье.

Заключение

Астма, с предшествующей историей курения. С трудом отвыкает от стероидов. Попробует постепенный отказ.

Гипертензия, хорошо контролируемая.

Контактный дерматит пальца.

Рекомендации

Полное исследование ФВД с определением объема легких.

Панель Chem-7 завтра.

Обучение измерению пикфлоу.

Уменьшить преднизон до 20 мг через день, чередуя с 18 мг через день.

Гидрокортизон, крем для пальца, 2 раза в день.

Повторный визит через неделю.

Подпись: Роберт Долин, д. м. н. Дата: апрель 8, 2000

5.7.2 Пример АКД-документа

Приведен корректный пример АКД-документа, соответствующий настоящему стандарту.

Примечание — Рекомендуется иметь в виду руководство по реализации «Using SNOMED CT in HL7 Version 3», которое пока что находится на стадии разработки. Это руководство, разрабатываемое совместно комитетом HL7 и Институтом американских патологоанатомов (College of American Pathologists), будет представлено на утверждение в комитет HL7 как справочный документ. Шаблоны использования SNOMED CT, приведенные в настоящем примере, будут переработаны в соответствии с рекомендациями окончательной версии этого руководства.

5.7.3 Пример XSLT-преобразования для визуализации АКД-документа

В подразделе приложения приведен пример XSLT-преобразования АКД-документа в формат HTML. Его можно использовать как отправную точку для местной разработки, аналогичной преобразованию. Ему присущи некоторые известные ограничения, в том числе:

- при местной разработке могут предъявляться другие требования к отображению заголовка АКД-документа;
- не поддерживается отображение области интереса;
- не поддерживается отображение вложенных мультимедийных данных (например, закодированных в формате Base 64 внутри АКД-документа);
- не поддерживается отображение удаленного фрагмента текста повествовательного блока.

5.8 Указания по реализации

5.8.1 Создание АКД-документов

Введение

Для создания АКД-документов используется все более возрастающее разнообразие средств и методов:

- операторский ввод: большинство клинических документов создаются с помощью речевого интерфейса. В настоящее время и крупные, и малые производители средств операторского ввода обеспечивают возможность передачи введенных документов в формате АКД. Некоторые из них обеспечивают встроенные возможности анализа текста на естественном языке, позволяющие создавать кодированные структуры внутри АКД-документов, являющихся результатом обработки диктовки;

- системы ведения электронных медицинских карт: многие производители систем ведения электронных медицинских карт обеспечивают экспорт документов в формате АКД. Обычно это не является стандартной функцией и делается по запросу. Для этих систем формат АКД представляется относительно простым форматом вывода;

- XML-формы: новое поколение средств конструирования XML-форм позволяет создавать формы ввода, при заполнении которых создаются АКД-документы;

- база знаний: по крайней мере один из основных поставщиков медицинской помощи США разработал редактор АКД-документов поверх базы знаний, предназначенный для структурированного ввода с доступом к базе;

- динамический запрос: в некоторых распределенных приложениях используется динамическое создание АКД-документов для предварительного заполнения из существующих хранилищ данных, например из баз данных результатов лабораторных анализов. Этот метод может использоваться в сочетании с другими методами.

В данном приложении конкретные средства и методы не рассматриваются, оно служит общим руководством по использованию АКД при создании документов.

Прежде чем начать: соответствие модели RIM:

- структуры, словарь, типы данных.

Создание экземпляра документа, соответствующего спецификации АКД, означает, что структура содержащейся в нем информации определена моделью RIM. Независимо от исходной точки разработки или метода генерации документов, вычислительная семантика полученного документа должна быть производной от отношений между классами, определенными в модели RIM, контролируемого словаря и типов данных модели V3 RIM. Любая реализация генератора АКД-документов должна начинаться с проверки того, как требования к документам соотносятся с моделью RIM, типами данных и словарем.

Модель RIM, однако, является очень абстрактной и позволяет использовать много обширных словарных доменов. Хотя возможность отображения на модель RIM является необходимым условием генерации АКД-документов, для создания документов недостаточно выбрать метод генерации или вид пользовательского интерфейса.

АКД является спецификацией обмена документами, а не конструирования документов:

- спецификация АКД не является определяющей для создания документов.

АКД является спецификацией формы обмена клиническими документами. В XML-схему АКД-документов заложены многие из требований соответствия спецификации АКД, но для большинства конструкторов документов эта схема оказывается слишком общей. Вообще говоря, стандарты интероперабельности и сетевого обмена не могут оказывать непосредственное влияние на графический интерфейс пользователя, предназначенный для конструирования документов. В конкретной области применения АКД — документирование медицинской помощи — общие требования к обмену данными пересекаются, но не совпадают с требованиями к такому интерфейсу пользователя.

Например, для удовлетворения требования АКД к человекочитаемости необходимо создать единую таблицу стилей, с помощью которой можно отобразить заверяемое клиническое содержание любого АКД-документа. Если бы элементы общей схемы АКД-документов были определены соответствующими разделами документа, к примеру, <historyOfPresentIllness> (история текущего заболевания) или <Subjective> (жалобы), то в таблицу стилей мог бы быть включен код, распознающий их как теги уровня раздела и обеспечивающий их адекватный вывод. Но подход АКД, при котором разделам соответствуют элементы section, а типы разделов задаются с помощью кодированного слова, означает, что расширяемость АКД обеспечивается не только за счет применения внешних словарных доменов, управляемых другими организациями, но и разработчики документов получают гибкие возможности создания иерархий разделов и присвоения им имен и тегов в соответствии с местными требованиями, которые позволяют сохранить совместимость в контексте обмена. Таким образом, использование специфичных имен тегов облегчает местную разработку графических интерфейсов пользователя, но там, где практика накладывает более жесткие требования, для АКД требовалось использовать более общий подход.

Необходимо учитывать оба комплекса требований — для конструирования документов и для их передачи. При определенной общности интересов, например в единой корпорации, в профессиональ-

ных ассоциациях и в некоторых случаях в местных и региональных органах управления здравоохранением, может быть достигнуто твердое соглашение о форме документа, и определения структуры документов для их конструирования и передачи совпадут. Учитывая разнообразие местных требований, можно утверждать, что универсального обмена документами не будет до тех пор, пока не будет выработано универсальное соглашение. Скучно повторять, что АКД останется общим стандартом обмена документами, и должны быть предложены другие подходы к разработке требований по проверке ввода данных и создания документов.

Общие подходы: ограничение или преобразование:

- ограничение: получение правильного АКД-документа непосредственно из системы конструирования документов с использованием схемы, отличающейся от схемы АКД-документов;
- преобразование: использование местной XML-схемы, преобразование в АКД-документ.

Первый способ состоит в наложении ограничений на схему АКД-документов, в результате которого будет получена спецификация конкретного типа документа (см. приведенное ниже обсуждение создания АКД-документов с использованием местной схемы). Существует несколько способов наложения ограничений. Один из них состоит в модификации самой схемы АКД-документов, превращающей ее в местный вариант (см. local.cda.xsd на рисунке 3). Модификации могут включать в себя ограничение уровней вложения, ограничение словарей данных и последовательности компонентов; например, можно потребовать, чтобы раздел section с кодом LOINC, соответствующим «жалобам», был первым в теле документа, а за ним следовал раздел section с кодом LOINC, соответствующим «результатам осмотра». Такие модификации могут быть отражены в самой XML-схеме или с помощью выражений XPath в местной схеме. Экземпляры документов, которые будут соответствовать этой ограниченной, местной версии схемы АКД-документов, будут, по определению, также и правильными АКД-документами.

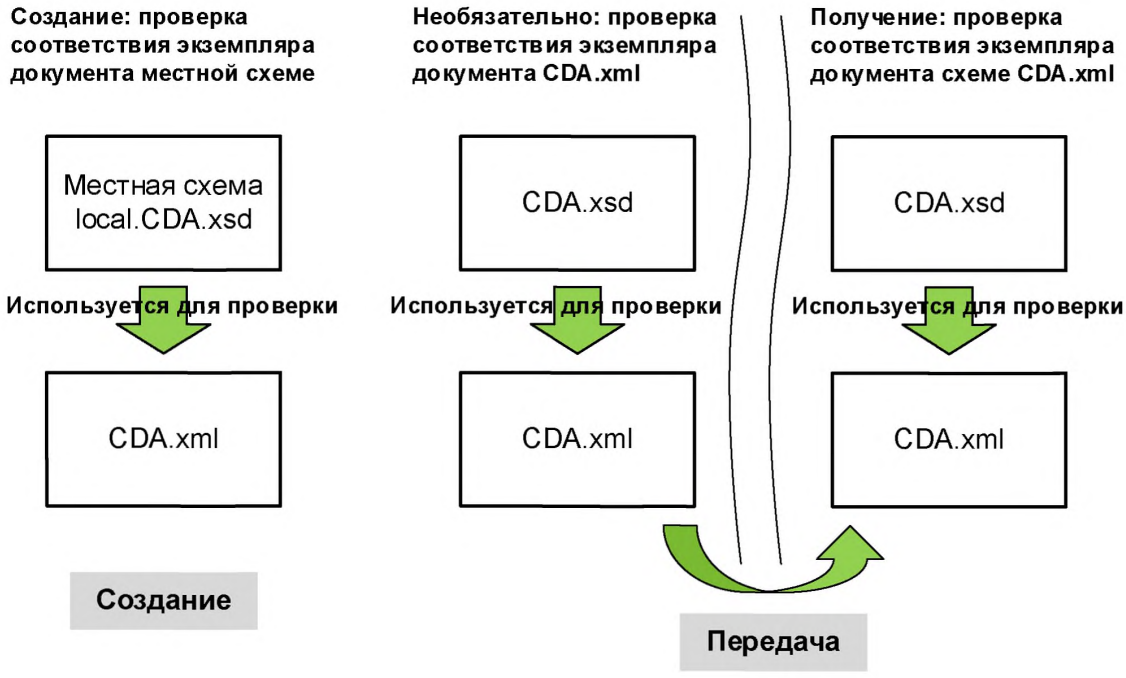


Рисунок 3 — Использование ограниченной местной схемы

Одним их типов ограничений схемы являются шаблоны. В настоящее время комитет HL7 работает над созданием формального способа описания шаблонов (см. 5.1.2.2).

Второй подход состоит в создании местной схемы и преобразовании местного экземпляра XML-документа в АКД-документ (см. рисунок 4).

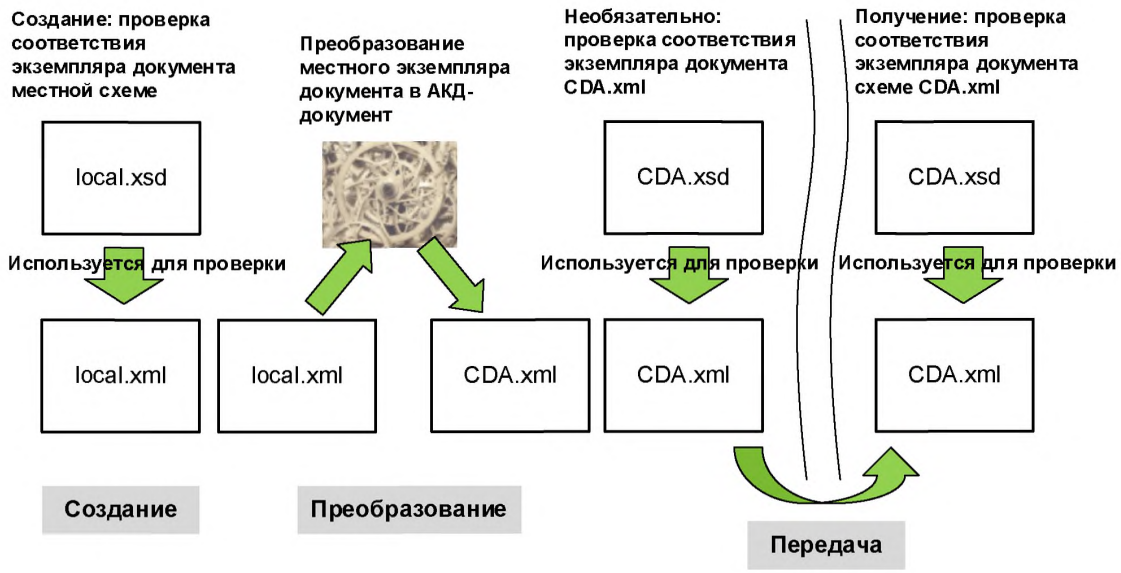


Рисунок 4 — Использование местной схемы и местного экземпляра XML-документа

5.8.2 Коды документов в классификации LOINC

Содержание таблицы 147 позаимствовано из версии 2.14 классификации LOINC (декабрь 2004 года). Оно представляет собой подмножество, состоящее из строк, у которых атрибут SCALE (шкала) имеет значение «DOC» (и статус не равен «DEL»). Модель документа в этой классификации включает в себя компоненты «тип услуги» (столбец COMPONENT), «условия медицинской помощи» (столбец SYSTEM), «предметная область» (столбец METHOD_TYP) и «профессиональный уровень» (также в столбце METHOD_TYP).

Тип услуги характеризует вид услуги или помощи, предоставленной пациенту или для пациента (либо иного субъекта услуги) и описанной в документе. К общим подклассам услуг можно отнести обследование, оценку состояния и управление лечением. Понятия временной последовательности, например начало услуги (поступление) или ее окончание (выписка), также включены в эту ось классификации.

Совокупность кодов условий медицинской помощи представляет собой умеренное расширение грубой классификации, используемой организацией CMS (Centers for Medicare and Medicaid Services — центры услуг Medicare и Medicaid). Понятие условий оказания медицинской помощи не эквивалентно месту ее оказания, смысл которого обычно более вольно трактуется местными правилами.

Предметная область характеризует предмет документа или категорию клинической специализации медицинской помощи, описанной в документе. Профессиональный уровень характеризует уровень образования или подготовки автора документа.

Таблица 147 — Коды документов в классификации LOINC

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
34862-3	Оценка состояния при поступлении (ADMISSION EVALUATION NOTE)	Стационар (INPATIENT)	Лечащий врач — врач общей практики (ATTENDING PHYSICIAN.GENERAL MEDICINE)
34744-3	Оценка состояния при поступлении (ADMISSION EVALUATION NOTE)	Различные ({SETTING})	Сестринское дело (NURSING)

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
34873-0	Оценка состояния при поступлении (ADMISSION EVALUATION NOTE)	Различные ({SETTING})	Хирургия (SURGERY)
34094-3	Анамнез и физикальное обследование при госпитализации (осмотр при поступлении) (ADMISSION HISTORY AND PHYSICAL NOTE)	Стационар (HOSPITAL)	Кардиология (CARDIOLOGY)
34763-3	Анамнез и физикальное обследование при госпитализации (осмотр при поступлении) (ADMISSION HISTORY AND PHYSICAL NOTE)	Различные ({SETTING})	Общая медицинская практика (GENERAL MEDICINE)
28615-3	Обследование логопедом (AUDIOLOGY STUDY)	Различные ({SETTING})	
18743-5	Протокол вскрытия (AUTOPSY NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
33720-4	Заключение банка крови (BLOOD BANK CONSULT)	Различные ({SETTING})	
34095-0	Подробный анамнез и физикальное обследование (первичный осмотр) (COMPREHENSIVE HISTORY & PHYSICAL NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34096-8	Подробный анамнез и физикальное обследование (первичный осмотр) (COMPREHENSIVE HISTORY & PHYSICAL NOTE)	Клиника сестринского ухода (NURSING HOME)	
34098-4	Коллегиальная оценка состояния (CONFERENCE EVALUATION NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34097-6	Коллегиальная оценка состояния (CONFERENCE EVALUATION NOTE)	Клиника сестринского ухода (NURSING HOME)	Исполнитель ({PROVIDER})
24611-6	Подтверждающая консультация (CONFIRMATORY CONSULTATION NOTE)	Амбулатория (OUTPATIENT)	Исполнитель ({PROVIDER})
11488-4	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34100-8	Консультация (CONSULTATION NOTE)	Отделение интенсивной терапии (CRITICAL CARE UNIT)	Исполнитель ({PROVIDER})
34104-0	Консультация (CONSULTATION NOTE)	Стационар (HOSPITAL)	Исполнитель ({PROVIDER})
34749-2	Консультация (CONSULTATION NOTE)	Амбулатория (OUTPATIENT)	Анестезия (ANESTHESIA)
34099-2	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Кардиология (CARDIOLOGY)
34756-7	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Стоматология (DENTISTRY)
34758-3	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Дерматология (DERMATOLOGY)
34760-9	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Диабетология (DIABETOLOGY)
34879-7	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Эндокринология (ENDOCRINOLOGY)

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
34761-7	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Гастроэнтерология (GASTROENTEROL OGY)
34764-1	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Общая медицинская практика (GENERAL MEDICINE)
34101-6	Консультация (CONSULTATION NOTE)	Амбулатория (OUTPATIENT)	Общая медицинская практика (GENERAL MEDICINE)
34771-6	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Общая хирургия (GENERAL SURGERY)
34776-5	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Геронтология (GERONTOLOGY)
34777-3	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Гинекология (GYNECOLOGY)
34779-9	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Онкогематология (HEMATOLOGY+ONCOLOGY)
34781-5	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Инфекционное заболевание (INFECTIOUS DISEASE)
34783-1	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Лечебная физкультура (KINESIOTHERAPY)
34785-6	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Психологическое здоровье (MENTAL HEALTH)
34795-5	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Нефрология (NEPHROLOGY)
34797-1	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Нейрология (NEUROLOGY)
34798-9	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Нейрохирургия (NEUROSURGERY)
34800-3	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Питание и диета (NUTRITION+DIETE TICS)
34803-7	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Трудотерапия (OCCUPATIONAL THERAPY)
34855-7	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Трудотерапия (OCCUPATIONAL THERAPY)
34805-2	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Онкология (ONCOLOGY)
34807-8	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Офтальмология (OPHTHALMOLOGY)
34810-2	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Оптометрия (OPTOMETRY)
34812-8	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Челюстно-лицевая хирургия (OROMAXILLOFACI AL SURGERY)
34814-4	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Ортопедия (ORTHOPEDICS)
34816-9	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Отоларингология (OTORHINOLARYNGOLOGY)
34820-1	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Фармакология (PHARMACY)
34822-7	Консультация	Различные	Физиотерапия и

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
	(CONSULTATION NOTE)	({SETTING})	реабилитация (PHYSICAL MEDICINE AND REHABILITATION)
34824-3	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Физиотерапия (PHYSICAL THERAPY)
34826-8	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Пластический хирург (PLASTIC SURGERY)
34828-4	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Подиатрия (PODIATRY)
34788-0	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Психиатрия (PSYCHIATRY)
34102-4	Консультация (CONSULTATION NOTE)	Стационар (HOSPITAL)	Психиатрия (PSYCHIATRY)
34791-4	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Психология (PSYCHOLOGY)
34103-2	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Пульмонология (PULMONARY)
34831-8	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Онкологическая лучевая терапия (RADIATION ONCOLOGY)
34833-4	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Санаторно-курортное лечение (RECREATIONAL THERAPY)
34835-9	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Реабилитация (REHABILITATION)
34837-5	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Дыхательная терапия (RESPIRATORY THERAPY)
34839-1	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Ревматология (RHEUMATOLOGY)
34841-7	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Социальная работа (SOCIAL WORK)
34845-8	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Логопедия (SPEECH THERAPY +AUDIOLOGY)
34847-4	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Хирургия (SURGERY)
34849-0	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Торакальная хирургия (THORACIC SURGERY)
34851-6	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Урология (UROLOGY)
34853-2	Консультация (CONSULTATION NOTE)	Различные ({SETTING})	Сосудистая хирургия (VASCULAR SURGERY)
34864-9	Консультация (COUNSELING NOTE)	Различные ({SETTING})	Психологическое здоровье (MENTAL HEALTH)
34869-8	Консультация (COUNSELING NOTE)	Различные ({SETTING})	Фармакология (PHARMACY)
34865-6	Консультация (COUNSELING NOTE)	Различные ({SETTING})	Психиатрия (PSYCHIATRY)
34866-4	Консультация (COUNSELING NOTE)	Различные ({SETTING})	Психология (PSYCHOLOGY)
34872-2	Консультация (COUNSELING NOTE)	Различные ({SETTING})	Социальная работа (SOCIAL WORK)
28622-9	Оценка состояния при выписке (DISCHARGE ASSESSMENT NOTE)	Различные ({SETTING})	Сестринское дело (NURSING)
28574-2	Выписной эпикриз (DISCHARGE NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
18842-5	Выписной эпикриз (DISCHARGE SUMMARIZATION NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34105-7	Выписной эпикриз (DISCHARGE SUMMARIZATION NOTE)	Стационар (HOSPITAL)	Исполнитель ({PROVIDER})
28655-9	Выписной эпикриз (DISCHARGE SUMMARIZATION NOTE)	Различные ({SETTING})	Лечащий врач (ATTENDING PHYSICIAN)
29761-4	Выписной эпикриз (DISCHARGE SUMMARIZATION NOTE)	Различные ({SETTING})	Стоматология (DENTISTRY)
34745-0	Выписной эпикриз (DISCHARGE SUMMARIZATION NOTE)	Различные ({SETTING})	Сестринское дело (NURSING)
11490-0	Выписной эпикриз (DISCHARGE SUMMARIZATION NOTE)	Различные ({SETTING})	Терапевт (PHYSICIAN)
34106-5	Выписной эпикриз (DISCHARGE SUMMARIZATION NOTE)	Стационар (HOSPITAL)	Терапевт (PHYSICIAN)
34895-3	Запись о санитарном просвещении пациента (EDUCATION NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34897-9	Запись о санитарном просвещении пациента (EDUCATION NOTE)	Различные ({SETTING})	Диабетология (DIABETOLOGY)
34902-7	Запись о санитарном просвещении пациента (EDUCATION NOTE)	Амбулатория (OUTPATIENT)	Геронтология (GERONTOLOGY)
34107-3	Запись о санитарном просвещении пациента (EDUCATION PROCEDURE NOTE)	Лечение на дому (HOME HEALTH)	Исполнитель ({PROVIDER})
34108-1	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT)	Амбулатория (OUTPATIENT)	Исполнитель ({PROVIDER})
34109-9	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34111-5	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Служба скорой помощи (EMERGENCY DEPARTMENT)	Исполнитель ({PROVIDER})
34112-3	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Стационар (INPATIENT)	Исполнитель ({PROVIDER})
34113-1	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Клиника сестринского ухода (NURSING HOME)	Исполнитель ({PROVIDER})
34750-0	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Анестезия (ANESTHESIA)
34856-5	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Антикоагуляционная терапия (ANTICOAGULATIO N)
34769-0	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Лечащий врач — врач общей практики (ATTENDING PHYSICIAN.GENERAL MEDICINE)

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
34773-2	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Лечащий врач — хирург (ATTENDING PHYSICIAN.GENERAL SURGERY)
34752-6	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Кардиология (CARDIOLOGY)
34753-4	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Амбулатория (OUTPATIENT)	Кардиология (CARDIOLOGY)
34754-2	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Интенсивная терапия (CRITICAL CARE)
34757-5	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Стоматология (DENTISTRY)
34759-1	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Дерматология (DERMATOLOGY)
34861-5	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Диабетология (DIABETOLOGY)
34110-7	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Амбулатория (OUTPATIENT)	Диабетология (DIABETOLOGY)
34878-9	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Скорая медицинская помощь (EMERGENCY MEDICINE)
34898-7	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Эндокринология (ENDOCRINOLOGY)
34762-5	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Гастроэнтерология (GASTROENTEROLOGY)
34765-8	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Общая медицинская практика (GENERAL MEDICINE)
34766-6	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Амбулатория (OUTPATIENT)	Общая медицинская практика (GENERAL MEDICINE)
34772-4	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Общая хирургия (GENERAL SURGERY)
34778-1	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Гинекология (GYNECOLOGY)
34780-7	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Онкогематология (HEMATOLOGY+ONCOLOGY)

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
34859-9	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Гиперлипидемия (HYPERLIPIDEMIA)
34860-7	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Артериальная гипертензия (HYPERTENSION)
34782-3	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Инфекционное заболевание (INFECTIOUS DISEASE)
34784-9	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Лечебная физкультура (KINESIOTHERAPY)
34767-4	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Студент медицинского ВУЗа. Общая медицинская практика (MEDICAL STUDENT.GENERAL MEDICINE)
34786-4	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Психическое здоровье (MENTAL HEALTH)
34794-8	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Многодисциплинарный (MULTIDISCIPLINARY)
34796-3	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Нефрология (NEPHROLOGY)
34905-0	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Нейрология (NEUROLOGY)
34799-7	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Нейрохирургия (NEUROSURGERY)
34768-2	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Медсестра общей медицинской практики (NURSE.GENERAL MEDICINE)
34746-8	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различны ({SETTING})	Сестринское дело (NURSING)
34801-1	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Питание и диета (NUTRI-TION+DIETE TICS)
34802-9	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Трудотерапия (OCCUPATIONAL THERAPY)
34804-5	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Трудотерапия (OCCUPATIONAL THERAPY)

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
34806-0	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Онкология (ONCOLOGY)
34808-6	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Офтальмология (OPHTHALMOLOGY)
34811-0	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Оптометрия (OPTOMETRY)
34813-6	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Челюстно-лицевая хирургия (OROMAXILLOFACI AL SURGERY)
34815-1	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Ортопедия (ORTHOPEDICS)
34817-7	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Отоларингология (OTORHINOLARYNGOLOGY)
34858-1	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Введение обезболивания (PAIN MANAGEMENT)
34819-3	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Патология (PATHOLOGY)
34821-9	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Фармакология (PHARMACY)
34823-5	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Физиотерапия и реабилитация (PHYSICAL MEDICINE AND REHABILITATION)
34825-0	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Физиотерапия (PHYSICAL THERAPY)
34827-6	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Пластический хирург (PLASTIC SURGERY)
34829-2	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Подиатрия (PODIATRY)
34789-8	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Психиатрия (PSYCHIATRY)
34792-2	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Психология (PSYCHOLOGY)
34830-0	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Пульмонология (PULMONARY)

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
34832-6	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Онкологическая лучевая терапия (RADIATION ONCOLOGY)
34834-2	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Санаторно-курортное лечение (RECREATIONAL THERAPY)
34836-7	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Реабилитация (REHABILITATION)
34838-3	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Дыхательная терапия (RESPIRATORY THERAPY)
34840-9	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Ревматология (RHEUMATOLOGY)
34842-5	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Социальная работа (SOCIAL WORK)
34846-6	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Логопедия (SPEECH THERAPY-+AUDIOLOGY)
34857-3	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Злоупотребление сильнодействующими и веществами (SUBSTANCE ABUSE)
34848-2	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Хирургия (SURGERY)
34850-8	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Амбулатория (OUTPATIENT)	Торакальная хирургия (THORACIC SURGERY)
34852-4	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Урология (UROLOGY)
34854-0	Оценка состояния и план лечения (EVALUATION AND MANAGEMENT NOTE)	Амбулатория (OUTPATIENT)	Сосудистая хирургия (VASCULAR SURGERY)
34114-9	Консилиум (GROUP COUNSELING NOTE)	Стационар (HOSPITAL)	Исполнитель ({PROVIDER})
34787-2	Консилиум (GROUP COUNSELING NOTE)	Различные ({SETTING})	Психическое здоровье (MENTAL HEALTH)
34790-6	Консилиум (GROUP COUNSELING NOTE)	Различные ({SETTING})	Психиатрия (PSYCHIATRY)
34793-0	Консилиум (GROUP COUNSELING NOTE)		Психология (PSYCHOLOGY)
34843-3	Консилиум (GROUP COUNSELING NOTE)	Различные ({SETTING})	Социальная работа (SOCIAL WORK)

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
11492-6	Анамнез и физикальное обследование (HISTORY & PHYSICAL NOTE)	Стационар (HOSPITAL)	Исполнитель ({PROVIDER})
34774-0	Анамнез и физикальное обследование (HISTORY & PHYSICAL NOTE)	Различные ({SETTING})	Общая хирургия (GENERAL SURGERY)
34115-6	Анамнез и физикальное обследование (HISTORY & PHYSICAL NOTE)	Стационар (HOSPITAL)	Студент медицинского ВУЗа (MEDICAL STUDENT)
28626-0	Анамнез и физикальное обследование (HISTORY & PHYSICAL NOTE)	Различные ({SETTING})	Терапевт (PHYSICIAN)
34116-4	Анамнез и физикальное обследование (HISTORY & PHYSICAL NOTE)	Клиника сестринского ухода (NURSING HOME)	Терапевт (PHYSICIAN)
34117-2	Анамнез и физикальное обследование (HISTORY AND PHYSICAL NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
18841-7	Консультация в стационаре (HOSPITAL CONSULTATIONS)	Различные ({SETTING})	
28636-9	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34118-0	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Лечение на дому (HOME HEALTH)	Исполнитель ({PROVIDER})
34119-8	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Клиника сестринского ухода (NURSING HOME)	Исполнитель ({PROVIDER})
34120-6	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Амбулатория (OUTPATIENT)	Исполнитель ({PROVIDER})
28654-2	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Лечащий врач (ATTENDING PHYSICIAN)
28581-7	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Мануальный терапевт (CHIROPRACTOR)
18763-3	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Врач-консультант (CONSULTING PHYSICIAN)
28572-6	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Стоматология (DENTISTRY)
28621-1	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Фельдшер (NURSE PRACTITIONER)
29753-1	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Сестринское дело (NURSING)
18734-4	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Трудотерапия (OCCUPATIONAL THERAPY)
18735-1	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Физиотерапия (PHYSICAL THERAPY)
18736-9	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Терапевт (PHYSICIAN)
18737-7	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Подиатрия (PODIATRY)

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
28635-1	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Психиатрия (PSYCHIATRY)
18738-5	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Психология (PSYCHOLOGY)
18739-3	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Социальная служба (SOCIAL SERVICE)
18740-1	Первичная оценка состояния (INITIAL EVALUATION NOTE)	Различные ({SETTING})	Логопедия (SPEECH THERAPY)
34121-4	Протокол вмешательства (INTERVENTIONAL PROCEDURE NOTE)	Различные ({SETTING})	
34896-1	Протокол вмешательства (INTERVENTIONAL PROCEDURE NOTE)	Различные ({SETTING})	Кардиология (CARDIOLOGY)
34899-5	Протокол вмешательства (INTERVENTIONAL PROCEDURE NOTE)	Различные ({SETTING})	Гастроэнтерология (GASTROENTEROLOGY)
34903-5	Запись (NOTE)	Различные ({SETTING})	Психическое здоровье (MENTAL HEALTH)
34906-8	Запись (NOTE)	Различные ({SETTING})	Духовная помощь (PASTORAL CARE)
11536-0	Записи (NOTES)	Различные ({SETTING})	Сестринское дело (NURSING)
34868-0	Протокол операции (OPERATIVE NOTE)	Различные ({SETTING})	Ортопедия (ORTHOPEDICS)
34818-5	Протокол операции (OPERATIVE NOTE)	Различные ({SETTING})	Отоларингология (OTORHINOLARYNGOLOGY)
34870-6	Протокол операции (OPERATIVE NOTE)	Различные ({SETTING})	Пластический хирург (PLASTIC SURGERY)
34871-4	Протокол операции (OPERATIVE NOTE)	Различные ({SETTING})	Подиатрия (PODIATRY)
34874-8	Протокол операции (OPERATIVE NOTE)	Различные ({SETTING})	Хирургия (SURGERY)
34877-1	Протокол операции (OPERATIVE NOTE)	Различные ({SETTING})	Урология (UROLOGY)
34122-2	Протокол диагностического исследования (PATHOLOGY PROCEDURE NOTE)	Различные ({SETTING})	Диагностика (PATHOLOGY)
34863-1	Постоперационная оценка состояния и план лечения (Постоперационный эпикриз) (POST-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Общая хирургия (GENERAL SURGERY)

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
34880-5	Постоперационная оценка состояния и план лечения (Постоперационный эпикриз) (POST-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Операционная сестра (NURSE.SURGERY)
34867-2	Постоперационная оценка состояния и план лечения (Постоперационный эпикриз) (POST-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Амбулатория (OUTPATIENT)	Офтальмология (OPHTHALMOLOGY)
34875-5	Постоперационная оценка состояния и план лечения (Постоперационный эпикриз) (POST-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Хирургия (SURGERY)
34751-8	Предоперационная оценка состояния и план лечения (предоперационный эпикриз) (PRE-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Анестезия (ANESTHESIA)
34123-0	Предоперационная оценка состояния и план лечения (предоперационный эпикриз) (PRE-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Стационар (HOSPITAL)	Анестезия (ANESTHESIA)
34775-7	Предоперационная оценка состояния и план лечения (предоперационный эпикриз) (PRE-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Общая хирургия (GENERAL SURGERY)
34881-3	Предоперационная оценка состояния и план лечения (предоперационный эпикриз) (PRE-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Операционная сестра (NURSE.SURGERY)
34747-6	Предоперационная оценка состояния и план лечения (предоперационный эпикриз) (PRE-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Сестринское дело (NURSING)
34809-4	Предоперационная оценка состояния и план лечения (предоперационный эпикриз) (PRE-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Офтальмология (OPHTHALMOLOGY)
34876-3	Предоперационная оценка состояния и план лечения (предоперационный эпикриз) (PRE-OPERATIVE EVALUATION AND MANAGEMENT NOTE)	Различные ({SETTING})	Хирургия (SURGERY)
28570-0	Протокол (PROCEDURE NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
28577-5	Протокол (PROCEDURE NOTE)	Различные ({SETTING})	Стоматология (DENTISTRY)
11505-5	Протокол (PROCEDURE NOTE)	Различные ({SETTING})	Терапевт (PHYSICIAN)
28625-2	Протокол (PROCEDURE NOTE)	Различные ({SETTING})	Подиатрия (PODIATRY)
28580-9	Дневниковая запись (PROGRESS NOTE)	Различные ({SETTING})	Мануальный терапевт (CHIROPRACTOR)
28575-9	Дневниковая запись (PROGRESS NOTE)	Различные ({SETTING})	Фельдшер (NURSE PRACTITIONER)
18744-3	Протокол исследования (STUDY REPORT)	Респираторная система (RESPIRATORY SYSTEM)	Бронхоскопия (BRONCHOSCOPY)
11506-3	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34126-3	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Отделение интенсивной терапии (CRITICAL CARE UNIT)	Исполнитель ({PROVIDER})
15507-7	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Служба скорой помощи (EMERGENCY DEPARTMENT)	Исполнитель ({PROVIDER})
34129-7	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Лечение на дому (HOME HEALTH)	Исполнитель ({PROVIDER})
34130-5	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Стационар (HOSPITAL)	Исполнитель ({PROVIDER})
34131-3	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Амбулатория (OUTPATIENT)	Исполнитель ({PROVIDER})
18733-6	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Лечащий врач (ATTENDING PHYSICIAN)
34124-8	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Амбулатория	Кардиология
	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	(OUTPATIENT)	(CARDIOLOGY)
34125-5	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Лечение на дому (HOME HEALTH CARE)	Менеджер медицинских услуг (CASE MANAGER)
18762-5	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Мануальный терапевт (CHIROPRACTOR)
28569-2	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Врач-консультант (CONSULTING PHYSICIAN)
34127-1	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Амбулатория (OUTPATIENT)	Стоматолог-гигиенист (DENTAL HYGIENIST)
28617-9	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Стоматология (DENTISTRY)
34128-9	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Амбулатория (OUTPATIENT)	Стоматология (DENTISTRY)
34900-1	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Общая медицинская практика (GENERAL MEDICINE)

Продолжение таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
34901-9	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Амбулатория (OUTPATIENT)	Общая медицинская практика (GENERAL MEDICINE)
34904-3	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Психическое здоровье (MENTAL HEALTH)
18764-1	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Фельдшер (NURSE PRACTITIONER)
28623-7	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Сестринское дело (NURSING)
11507-1	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Трудотерапия (OCCUPATIONAL THERAPY)
34132-1	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Амбулатория (OUTPATIENT)	Фармакология (PHARMACY)
11508-9	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Физиотерапия (PHYSICAL THERAPY)
11509-7	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Подиатрия (PODIATRY)
28627-8	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Психиатрия (PSYCHIATRY)
11510-5	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Психология (PSYCHOLOGY)
28656-7	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Социальная служба (SOCIAL SERVICE)
11512-1	Очередная оценка состояния (SUBSEQUENT EVALUATION NOTE)	Различные ({SETTING})	Логопедия (SPEECH THERAPY)
34133-9	Заключение по эпизоду лечения (этапный эпикриз) (SUMMARIZATION OF EPISODE NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34134-7	Контрольная запись (SUPERVISORY NOTE)	Амбулатория (OUTPATIENT)	Лечащий врач (ATTENDING PHYSICIAN)
34135-4	Контрольная запись (SUPERVISORY NOTE)	Амбулатория (OUTPATIENT)	Лечащий врач — кардиолог (ATTENDING PHYSICIAN.CARDIOLOGY)
34136-2	Контрольная запись (SUPERVISORY NOTE)	Амбулатория (OUTPATIENT)	Лечащий врач — гастроэнтеролог (ATTENDING PHYSICIAN. GASTROENTEROLOGY)
11504-8	Протокол хирургического вмешательства (SURGICAL OPERATION NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34137-0	Протокол хирургического вмешательства (SURGICAL OPERATION NOTE)	Амбулатория (OUTPATIENT)	Исполнитель ({PROVIDER})
28583-3	Протокол хирургического вмешательства (SURGICAL OPERATION NOTE)	Различные ({SETTING})	Стоматология (DENTISTRY)
28573-4	Протокол хирургического вмешательства (SURGICAL OPERATION NOTE)	Различные ({SETTING})	Терапевт (PHYSICIAN)

Окончание таблицы 147

LOINC_ NUM	COMPONENT (тип услуги)	SYSTEM (условия медицинской помощи)	METHOD_TYP (предметная область и/или профессиональный уровень)
28624-5	Протокол хирургического вмешательства (SURGICAL OPERATION NOTE)	Различные ({SETTING})	Подиатрия (PODIATRY)
34138-8	История заболевания и физикальное исследование (TARGETED HISTORY AND PHYSICAL NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34748-4	Запись об обращении по телефону (TELEPHONE ENCOUNTER NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34139-6	Запись об обращении по телефону (TELEPHONE ENCOUNTER NOTE)	Различные ({SETTING})	Сестринское дело (NURSING)
34844-1	Запись об обращении по телефону (TELEPHONE ENCOUNTER NOTE)	Амбулатория (OUTPATIENT)	Социальная работа (SOCIAL WORK)
34140-4	Заключение при переводе пациента (переводной эпикриз) (TRANSFER SUMMARIZATION NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
18761-7	Заключение при переводе пациента (переводной эпикриз) (TRANSFER SUMMARIZATION NOTE)	Различные ({SETTING})	Исполнитель ({PROVIDER})
34755-9	Заключение при переводе пациента (переводной эпикриз) (TRANSFER SUMMARIZATION NOTE)	Различные ({SETTING})	Интенсивная терапия (CRITICAL CARE)
34770-8	Заключение при переводе пациента (переводной эпикриз) (TRANSFER SUMMARIZATION NOTE)	Различные ({SETTING})	Общая медицинская практика (GENERAL MEDICINE)
28651-8	Заключение при переводе пациента (переводной эпикриз) (TRANSFER SUMMARIZATION NOTE)	Различные ({SETTING})	Сестринское дело (NURSING)
28616-1	Заключение при переводе пациента (переводной эпикриз) (TRANSFER SUMMARIZATION NOTE)	Различные ({SETTING})	Терапевт (PHYSICIAN)
28618-7	Запись о посещении (VISIT NOTE)	Различные ({SETTING})	Стоматология (DENTISTRY)
28578-3	Запись о посещении (VISIT NOTE)	Различные ({SETTING})	Трудотерапия (OCCUPATIONAL THERAPY)
28579-1	Запись о посещении (VISIT NOTE)	Различные ({SETTING})	Физиотерапия (PHYSICAL THERAPY)
28568-4	Запись о посещении (VISIT NOTE)	Служба скорой помощи (EMERGENCY DEPARTMENT)	Терапевт (PHYSICIAN)
28571-8	Запись о посещении (VISIT NOTE)	Различные ({SETTING})	Логопедия (SPEECH THERAPY)
28628-6	Запись о посещении (VISIT NOTE)	Различные ({SETTING})	Психиатрия (PSYCHIATRY)

5.8.3 АКД и семантическая интероперабельность

Долгосрочной целью АКД и других спецификаций семейства стандартов V3 является постепенное достижение все большей и большей «семантической интероперабельности», которую можно определить как способность двух приложений совместно использовать данные без предварительных доработок, так, чтобы компоненты каждого приложения, отвечающие за поддержку принятия решений, могли продолжать надежную работу с полученными данными.

Разработчики АКД старались достичь наивысшего уровня ограничений, возможного для международного стандарта. В тех случаях, когда не удается достичь международного консенсуса и когда сценарии использования в разных странах пока что препятствуют консенсусу, разработчики вынуждены оставаться на более общем уровне. В таких предметных областях усилия направлены на гармонизацию спецификации и достижение консенсуса, способствующие дальнейшей семантической интероперабельности, что будет отражено в следующих версиях АКД.

Хотя теоретические основы, заложенные в модель RIM, в спецификацию АКД и в общую модель клинических утверждений (HL7 Clinical Statement Model), являются критичными компонентами семантической интероперабельности, они пока что не являются достаточными, в особенности из-за отсутствия глобального решения по терминологии и того факта, что каждая терминология различным способом пересекается с моделью RIM. Такие терминологические решения лежат вне области применения АКД и должны обсуждаться на различных национальных и международных форумах.

5.8.4 Изменения по сравнению с первым выпуском АКД

Первый выпуск АКД (CDA, Release One) был одобрен организацией ANSI в ноябре 2000 г. и представлял собой первую спецификацию, произведенную от модели RIM (Reference Information Model). С тех пор модель RIM перешла в более зрелое состояние, равно как и методология, используемая для создания спецификаций, производных от модели RIM. Кроме того, первые разработчики, принявшие АКД на вооружение, предоставили новые варианты использования для включения в спецификацию.

Базовая модель во втором выпуске АКД существенно не изменилась. АКД-документ имеет заголовок и тело. Тело содержит вложенные разделы. Содержание этих разделов может быть закодировано используя стандартные словари, и может включать в себя «подразделы». В первом выпуске АКД подразделы могли включать в себя такие данные, как строки символов, гиперссылки и мультимедийные данные.

К числу основных эволюционных изменений, которые привели к созданию второго выпуска АКД, можно отнести то, что заголовок и тело стали полностью производными от модели RIM, а ассортимент подразделов, которые могут включаться в разделы АКД-документов, значительно вырос. Второй выпуск АКД обеспечивает возможность формального представления клинического содержания до той степени, которая обеспечивается моделью RIM.

Во втором выпуске АКД нашли свое отражение новые достижения комитета HL7 по разработке стандартов, использующих модели, основанные на языке XML. Вследствие эволюции модели RIM и методологии комитета HL7 по разработке стандартов, произошедшей с ноября 2000 г., во второй выпуск АКД был внесен ряд изменений по сравнению с первым выпуском.

5.8.4.1 Запрещенные компоненты

Следующие компоненты оставлены для обратной совместимости с первым выпуском АКД, но во втором выпуске их использование запрещено:

- ClinicalDocument/copyTime,
- ClinicalDocument/authenticator/signatureCode/@code="X",
- ClinicalDocument/legalAuthenticator/signatureCode/@code="X",
- ClinicalDocument/assignedAuthor/assignedAuthoringDevice/MaintainedEntity,
- ClinicalDocument/recordTarget/patientRole/patient/id,
- linkHtml.name,
- table.border, table.cellspacing, table.cellpadding.

Дальнейшее использование этих компонентов не рекомендовано.

5.8.4.2 Соответствие между первым и вторым выпусками АКД

В таблицах 148, 149 и 150 перечислены компоненты первого выпуска АКД и соответствующие им компоненты второго выпуска. В некоторых случаях, когда спецификация первого выпуска неоднозначна или когда участник, определенный в первом выпуске, был разделен на несколько участников во втором выпуске, при формальном отображении следует учитывать местное использование спецификации. Для этих случаев приводятся рекомендации, а не формальные правила.

Таблица 148 — Соответствие словарей первого и второго выпуска АКД

Словарь АКД, выпуск 1	Соответствующий словарь АКД, выпуск 2	Примечания
clinical_document_header / document_type_cd <= DocumentType (2.16.840.1.113883.6.1.10870) (CWE)	ClinicalDocument / code <= DocumentType (2.16.840.1.113883.6.1) (CWE)	
clinical_document_header / confidentiality_cd <= ServiceConfidentiality (2.16.840.1.113883.5.10228) (CWE)	ClinicalDocument / confidentialityCode <= x_BasicConfidentialityKind (2.16.840.1.113883.5.25) (CWE)	Из набора значений удалены «C», «D», «I», «S», «T». К набору значений добавлено «V»
clinical_document_header / document_relationship / document_relationship. type_cd <= ServiceRelationship (2.16.840.1.113883.5.10317) (CNE)	ClinicalDocument / relatedDocument / @typeCode <= x_ActRelationshipDocument (2.16.840.1.113883.5.1002) (CNE)	К набору значений добавлено «XFRM»
clinical_document_header / fulfills_order / fulfills_order. type_cd <= ServiceRelationship (2.16.840.1.113883.5.10317) (CNE)	ClinicalDocument / inFulfillmentOf / @typeCode <= FLFS (2.16.840.1.113883.5.1002) (CNE)	Набор значений не изменен
clinical_document_header / patient_encounter / practice_ setting_cd <= PracticeSetting (2.16.840.1.113883.5.10588) (CWE)	ClinicalDocument / componentOf / encompassingEncounter / location / healthCareFacility / code <= ServiceDeliveryLocationRoleType (2.16.840.1.113883.5.111) (CWE)	
clinical_document_header / authenticator / authenticator. type_cd <= ServiceActor (2.16.840.1.113883.5.10246) (CNE)	ClinicalDocument / authenticator / @typeCode <= AUTHEN (2.16.840.1.113883.5.90) (CNE)	Фиксированное значение изменено с «VRF» на «AUTHEN»
clinical_document_header / authenticator / signature_cd <= ServiceActorSignature (2.16.840.1.113883.5.10282) (CNE)	ClinicalDocument / authenticator / signatureCode <= ParticipationSignature (2.16.840.1.113883.5.89) (CNE)	Значение «X» запрещено
clinical_document_header / authenticator / person / person_name / person_name. type_cd <= PersonNamePurpose (2.16.840.1.113883.5.200) (CWE)	ClinicalDocument / authenticator / assignedEntity / assignedPerson / name / @use <= PersonNameUse (2.16.840.1.113883.5.45) (CNE)	Квалификатор расширяемости изменен с CWE на CNE. Во втором выпуске АКД набор значений включает в себя следующие перечисляемые значения, определенные в первом выпуске («A», «C», «I», «L», «R»). Для преобразования из первого выпуска во второй может потребоваться местная разметка в тех случаях, когда набор значений, взятых из первого выпуска, был расширен
	ClinicalDocument / legalAuthenticator / @typeCode <= LA (2.16.840.1.113883.5.90) (CNE)	Фиксированное значение изменено с «SPV» на «AUTHEN»
clinical_document_header / legal_authenticator / signature_ cd <= ServiceActorSignature (2.16.840.1.113883.5.10282) (CNE)	ClinicalDocument / legalAuthenticator / signatureCode <= ParticipationSignature (2.16.840.1.113883.5.89) (CNE)	Значение «X» теперь запрещено

Продолжение таблицы 148

Словарь АКД, выпуск 1	Соответствующий словарь АКД, выпуск 2	Примечания
clinical_document_header / intended_recipient / intended_ recipient.type_cd <= ServiceActor (2.16.840.1.113883.5.10246) (CNE)	ClinicalDocument / intendedRecipient / @typeCode <= x_InformationRecipient (2.16.840.1.113883.5.90) (CNE)	К набору значений добавлено «PRCP»
clinical_document_header / originator / originator.type_cd <= ServiceActor (2.16.840.1.113883.5.10246) (CNE)	ClinicalDocument / author / @typeCode <= AUT (2.16.840.1.113883.5.90) (CNE)	Набор значений не изменен
clinical_document_header / originating_organization / originating_ organization.type_cd <= ServiceActor (2.16.840.1.113883.5.10246) (CNE)	ClinicalDocument / custodian / @typeCode <= CST (2.16.840.1.113883.5.90) (CNE)	Набор значений не изменен
clinical_document_header / transcriptionist / transcriptionist. type_cd <= ServiceActor (2.16.840.1.113883.5.10246) (CNE)	ClinicalDocument / dataEnterer / @typeCode <= ENT (2.16.840.1.113883.5.90) (CNE)	Набор значений не изменен
clinical_document_header / provider / provider.type_cd <= ServiceActor (2.16.840.1.113883.5.10246) (CNE) (Value set «ASS», «CON», «PRF»)	ClinicalDocument / serviceEvent / performer / @typeCode <= x_ServiceEventPerformer (2.16.840.1.113883.5.90) (CNE) (набор значений “PRF”, “PPRF”, “SPRF”) ClinicalDocument / encompassingEncounter / responsibleParty / @typeCode <= RESP (2.16.840.1.113883.5.90) (CNE) ClinicalDocument / encompassingEncounter / encounterParticipant / @typeCode <= x_EncounterParticipant (2.16.840.1.113883.5.90) (CNE) (набор значений «ADM», «ATND», «CON» , «DIS» , «REF»)	Участник provider, определенный в первом выпуске АКД, расщеплен на три разных участника. Для определения точного соответствия необходимо учесть местное использование элемента provider в первом выпуске. Значение «ASS» в первом выпуске АКД коррелирует со значением «SPRF» во втором выпуске
clinical_document_header / provider / function_cd <= ServiceActorFunction (2.16.840.1.113883.5.10267) (CWE)	ClinicalDocument / serviceEvent / performer / functionCode <= ParticipationFunction (2.16.840.1.113883.5.88) (CWE)	
clinical_document_header / service_actor / service_actor. type_cd <= ServiceActor (2.16.840.1.113883.5.10246) (CWE)	ClinicalDocument / participant / @ typeCode <= ParticipationType (2.16.840.1.113883.5.90) (CNE)	Квалификатор расширяемости изменен с CWE на CNE. Для преобразования из первого выпуска во второй может потребоваться местная разметка в тех случаях, когда набор значений, взятых из первого выпуска, был расширен
clinical_document_header / service_actor / signature_cd <= ServiceActorSignature (2.16.840.1.113883.5.10282) (CNE)	ClinicalDocument / authenticator / signatureCode <= ParticipationSignature (2.16.840.1.113883.5.89) (CNE)	Значение «X» теперь запрещено
clinical_document_header / patient / administrative_gender_ cd <= AdministrativeGender (2.16.840.1.113883.5.1) (CWE)	ClinicalDocument / recordTarget / patientRole / patient / administrativeGenderCode <= AdministrativeGender (2.16.840.1.113883.5.1) (CWE)	

Продолжение таблицы 148

Словарь АКД, выпуск 1	Соответствующий словарь АКД, выпуск 2	Примечания
clinical_document_header / originating_device / originating_ device.type_cd <= ServiceTargetType (2.16.840.1.113883.5.10285) (CNE)	ClinicalDocument / author / @typeCode <= AUT (2.16.840.1.113883.5.90) (CNE)	Значение «ODV», определенное в первом выпуске АКД, отображается на значение «AUT» во втором выпуске
clinical_document_header / originating_device / device / responsibility / responsibility. type_cd <= MaterialResponsibility (2.16.840.1.113883.5.10416) (CWE)	ClinicalDocument / author / assignedAuthor / assignedAuthoringDevice / asMaintainedEntity / @classCode <= MNT (2.16.840.1.113883.5.110) (CNE)	Квалификатор расширяемости изменен с CWE на CNE. Во втором выпуске АКД набор значений включает в себя следующие перечисляемые значения, определенные в первом выпуске «MNT»). Для преобразования из первого выпуска во второй может потребоваться местная разметка в тех случаях, когда набор значений, взятых из первого выпуска, был расширен
clinical_document_header / service_target / service_target. type_cd <= ServiceTargetType (2.16.840.1.113883.5.10285) (CWE)	ClinicalDocument / participant / @ typeCode <= ParticipationType (2.16.840.1.113883.5.90) (CNE)	Квалификатор расширяемости изменен с CWE на CNE. Для преобразования из первого выпуска во второй может потребоваться местная разметка в тех случаях, когда набор значений, взятых из первого выпуска, был расширен
section / caption / caption_cd (CWE)	section / code <= DocumentSectionType (2.16.840.1.113883.6.1) (CWE)	См. также элемент caption/caption_ cd в следующей строке. Соответствие заголовка раздела во втором выпуске АКД отличается от соответствия других заголовков.
caption / caption_cd (CWE)	@styleCode <= StyleType (2.16.840.1.113883.19.5.1) (CWE)	См. также элемент section/caption/ caption_cd в предыдущей строке. Соответствие заголовка раздела во втором выпуске АКД отличается от соответствия других заголовков

Таблица 149 — Соответствие заголовков документа в первом и втором выпуске АКД

Компонент АКД, выпуск 1	Соответствующий компонент АКД, выпуск 2	Примечания
levelone	ClinicalDocument	
levelone / clinical_document_header	ОТСУТСТВУЕТ	Тег оболочки заголовка в настоящее время отсутствует.
clinical_document_header / id	ClinicalDocument / id	
clinical_document_header / set_id	ClinicalDocument / setId	
clinical_document_header / version_nbr	ClinicalDocument / versionNumber	
clinical_document_header / document_type_cd	ClinicalDocument / code	

Продолжение таблицы 149

Компонент АКД, выпуск 1	Соответствующий компонент АКД, выпуск 2	Примечания
clinical_document_header / service_tmr	ClinicalDocument / documentationOf / serviceEvent / effectiveTime	Тип данных GTS, использованный в первом выпуске АКД, ограничен до интервала дат и времени во втором выпуске. Если необходимо добавить более сложную спецификацию, может использоваться местная разметка
clinical_document_header / origination_dttm	ClinicalDocument / effectiveTime	
clinical_document_header / copy_dttm	ClinicalDocument / copyTime	
clinical_document_header / confidentiality_cd	ClinicalDocument / confidentialityCode	Кратность изменена с [0..*] на [0..1]. В первом выпуске АКД признак конфиденциальности задается в заголовке и используется в теле. Во втором выпуске признак конфиденциальности задается в заголовке и может быть переопределен в теле
clinical_document_header / document_relationship	ClinicalDocument / relatedDocument	
document_relationship / document_ relationship.type_cd	relatedDocument / @typeCode	
document_relationship / related_ document	relatedDocument / parentDocument	
document_relationship / related_ document / id	relatedDocument / parentDocument / id	
document_relationship / related_ document / set_id	relatedDocument / parentDocument / setId	
document_relationship / related_ document / version_nbr	relatedDocument / parentDocument / versionNumber	
clinical_document_header / fulfills_order	ClinicalDocument / inFulfillmentOf	
fulfills_order / fulfills_order.type_cd	inFulfillmentOf / @typeCode	
fulfills_order / order	inFulfillmentOf / order	
fulfills_order / order / id	inFulfillmentOf / order / id	
clinical_document_header / patient_ encounter	ClinicalDocument / componentOf / encompassingEncounter	
patient_encounter / id	encompassingEncounter / id	
fulfills_order / order	inFulfillmentOf / order	
fulfills_order / order / id	inFulfillmentOf / order / id	
clinical_document_header / patient_ encounter	ClinicalDocument / componentOf / encompassingEncounter	
patient_encounter / id	encompassingEncounter / id	

Продолжение таблицы 149

Компонент АКД, выпуск 1	Соответствующий компонент АКД, выпуск 2	Примечания
patient_encounter / practice_ setting_cd	encompassingEncounter / location / healthCareFacility / code	
patient_encounter / encounter_tmr	encompassingEncounter / effectiveTime	
patient_encounter / service_location	encompassingEncounter / location / healthCareFacility	
patient_encounter / service_ location / id	encompassingEncounter / location / healthCareFacility / id	
patient_encounter / service_location / addr	encompassingEncounter / location / healthCareFacility / location / addr	
clinical_document_header / authenticator	ClinicalDocument / authenticator	Тип интервала дат и времени, использованный в первом выпуске АКД, ограничен до момента даты и времени во втором выпуске. Если необходимо добавить более сложную спецификацию, может использоваться местная разметка
authenticator / authenticator.type_cd	authenticator / @typeCode	
authenticator / participation_tmr	authenticator / time	
authenticator / signature_cd	authenticator / signatureCode	В первом выпуске АКД можно указать либо планируемую подпись контролера (код подписи «X») или наличие его подписи (код подписи «S»). Во втором выпуске разрешен только факт подписи, поэтому значение «X» запрещено
authenticator / person	authenticator / assignedEntity / assignedPerson	
authenticator / person / id	authenticator / assignedEntity / id	
authenticator / person / person_name	authenticator / assignedEntity / assignedPerson / name	
authenticator / person / person_name / effective_tmr	authenticator / assignedEntity / assignedPerson / name / validTime	
authenticator / person / person_name / nm	authenticator / assignedEntity / assignedPerson / name	
authenticator / person / person_name / person_name.type_cd	authenticator / assignedEntity / assignedPerson / name / @use	
authenticator / person / addr	authenticator / assignedEntity / addr /	
authenticator / person / telecom	authenticator / assignedEntity / telecom	
clinical_document_header / legal_ authenticator	ClinicalDocument / legalAuthenticator	
legal_authenticator / legal_ authenticator.type_cd	legalAuthenticator / @typeCode	

Продолжение таблицы 149

Компонент АКД, выпуск 1	Соответствующий компонент АКД, выпуск 2	Примечания
legal_authenticator / participation_tmr	legalAuthenticator / time	Тип интервала дат и времени, использованный в первом выпуске АКД, ограничен до момента даты и времени во втором выпуске. Если необходимо добавить более сложную спецификацию, может использоваться местная разметка
legal_authenticator / signature_cd	legalAuthenticator / signatureCode	В первом выпуске АКД можно указать либо планируемую подпись контролера (код подписи «X») или наличие его подписи (код подписи «S»). Во втором выпуске разрешен только факт подписи, поэтому значение «X» запрещено
legal_authenticator / person	legalAuthenticator / assignedEntity / assignedPerson	
clinical_document_header / intended_recipient	ClinicalDocument / intendedRecipient	
intended_recipient / intended_recipient.type_cd	intendedRecipient / @typeCode	
clinical_document_header / originator	ClinicalDocument / author	
originator / originator.type_cd	author / @typeCode	
originator / participation_tmr	author / time	Тип интервала дат и времени, использованный в первом выпуске АКД, ограничен до момента даты и времени во втором выпуске. Если необходимо добавить более сложную спецификацию, может использоваться местная разметка
clinical_document_header / originating_organization	ClinicalDocument / author ClinicalDocument / custodian	Участник originating_organization, определенный в первом выпуске АКД, соответствует атрибуту scoper элемента author И контролирующей организации, поскольку он охватывает оба этих понятия в первом выпуске
originating_organization / originating_organization.type_cd	custodian / @typeCode	
originating_organization / organization	author / assignedAuthor / representedOrganization custodian / assignedCustodian / representedCustodianOrganization	
originating_organization / organization / id	author / assignedAuthor / representedOrganization / id custodian / assignedCustodian / representedCustodianOrganization / id	

Продолжение таблицы 149

Компонент АКД, выпуск 1	Соответствующий компонент АКД, выпуск 2	Примечания
originating_organization / organization / organization.nm	author / assignedAuthor / representedOrganization / name custodian / assignedCustodian / representedCustodianOrganization / name	
originating_organization / organization / addr	author / assignedAuthor / representedOrganization / addr custodian / assignedCustodian / representedCustodianOrganization / addr	
clinical_document_header / transcriptionist	ClinicalDocument / dataEnterer	
transcriptionist / transcriptionist. type_cd	dataEnterer / @typeCode	
transcriptionist / participation_tmr	dataEnterer / time	Тип интервала дат и времени, использованный в первом выпуске АКД, ограничен до момента даты и времени во втором выпуске. Если необходимо добавить более сложную спецификацию, может использоваться местная разметка
transcriptionist / participation_tmr	dataEnterer / time	Тип интервала дат и времени, использованный в первом выпуске АКД, ограничен до момента даты и времени во втором выпуске. Если необходимо добавить более сложную спецификацию, может использоваться местная разметка
clinical_document_header / provider	ClinicalDocument / serviceEvent / performer ClinicalDocument / encompassingEncounter / responsibleParty ClinicalDocument / encompassingEncounter / encounterParticipant	Участник provider, определенный в первом выпуске АКД, расщеплен на три разных участника. Для определения точного соответствия необходимо учесть местное использование элемента provider в первом выпуске
clinical_document_header / service_ actor	ClinicalDocument / participant	В зависимости от местного использования первого выпуска АКД может также отображаться на одного из нескольких более специфичных участников (например, informant, performer, responsibleParty, encounterParticipant)
service_actor / service_actor.type_cd	participant / @typeCode	
service_actor / participation_tmr	participant / time	

Продолжение таблицы 149

Компонент АКД, выпуск 1	Соответствующий компонент АКД, выпуск 2	Примечания
service_actor / signature_cd	authenticator / signatureCode	В зависимости от местного использования первого выпуска АКД подписи участника service_actor могут быть представлены во втором выпуске как подписи дополнительных контролеров
clinical_document_header / patient	ClinicalDocument / recordTarget	
patient / patient.type_cd	recordTarget / @typeCode	
patient / participation_tmr	NOT PRESENT	При необходимости для отображения может использоваться местная разметка
patient / person	recordTarget / patientRole / patient	
patient / person / id	recordTarget / patientRole / patient / id	
patient / is_known_by	recordTarget / patientRole	
patient / is_known_by / id	recordTarget / patientRole / id	
patient / is_known_by / is_known_to	recordTarget / patientRole / providerOrganization	Первый выпуск АКД допускает несколько отношений is_known_by, каждое со своим собственным отношением is_known_to. Во втором выпуске может быть указан только один атрибут providerOrganization. Там, где в первом выпуске перечислено несколько организаций, одной из них является контролирующая организация, соответствующая атрибуту providerOrganization. Для указания других значений может быть использована местная разметка
patient / is_known_by / is_known_to / id	recordTarget / patientRole / providerOrganization / id	
patient / birth_dttm	recordTarget / patientRole / patient / birthTime	
patient / administrative_gender_cd	recordTarget / patientRole / patient / administrativeGenderCode	
clinical_document_header / originating_device	ClinicalDocument / author	
originating_device / originating_device.type_cd	author / @typeCode	
originating_device / participation_tmr	author / time	Тип интервала дат и времени, использованный в первом выпуске АКД, ограничен до момента даты и времени во втором выпуске. Если необходимо добавить более сложную спецификацию, может использоваться местная разметка

Окончание таблицы 149

Компонент АКД, выпуск 1	Соответствующий компонент АКД, выпуск 2	Примечания
originating_device / device	author / assignedAuthor	
originating_device / device / id	author / assignedAuthor / id	
originating_device / device / responsibility	author / assignedAuthor / assignedAuthoringDevice / asMaintainedEntity	
originating_device / device / responsibility / responsibility.type_cd	author / assignedAuthor / assignedAuthoringDevice / asMaintainedEntity / @classCode	
originating_device / device / responsibility / responsibility_tmr	author / assignedAuthor / assignedAuthoringDevice / asMaintainedEntity / effectiveTime	
clinical_document_header / service_ target	ClinicalDocument / participant	В зависимости от местного использования первого выпуска АКД может также отображаться на одного из нескольких более специфичных участников (например, informant, performer, responsibleParty, encounterParticipant)
service_target / service_target. type_cd	participant / @typeCode	
service_target / participation_tmr	participant / time	

Таблица 150 — Соответствие тел документа в первом и втором выпуске АКД

Компонент АКД, выпуск 1	Соответствующий компонент АКД, выпуск 2	Примечания
levelone / body / non_xml	ClinicalDocument / component / nonXMLBody	
levelone / body / section	ClinicalDocument / component / structuredBody / section	
section / @originator	section / author	В первом выпуске АКД все авторы перечисляются в заголовке, а в теле на них даются ссылки. Во втором выпуске авторы, указанные в заголовке, распространяются на тело и могут быть там переопределены (как описано в подразделе «CDA Context»)
section / @confidentiality	section / confidentialityCode	В первом выпуске АКД все коды конфиденциальности перечисляются в заголовке, а в теле на них даются ссылки. Во втором выпуске коды конфиденциальности, указанные в заголовке, распространяются на тело и могут быть там переопределены (как описано в подразделе «CDA Context»)
section / @xml:lang	section / languageCode	
section / caption	section / title	

Продолжение таблицы 150

Компонент АКД, выпуск 1	Соответствующий компонент АКД, выпуск 2	Примечания
section / caption / link	ОТСУТСТВУЕТ	От местных правил может зависеть, надо ли отображать этот элемент на тег linkHtml в повествовательном блоке, определенный во втором выпуске АКД.
section / caption / caption_cd	section / code	См. также ниже строку с описанием отображения элемента caption/ caption_cd. Соответствие заголовка раздела во втором выпуске АКД отличается от соответствия других заголовков
@originator	См. примечания	Во втором выпуске АКД авторство раздела относится к повествовательному блоку. Если требуется приписать авторство конкретному утверждению, описанному во вложенном подразделе, то можно в нем сделать ссылку на повествовательный блок
@confidentiality	ОТСУТСТВУЕТ	Конфиденциальность не может быть описана на уровне, более низком, нежели раздел
@xml:lang	@language	Атрибут language во втором выпуске АКД используется так же, как атрибут xml:lang в первом выпуске
content	content	
link	linkHtml	В первом выпуске АКД элемент link более не существует. Он имеет обязательный дочерний элемент link_html, который соответствует тегу linkHtml во втором выпуске
coded_entry	section / entry / act	
coded_entry / coded_entry.id	section / entry / act / id	
coded_entry / coded_entry.value	section / entry / act / code	
observation_media	section / entry / observationMedia	Для ссылки на подраздел типа observationMedia используйте в повествовательном блоке тег renderMultiMedia
observation_media / observation_ media.id	section / entry / observationMedia / id	
observation_media / observation_ media.value	section / entry / observationMedia / value	

Окончание таблицы 150

Компонент АКД, выпуск 1	Соответствующий компонент АКД, выпуск 2	Примечания
local_markup	См. примечания	Местная разметка запрещена в повествовательном блоке второго выпуска АКД, поскольку она препятствует возможности распознания разметки, необходимой для правильного отображения документа. Другие пространства имен можно использовать вне повествовательного блока (как описано в подразделе «CDA Extensibility»)
caption	caption	
caption / caption_cd	@styleCode	См. также выше строку с описанием отображения элемента section/caption/caption_cd. Соответствие заголовка раздела во втором выпуске АКД отличается от соответствия других заголовков
paragraph	paragraph	
List	list	
Table	table	Во втором выпуске АКД модель содержания элемента table более не содержит элемент «tr». Он теперь включен в элемент «tbody»

6 Иерархическое описание АКД-документов и графическое представление модели R-MIM

6.1 Иерархическое описание АКД-документов (POCD_HD000040)

Иерархическое описание АКД-документов (POCD_HD000040) приведено в таблице 151.

Таблица 151— Иерархическое описание АКД-документов (POCD_HD000040)

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
	CDA (POCD_HD000040) Иерархическое описание											
	Clinical Document	0..1			Клинический документ	Document	ClinicalDocument	H				
1	typeId	1..1	O	O2	Тип предметной области	InfrastructureRoot	II	T				По умолчанию: @root=»2.16.840.1.113883.1.3»; @extension=» POCD_HD000040»
2	classCode	1..1	O	O2	Тип информационного объекта	Act	CS	T	DOCCLIN	CNE		По умолчанию: DOCCLIN
3	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		По умолчанию: EVN
4	id	1..1		O2	Идентификатор документа	Act	II	T				
5	code	1..1		O2	Тип документа	Act	CE	T	DocumentType	CWE		
6	title	0..1			Наименование документа	Act	ST	T				
7	effectiveTime	1..1		O2	Дата и время создания	Act	TS	T				
8	confidentialityCode	1..1		O2	Код конфиденциальности	Act	CE	T	x_Basic ConfidentialityKind	CWE		
9	languageCode	0..1			Язык документа	Act	CS	T	HumanLanguage	CNE		По умолчанию: @code System=» 2.16.840.1.113883.6.121»

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
10	setId	0..1			Идентификатор набора документов	Context-Structure	II	T				
11	versionNumber	0..1			Номер версии	Context-Structure	INT	T				
12	copyTime	0..1			Время копирования	Document	TS	T				В текущей версии запрещен
13	recordTarget	1..*			Целевая медицинская карта	Act	SET<RecordTarget>	H				
14	typeCode	1..1	O	O2	Категория объекта	Participation	CS	T	RCT	CNE		По умолчанию: RCT
15	contextControlCode	1..1	O	O2	Код управления контекстом	Participation	CS	T	OP	CNE		По умолчанию: OP
16	patientRole	1..1			Роль пациента	Participation	PatientRole	H				
17	classCode	1..1	O	O2	Класс объекта	Role	CS	T	PAT	CNE		По умолчанию: PAT
18	id	1..*			Идентификатор пациента	Role	SET<II>	T				
19	addr	0..*			Адрес места жительства	Role	SET<AD>	T				
20	telecom	0..*			Телекоммуникационный адрес	Role	SET<TEL>	T				
21	patient	0..1			Пациент	Role	Patient	H				
22	classCode	1..1	O	O2	Класс объекта	Entity	CS	T	PSN	CNE		По умолчанию: PSN
23	determinerCode	1..1	O	O2	Категория объекта	Entity	CS	T	INSTANCE	CNE		По умолчанию: INSTANCE

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
24	id	0..1			Идентификатор	Entity	II	T				В текущей версии запрещен
25	name	0..*			ФИО	Entity	SET<PN>	T				
26	administrative GenderCode	0..1			Административный пол	Living-Subject	CE	T	AdministrativeGender	CWE		
27	birthTime	0..1			Дата и время рождения	Living-Subject	TS	T				
28	maritalStatusCode	0..1			Семейное положение	Person	CE	T	MaritalStatus	CWE		
29	religious AffiliationCode	0..1			Вероисповедование	Person	CE	T	ReligiousAffiliation	CWE		
30	raceCode	0..1			Раса	Person	CE	T	Race	CWE		
31	ethnicGroupCode	0..1			Этническая группа	Person	CE	T	Ethnicity	CWE		
32	guardian	0..*			Представитель пациента	Entity	SET<Guardian>	H				
33	classCode	1..1	O	O2	Класс объекта	Role	CS	T	GUARD	CNE		По умолчанию: GUARD
34	id	0..*			Идентификатор представителя	Role	SET<II>	T				
35	code	0..1			Отношение к пациенту	Role	CE	T	RoleCode	CWE		
36	addr	0..*			Адрес	Role	SET<AD>	T				
37	telecom	0..*			Телекоммуникационный адрес	Role	SET<TEL>	T				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
38	guardianChoice	1..1			Выбор организации или физического лица	Role	Person | Organization	Н			Да	
39	guardianPerson	1..1			Представитель-физическое лицо	Living-Subject	Person	Н				
40	classCode	1..1	О	О2	Класс объекта	Entity	CS	Т	PSN	CNE		По умолчанию: PSN
41	determinerCode	1..1	О	О2	Категория объекта	Entity	CS	Т	INSTANCE	CNE		По умолчанию: INSTANCE
42	name	0..*			ФИО	Entity	SET<PN>	Т				
43	guardian Organization	1..1			Представитель-организация	Entity	Organization	И				
44	birthplace	0..1			Место рождения	Entity	Birthplace	Н				
45	classCode	1..1	О	О2	Класс объекта	Role	CS	Т	BIRTHPL	CNE		По умолчанию: BIRTHPL
46	place	1..1			Место	Role	Place	Н				
47	classCode	1..1	О	О2	Класс объекта	Entity	CS	Т	PLC	CNE		По умолчанию: PLC
48	determinerCode	1..1	О	О2	Категория объекта	Entity	CS	Т	INSTANCE	CNE		По умолчанию: INSTANCE
49	name	0..1			Название	Entity	EN	Т				
50	addr	0..1			Адрес места	Place	AD	Т				
51	language Communication	0..*			Язык общения	Entity	SET<LanguageCommunication>	Н				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
52	languageCode	0..1			Язык	Language Communication	CS	T	HumanLanguage	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.6.121»
53	modeCode	0..1			Код завершенности	Language Communication	CE	T	LanguageAbilityMode	CWE		
54	proficiency LevelCode	0..1			Степень владения языком	Language Communication	CE	T	LanguageAbilityProficiency	CWE		
55	preferenceInd	0..1			Предпочтительный	Language Communication	BL	T				
56	provider Organization	0..1			Поставщик медицинской помощи	Role	Organization	H				
57	classCode	1..1	O	O2	Класс объекта	Entity	CS	T	ORG	CNE		По умолчанию: ORG
58	determinerCode	1..1	O	O2	Категория объекта	Entity	CS	T	INSTANCE	CNE		По умолчанию: INSTANCE
59	id	0..*			Идентификатор поставщика	Entity	SET<II>	T				
60	name	0..*			Название организации	Entity	SET<ON>	T				
61	telecom	0..*			Телекоммуникационный адрес	Entity	SET<TEL>	T				
62	addr	0..*			Почтовый адрес	Organization	SET<AD>	T				
63	standardIndustry ClassCode	0..1			ОКОНХ	Organization	CE	T	OrganizationIndustryClass	CWE		

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
64	asOrganization PartOf	0..1			Вышестоящая организация	Entity	OrganizationPartOf	H				
65	classCode	1..1	O	O2	Класс объекта	Role	CS	T	PART	CNE		По умолчанию: PART
66	id	0..*		O2	Идентификатор организации	Role	SET<II>	T				
67	code	0..1			Код роли	Role	CE	T	RoleCode	CWE		
68	statusCode	0..1			Код отношения к поставщику	Role	CS	T	RoleStatus	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.14»
69	effectiveTime	0..1			Дата и время установления отношения	Role	IVL<TS>	T				
70	wholeOrganization	0..1			Сведения об организации	Role	Organization	P				
71	author	1..*			Автор документа	Act	SET<Author>	H				
72	typeCode	1..1	O	O2	Тип объекта	Participation	CS	T	AUT	CNE		По умолчанию: AUT
73	functionCode	0..1			Код функции	Participation	CE	T	ParticipationFunction	CWE		
74	contextControlCode	1..1	O	O2	Код управления контекстом	Participation	CS	T	OP	CNE		По умолчанию: OP
75	time	1..1		O2	Participation	Participation	TS	T				
76	assignedAuthor	1..1			Назначенный автор	Participation	AssignedAuthor	H				
77	classCode	1..1	O	O2	Код класса объекта	Role	CS	T	ASSIGNED	CNE		По умолчанию: ASSIGNED

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
78	id	1..*		O2	Идентификатор автора	Role	SET<II>	Т				
79	code	0..1			Код роли	Role	CE	Т	RoleCode	CWE		
80	addr	0..*			Адрес	Role	SET<AD>	Т				
81	telecom	0..*			Телекоммуникационный адрес	Role	SET<TEL>	Т				
82	assigned AuthorChoice	0..1			Выбор физического лица или устройства	Role	Person | AuthoringDevice	Н			Да	
83	assignedPerson	1..1			Назначенное лицо	Living-Subject	Person	И				
84	assigned AuthoringDevice	1..1			Назначенное устройство	ManufacturedMaterial	AuthoringDevice	Н				
85	classCode	1..1	O	O2	Код класса объекта	Entity	CS	Т	DEV	CNE		По умолчанию: DEV
86	determinerCode	1..1	O	O2	Код категории объекта	Entity	CS	Т	INSTANCE	CNE		По умолчанию: INSTANCE
87	code	0..1			Код роли	Entity	CE	Т	EntityCode	CWE		
88	manufacturer ModelName	0..1			Модель устройства	Device	SC	Т				
89	softwareName	0..1			Наименование программного обеспечения	Device	SC	Т				
90	asMaintainedEntity	0..*			Управляемое устройство	Entity	SET<Maintained-Entity>	Н				В текущей версии запрещен
91	classCode	1..1	O	O2	Класс роли	Role	CS	Т	MNT	CNE		По умолчанию: MNT

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
92	effectiveTime	0..1			Время использования	Role	IVL<TS>	Т				
93	maintainingPerson	1..1			Оператор устройства	Role	Person	И				
94	represented Organization	0..1			Место работы	Role	Organization	И				
95	dataEnterer	0..1			Оператор по вводу данных	Act	DataEnterer	Н				
96	typeCode	1..1	О	О2	Тип объекта	Participation	CS	Т	ENT	CNE		По умолчанию: ENT
97	contextControlCode	1..1	О	О2	Код управления контекстом	Participation	CS	Т	OP	CNE		По умолчанию: OP
98	time	0..1			Время участия	Participation	TS	Т				
99	assignedEntity	1..1			Назначенный субъект	Participation	AssignedEntity	Н				
100	classCode	1..1	О	О2	Класс роли	Role	CS	Т	ASSIGNED	CNE		По умолчанию: ASSIGNED
101	id	1..*		О2	Идентификатор субъекта	Role	SET<II>	Т				
102	code	0..1			Код роли	Role	CE	Т	RoleCode	CWE		
103	addr	0..*			Адрес	Role	SET<AD>	Т				
104	telecom	0..*			Телекоммуникационный адрес	Role	SET<TEL>	Т				
105	assignedPerson	0..1			Назначенное лицо	Role	Person	И				
106	represented Organization	0..1			Место работы	Role	Organization	И				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
107	informant	0..*			Информатор	Act	SET<Informant12>	Н				
108	typeCode	1..1	О	О2	Тип объекта	Participation	CS	Т	INF	CNE		По умолчанию: INF
109	contextControlCode	1..1	О	О2	Код управления контекстом	Participation	CS	Т	OP	CNE		По умолчанию: OP
110	informantChoice	1..1			Выбор типа участника	Participation	AssignedEntity | RelatedEntity	Н			Да	
111	assignedEntity	1..1			Назначенный субъект	Role	AssignedEntity	И				
112	relatedEntity	1..1			Близкое лицо или организация	Role	RelatedEntity	Н				
113	classCode	1..1	О	О2	Класс роли	Role	CS	Т	RoleClassMutualRelationship	CNE		
114	code	0..1			Код	Role	CE	Т	PersonalRelationshipRoleType	CWE		
115	addr	0..*			Адрес	Role	SET<AD>	Т				
116	telecom	0..*			Телекоммуникационный адрес	Role	SET<TEL>	Т				
117	effectiveTime	0..1			Время участия	Role	IVL<TS>	Т				
118	relatedPerson	0..1			Близкое лицо	Role	Person	И				
119	custodian	1..1			Обладатель документа	Act	Custodian	Н				
120	typeCode	1..1	О	О2	Тип объекта	Participation	CS	Т	CST	CNE		По умолчанию: CST

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
121	assignedCustodian	1..1			Уполномоченный обладатель	Participation	AssignedCustodian	H				
122	classCode	1..1	O	O2	Класс роли	Role	CS	T	ASSIGNED	CNE		По умолчанию: ASSIGNED
123	representedCustodianOrganization	1..1			Организация-обладатель	Role	CustodianOrganization	H				
124	classCode	1..1	O	O2	Класс роли	Entity	CS	T	ORG	CNE		По умолчанию: ORG
125	determinerCode	1..1	O	O2	Категория объекта	Entity	CS	T	INSTANCE	CNE		По умолчанию: INSTANCE
126	id	1..*	O	O2	Идентификатор	Entity	SET<II>	T				
127	name	0..1			Наименование	Entity	ON	T				
128	telecom	0..1			Телекоммуникационный адрес	Entity	TEL	T				
129	addr	0..1			Адрес	Organization	AD	T				
130	informationRecipient	0..*			Получатель информации	Act	SET<InformationRecipient>	H				
131	typeCode	1..1	O	O2	Код типа субъекта	Participation	CS	T	x_InformationRecipient	CNE		По умолчанию: PRCP
132	intendedRecipient	1..1			Предполагаемый получатель	Participation	IntendedRecipient	H				
133	classCode	1..1	O	O2	Класс роли	Role	CS	T	x_InformationRecipientRole	CNE		По умолчанию: ASSIGNED
134	id	0..*		O2	Идентификатор получателя	Role	SET<II>	T				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
135	addr	0..*			Адрес	Role	SET<AD>	Т				
136	telecom	0..*			Телекоммуникационный адрес	Role	SET<TEL>	Т				
137	information Recipient	0..1			Получатель информации	Role	Person	И				
138	received Organization	0..1			Организация-получатель	Role	Organization	И				
139	legalAuthenticator	0..1			Юридически ответственное лицо	Act		Н				
140	typeCode	1..1	О	О2	Код типа объекта	Participation	CS	Т	LA	CNE		По умолчанию: LA
141	contextControlCode	1..1	О	О2	Код управления контекстом	Participation	CS	Т	OP	CNE		По умолчанию: OP
142	time	1..1		О2	Время участия	Participation	TS	Т				
143	signatureCode	1..1		О2	Код подписи	Participation	CS	Т	ParticipationSignature	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.89»
144	assignedEntity	1..1			Уполномоченный субъект	Participation	AssignedEntity	И				
145	authenticator	0..*			Контролер	Act	SET<Authenticator>	Н				
146	typeCode	1..1	О	О2	Код типа субъекта	Participation	CS	Т	AUTHEN	CNE		По умолчанию: AUTHEN
147	time	1..1		О2	Время участия	Participation	TS	Т				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
148	signatureCode	1..1		O2	Код подписи	Participation	CS	Т	ParticipationSignature	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.89»
149	assignedEntity	1..1			Уполномоченный субъект	Participation	AssignedEntity	И				
150	participant	0..*			Участник	Act	SET<Participant1>	Н				
151	typeCode	1..1	O	O2	Код типа объекта	Participation	CS	Т	ParticipationType	CNE		
152	functionCode	0..1			Код функции	Participation	CE	Т	ParticipationFunction	CWE		
153	contextControlCode	1..1	O	O2	Код управления контекстом	Participation	CS	Т	OP	CNE		По умолчанию: OP
154	time	0..1			Время участия	Participation	IVL<TS>	Т				
155	associatedEntity	1..1			Связанный объект	Participation	AssociatedEntity	Н				
156	classCode	1..1	O	O2	Класс роли	Role	CS	Т	RoleClassAssociative	CNE		
157	id	0..*			Идентификатор	Role	SET<II>	Т				
158	code	0..1			Код	Role	CE	Т	RoleCode	CWE		
159	addr	0..*			Адрес	Role	SET<AD>	Т				
160	telecom	0..*			Телекоммуникационный адрес	Role	SET<TEL>	Т				
161	associatedPerson	0..1			Связанное лицо	Role	Person	И				
162	scoping Organization	0..1			Контролирующая организация	Role	Organization	И				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
163	inFulfillmentOf	0..*			Для выполнения направления	Act	SET<InFulfillmentOf>	H				
164	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	T	FLFS	CNE		По умолчанию: FLFS
165	order	1..1			Направление	ActRelationship	Order	H				
166	classCode	1..1	O	O2	Код роли	Act	CS	T	ACT	CNE		По умолчанию: ACT
167	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	RQO	CNE		По умолчанию: RQO
168	id	1..*		O2	Идентификатор	Act	SET<II>	T				
169	code	0..1			Код	Act	CE	T	ActCode	CWE		
170	priorityCode	0..1			Срочность	Act	CE	T	ActPriority	CWE		
171	documentationOf	0..*			Для документирования	Act	SET<DocumentationOf>	H				
172	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	T	DOC	CNE		По умолчанию: DOC
173	serviceEvent	1..1			Медицинская услуга	ActRelationship	ServiceEvent	H				
174	classCode	1..1	O	O2	Код роли	Act	CS	T	ACT	CNE		По умолчанию: ACT
175	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		По умолчанию: EVN
176	id	0..*			Идентификатор обращения	Act	SET<II>	T				
177	code	0..1			Код	Act	CE	T	ActCode	CWE		

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
178	effectiveTime	0..1			Время обращения	Act	IVL<TS>	Т				
179	performer	0..*			Исполнитель	Act	SET<Performer1>	Н				
180	typeCode	1..1	О	О2	Код типа объекта	Participation	CS	Т	x_ServiceEventPerformer	CNE		
181	functionCode	0..1			Код функции	Participation	CE	Т	ParticipationFunction	CWE		
182	time	0..1			Время участия	Participation	IVL<TS>	Т				
183	assignedEntity	1..1			Уполномоченный субъект	Participation	AssignedEntity	И				
184	relatedDocument	0..*			Связанный документ	Act	SET<RelatedDocument>	Н				
185	typeCode	1..1	О	О2	Код типа объекта	ActRelationship	CS	Т	x_ActRelationshipDocument	CNE		Родительский документ может иметь следующие связи со своими потомками: одну связь дополнения (APND); одну связь замены (RPLC); одну связь преобразования (XFRM); одну связь преобразования (XFRM) и одну связь дополнения (APND); одну связь преобразования (XFRM) и одну связь замены (RPLC)

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
186	parentDocument	1..1			Родительский документ	ActRelationship	ParentDocument	H				
187	classCode	1..1	O	O2	Класс роли	Act	CS	T	DOCCLIN	CNE		По умолчанию: DOCCLIN
188	moodCode	1..1	O	O2	Класс завершенности	Act	CS	T	EVN	CNE		По умолчанию: EVN
189	id	1..*		O2	Идентификатор	Act	SET<II>	T				
190	code	0..1			Код	Act	CD	T	DocumentType	CWE		
191	text	0..1			Текст	Act	ED	T				Элемент ParentDocument.text может быть использован, чтобы указать тип среды родительского документа (MIME Media Type). Его нельзя использовать, чтобы включить родительский документ внутрь этого элемента, поэтому элемент ParentDocument.text.BIN запрещен.
192	setId	0..1			Идентификатор группы документов	Context-Structure	II	T				
193	versionNumber	0..1			Номер версии	Context-Structure	INT	T				
194	authorization	0..*			Авторизация	Act	SET<Authorization>	H				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
195	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	T	AUTH	CNE		По умолчанию: AUTH
196	consent	1..1			Информированное согласие	ActRelationship	Consent	H				
197	classCode	1..1	O	O2	Класс роли	Act	CS	T	CONS	CNE		По умолчанию: CONS
198	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		По умолчанию: EVN
199	id	0..*			Идентификатор	Act	SET<II>	T				
200	code	0..1			Код	Act	CE	T	ActCode	CWE		
201	statusCode	1..1		O2	Код статуса	Act	CS	T	completed	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.14»
202	componentOf	0..1			Компонент обращения пациента	Act	Component1	H				
203	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	T	COMP	CNE		По умолчанию: COMP
204	encompassing Encounter	1..1			Обращение пациента	ActRelationship	EncompassingEncounter	H				
205	classCode	1..1	O	O2	Класс роли	Act	CS	T	ENC	CNE		По умолчанию: ENC
206	moodCode	1..1	O	O2	Act	Act	CS	T	EVN	CNE		По умолчанию: EVN
207	id	0..*			Идентификатор	Act	SET<II>	T				
208	code	0..1			Код	Act	CE	T	ActEncounterCode	CWE		

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
209	effectiveTime	1..1		O2	Время посещения	Act	IVL<TS>	Т				
210	discharge DispositionCode	0..1			Код места выписки	PatientEncounter	CE	Т	EncounterDischarge-Disposition	CWE		
211	responsibleParty	0..1			Ответственная сторона	Act	ResponsibleParty	Н				
212	typeCode	1..1	O	O2	Код типа объекта	Participation	CS	Т	RESP	CNE		По умолчанию: RESP
213	assignedEntity	1..1			Назначенная сторона	Participation	AssignedEntity	И				
214	encounter Participant	0..*			Участник посещения	Act	SET<Encounter-Participant>	Н				
215	typeCode	1..1	O	O2	Код типа объекта	Participation	CS	Т	x_EncounterParticipant	CNE		
216	time	0..1			Время участия	Participation	IVL<TS>	Т				
217	assignedEntity	1..1			Назначенная сторона	Participation	AssignedEntity	И				
218	location	0..1			Место посещения	Act	Location	Н				
219	typeCode	1..1	O	O2	Код типа объекта	Participation	CS	Т	LOC	CNE		По умолчанию: LOC
220	healthCareFacility	1..1			Отделение	Participation	HealthCare Facility	Н				
221	classCode	1..1	O	O2	Класс роли	Role	CS	Т	SDLOC	CNE		По умолчанию: SDLOC
222	id	0..*			Идентификатор	Role	SET<II>	Т				
223	code	0..1			Код	Role	CE	Т	ServiceDelivery LocationRoleType	CWE		

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
224	location	0..1			Место оказания медицинской помощи	Role	Place	И				
225	serviceProvider Organization	0..1			Медицинская организация	Role	Organization	И				
226	component	1..1			Компонент документа	Act	Component2	Н				
227	typeCode	1..1	О	О2	Код типа объекта	ActRelationship	CS	Т	COMP	CNE		По умолчанию: COMP
228	context ConductionInd	1..1	О	О2	Код управления контекстом	ActRelationship	BL	Т				По умолчанию: true
229	bodyChoice	1..1			Выбор типа тела документа	ActRelationship	NonXMLBody | StructuredBody	Н			Да	
230	nonXMLBody	1..1			Неструктурированный документ	Act	NonXMLBody	Н				
231	classCode	1..1	О	О2	Класс роли	Act	CS	Т	DOCBODY	CNE		По умолчанию: DOCBODY
232	moodCode	1..1	О	О2	Код завершенности	Act	CS	Т	EVN	CNE		По умолчанию: EVN
233	text	1..1			Текст	Act	ED	Т				
234	confidentialityCode	0..1			Код конфиденциальности	Act	CE	Т	x_BasicConfidentialityKind	CWE		
235	languageCode	0..1			Язык	Act	CS	Т	HumanLanguage	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.6.121»
236	structuredBody	1..1			Структурированный документ	Act	StructuredBody	Н				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
237	classCode	1..1	O	O2	Класс роли	Act	CS	T	DOCBODY	CNE		По умолчанию: DOCBODY
238	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		По умолчанию: EVN
239	confidentialityCode	0..1			Код конфиденциальности	Act	CE	T	x_BasicConfidentialityKind	CWE		
240	languageCode	0..1			Язык	Act	CS	T	HumanLanguage	CNE		По умолчанию: @codeSystem=»2.16.840.1.113883.6.121»
241	component	1..*			Структурный компонент	Act	SET<Component3>	H				
242	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	T	COMP	CNE		По умолчанию: COMP
243	context ConductionInd	1..1	O	O2	Код управления контекстом	ActRelationship	BL	T				По умолчанию: true
244	section	1..1			Раздел	ActRelationship	Section	H				
245	classCode	1..1	O	O2	Класс роли	Act	CS	T	DOCSECT	CNE		По умолчанию: DOCSECT
246	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		По умолчанию: EVN
247	id	0..1			Идентификатор	Act	II	T				
248	code	0..1			Код	Act	CE	T	DocumentSectionType	CWE		
249	title	0..1			Название	Act	ST	T				
250	text	0..1		O2	Текст	Act	ED	T				По умолчанию: @mediaType=»text/x-hl7-text+xml»

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
251	confidentialityCode	0..1			Код конфиденциальности	Act	CE	T	x_Basic ConfidentialityKind	CWE		
252	languageCode	0..1			Язык	Act	CS	T	HumanLanguage	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.6.121»
253	subject	0..1			Субъект	Act	Subject	H				
254	typeCode	1..1	O	O2	Код типа объекта	Participation	CS	T	SBJ	CNE		По умолчанию: SBJ
255	contextControlCode	1..1	O	O2	Код управления контекстом	Participation	CS	T	OP	CNE		По умолчанию: OP
256	awarenessCode	0..1			Степень озабоченности субъекта	Participation	CE	T	TargetAwareness	CWE		
257	relatedSubject	1..1			Субъект содержания раздела	Participation	RelatedSubject	H				
258	classCode	1..1	O	O2	Класс роли	Role	CS	T	x_DocumentSubject	CNE		По умолчанию: PRS
259	code	0..1			Код	Role	CE	T	PersonalRelationshipRoleType	CWE		
260	addr	0..*			Адрес	Role	SET<AD>	T				
261	telecom	0..*			Телекоммуникационный адрес	Role	SET<TEL>	T				
262	subject	0..1			Субъект	Role	SubjectPerson	H				
263	classCode	1..1	O	O2	Класс роли	Entity	CS	T	PSN	CNE		По умолчанию: PSN
264	determinerCode	1..1	O	O2	Вид объекта или экземпляр объекта	Entity	CS	T	INSTANCE	CNE		По умолчанию: INSTANCE

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
265	name	0..*			ФИО	Entity	SET<PN>	Т				
266	administrative GenderCode	0..1			Пол	Living-Subject	CE	Т	AdministrativeGender	CWE		
267	birthTime	0..1			Дата и время рождения	Living-Subject	TS	Т				
268	author	0..*			Автор	Act	SET<Author>	И				
269	informant	0..*			Информатор	Act	SET<Informant12>	И				
270	entry	0..*			Подраздел	Act	SET<Entry>	Н				
271	typeCode	1..1	О	О2	Код типа объекта	ActRelationship	CS	Т	x_ActRelationshipEntry	CNE		По умолчанию: COMP
272	context ConductionInd	1..1	О	О2	Код управления контекстом	ActRelationship	BL	Т				По умолчанию: true
273	clinicalStatement	1..1			Клиническое утверждение	ActRelationship	Act | Encounter | Observation | ObservationMedia | Organizer | Procedure | RegionOfInterest | SubstanceAdministration | Supply	Н			Да	
274	subject	0..1			Субъект	Act	Subject	И				
275	specimen	0..*			Биоматериал	Act	SET<Specimen>	Н				
276	typeCode	1..1	О	О2	Код типа объекта	Participation	CS	Т	SPC	CNE		По умолчанию: SPC
277	specimenRole	1..1			Роль биоматериала	Participation	SpecimenRole	Н				
278	classCode	1..1	О	О2	Класс роли	Role	CS	Т	SPEC	CNE		По умолчанию: SPEC

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
279	id	0..*			Идентификатор	Role	SET<II>	Т				
280	specimen PlayingEntity	0..1			Вид биоматериала	Role	PlayingEntity	Н				
281	classCode	1..1	О	О2	Класс роли	Entity	CS	Т	ENT	CNE		По умолчанию: ENT
282	determinerCode	1..1	О	О2	Вид объекта или экземпляр объекта	Entity	CS	Т	INSTANCE	CNE		По умолчанию: INSTANCE
283	code	0..1			Тип биоматериала	Entity	CE	Т	EntityCode	CWE		
284	quantity	0..*			Количество биоматериала	Entity	SET<PQ>	Т				
285	name	0..*			Наименование	Entity	SET<PN>	Т				
286	desc	0..1			Описание биоматериала	Entity	ED	Т				
287	performer	0..*			Исполнитель	Act	SET<Performer2>	Н				
288	typeCode	1..1	О	О2	Код типа объекта	Participation	CS	Т	PRF	CNE		По умолчанию: PRF
289	time	0..1			Время участия	Participation	IVL<TS>	Т				
290	modeCode	0..1			Код завершенности	Participation	CE	Т	ParticipationMode	CWE		
291	assignedEntity	1..1			Назначенная сторона	Participation	AssignedEntity	И				
292	author	0..*			Автор	Act	SET<Author>	И				
293	informant	0..*			Информатор	Act	SET<Informant12>	И				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
294	participant	0..*			Участник	Act	SET<Participant2>	H				
295	typeCode	1..1	O	O2	Код типа объекта	Participation	CS	T	ParticipationType	CNE		
296	contextControlCode	1..1	O	O2	Код управления контекстом	Participation	CS	T	OP	CNE		По умолчанию: OP
297	time	0..1			Время участия	Participation	IVL<TS>	T				
298	awarenessCode	0..1			Степень озабоченности субъекта	Participation	CE	T	TargetAwareness	CWE		
299	participantRole	1..1			Роль участника	Participation	ParticipantRole	H				
300	classCode	1..1	O	O2	Класс роли	Role	CS	T	ROL	CNE		
301	id	0..*			Идентификатор	Role	SET<II>	T				
302	code	0..1			Код	Role	CE	T	RoleCode	CWE		
303	addr	0..*			Адрес	Role	SET<AD>	T				
304	telecom	0..*			Телекоммуникационный адрес	Role	SET<TEL>	T				
305	playingEntityChoice	0..1			Выбор типа участника	Role	Device | PlayingEntity	H			Да	
306	playingDevice	1..1			Устройство	ManufacturedMaterial	Device	H				
307	classCode	1..1	O	O2	Класс роли	Entity	CS	T	DEV	CNE		По умолчанию: DEV

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
308	determinerCode	1..1	O	O2	Вид объекта или экземпляр объекта	Entity	CS	T	INSTANCE	CNE		По умолчанию: INSTANCE
309	code	0..1			Код	Entity	CE	T	EntityCode	CWE		
310	manufacturer ModelName	0..1			Производитель	Device	SC	T				
311	softwareName	0..1			Наименование программного обеспечения	Device	SC	T				
312	playingEntity	1..1			Участвующий объект	Entity	PlayingEntity	И				
313	scopingEntity	0..1			Ответственная сторона	Role	Entity	Н				
314	classCode	1..1	O	O2	Класс роли	Entity	CS	T	ENT	CNE		По умолчанию: ENT
315	determinerCode	1..1	O	O2	Вид объекта или экземпляр объекта	Entity	CS	T	INSTANCE	CNE		По умолчанию: INSTANCE
316	id	0..*			Идентификатор	Entity	SET<II>	T				
317	code	0..1			Код	Entity	CE	T	EntityCode	CWE		
318	desc	0..1			Описание	Entity	ED	T				
319	entryRelationship	0..*			Связь с компонентом подраздела	Act	SET<EntryRelationship>	Н				
320	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	T	x_ActRelationshipEntryRelationship	CNE		
321	inversionInd	0..1			Признак обращения связи	ActRelationship	BL	T				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
322	context ConductionInd	1..1	O	O2	Код управления контекстом	ActRelationship	BL	T				По умолчанию: true
323	sequenceNumber	0..1			Порядковый номер	ActRelationship	INT	T				
324	negationInd	0..1			Признак отрицания	ActRelationship	BL	T				
325	seperatableInd	0..1			Признак независимого использования	ActRelationship	BL	T				
326	clinicalStatement	1..1			Клиническое утверждение	ActRelationship	clinicalStatement | Act | Encounter | Observation | ObservationMedia | Organizer | Procedure | RegionOfInterest | SubstanceAdministration | Supply	P			Да	
327	reference	0..*			Ссылка	Act	SET<Reference>	H				
328	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	T	x_ActRelationshipExternalReference	CNE		
329	seperatableInd	0..1			Признак независимого использования	ActRelationship	BL	T				
330	externalActChoice	1..1			Выбор типа внешнего действия	ActRelationship	ExternalAct | ExternalObservation | ExternalProcedure | ExternalDocument	H			Да	
331	externalAct	1..1			Внешнее действие	Act	ExternalAct	H				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
332	classCode	1..1	O	O2	Класс роли	Act	CS	T	ACT	CNE		По умолчанию: ACT
333	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		По умолчанию: EVN
334	id	0..*			Идентификатор	Act	SET<II>	T				
335	code	0..1			Код	Act	CD	T	ActCode	CWE		
336	text	0..1			Текст	Act	ED	T				
337	externalObservation	1..1			Внешнее исследование	Act	External Observation	H				
338	classCode	1..1	O	O2	Класс роли	Act	CS	T	OBS	CNE		По умолчанию: OBS
339	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		По умолчанию: EVN
340	id	0..*			Идентификатор	Act	SET<II>	T				
341	code	0..1			Код	Act	CD	T	ActCode	CWE		
342	text	0..1			Текст	Act	ED	T				
343	externalProcedure	1..1			Внешняя процедура	Act	External Procedure	H				
344	classCode	1..1	O	O2	Класс роли	Act	CS	T	PROC	CNE		По умолчанию: PROC
345	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		По умолчанию: EVN
346	id	0..*			Идентификатор	Act	SET<II>	T				
347	code	0..1			Код	Act	CD	T	ActCode	CWE		
348	text	0..1			Текст	Act	ED	T				
349	externalDocument	1..1			Внешний документ	Context-Structure	External Document	H				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
350	classCode	1..1	O	O2	Класс роли	Act	CS	T	DOC	CNE		По умолчанию: DOC
351	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		По умолчанию: EVN
352	id	0..*			Идентификатор	Act	SET<II>	T				
353	code	0..1			Код	Act	CD	T	DocumentType	CWE		
354	text	0..1			Текст	Act	ED	T				
355	setId	0..1			Идентификатор группы документов	Context-Structure	II	T				
356	versionNumber	0..1			Номер версии	Context-Structure	INT	T				
357	precondition	0..*			Условие выполнения	Act	SET<Precondition>	H				
358	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	T	PRCN	CNE		По умолчанию: PRCN
359	criterion	1..1			Критерий	ActRelationship	Criterion	H				
360	classCode	1..1	O	O2	Класс роли	Act	CS	T	OBS	CNE		По умолчанию: OBS
361	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN.CRT	CNE		По умолчанию: EVN.CRT
362	code	0..1			Код	Act	CD	T	ActCode	CWE		
363	text	0..1			Текст	Act	ED	T				
364	value	0..1			Значение	Observation	ANY	T				
365	act	1..1			Действие	Act	Act	H				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
366	classCode	1..1	O	O2	Класс роли	Act	CS	T	x_ActClassDocumentEntryAct	CNE		
367	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	x_DocumentActMood	CNE		
368	id	0..*			Идентификатор	Act	SET<II>	T				
369	code	1..1		O2	Код	Act	CD	T	ActCode	CWE		
370	negationInd	0..1			Признак отрицания	Act	BL	T				
371	text	0..1			Текст	Act	ED	T				
372	statusCode	0..1			Код статуса	Act	CS	T	ActStatus	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.14»
373	effectiveTime	0..1			Время исполнения	Act	IVL<TS>	T				
374	priorityCode	0..1			Срочность	Act	CE	T	ActPriority	CWE		
375	languageCode	0..1			Язык	Act	CS	T	HumanLanguage	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.6.121»
376	encounter	1..1			Обращение пациента	Act	Encounter	H				
377	classCode	1..1	O	O2	Класс роли	Act	CS	T	ENC	CNE		
378	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	x_DocumentEncounterMood	CNE		
379	id	0..*			Идентификатор	Act	SET<II>	T				
380	code	0..1			Код	Act	CD	T	ActEncounterCode	CWE		
381	text	0..1			Текст	Act	ED	T				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
382	statusCode	0..1			Код статуса	Act	CS	T	ActStatus	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.14»
383	effectiveTime	0..1			Время обращения	Act	IVL<TS>	T				
384	priorityCode	0..1			Срочность	Act	CE	T	ActPriority	CWE		
385	observation	1..1			Исследование	Act	Observation	H				DesignNote: Observation.value has cardinality [0..*], which doesn't show up in the Visio representation.
386	classCode	1..1	O	O2	Класс роли	Act	CS	T	OBS	CNE		
387	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	x_ActMoodDocumentObservation	CNE		
388	id	0..*			Идентификатор	Act	SET<II>	T				
389	code	1..1		O2	Код	Act	CD	T	ObservationType	CWE		
390	negationInd	0..1			Признак отрицания	Act	BL	T				
391	derivationExpr	0..1			Метод вычислений	Act	ST	T				
392	text	0..1			Текст	Act	ED	T				
393	statusCode	0..1			Код статуса	Act	CS	T	ActStatus	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.14»
394	effectiveTime	0..1			Время исследования	Act	IVL<TS>	T				
395	priorityCode	0..1			Срочность	Act	CE	T	ActPriority	CWE		

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
396	repeatNumber	0..1			Номер повторения	Act	IVL<INT>	T				
397	languageCode	0..1			Язык	Act	CS	T	HumanLanguage	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.6.121»
398	value	0..*			Значение	Observation	SET<ANY>	T				
399	interpretationCode	0..*			Код интерпретации	Observation	SET<CE>	T	ObservationInterpretation	CNE		
400	methodCode	0..*			Код метода	Observation	SET<CE>	T	ObservationMethod	CWE		
401	targetSiteCode	0..*			Код исследуемого места	Observation	SET<CD>	T	ActSite	CWE		
402	referenceRange	0..*			Референтыний предел	Act	SET<ReferenceRange>	H				
403	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	T	REFV	CNE		По умолчанию: REFV
404	observationRange	1..1			Предельное значение	ActRelationship	ObservationRange	H				
405	classCode	1..1	O	O2	Класс роли	Act	CS	T	OBS	CNE		По умолчанию: OBS
406	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN.CRT	CNE		По умолчанию: EVN.CRT
407	code	0..1			Код	Act	CD	T	ActCode	CWE		
408	text	0..1			Текст	Act	ED	T				
409	value	0..1			Значение	Observation	ANY	T				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
410	interpretationCode	0..1			Код интерпретации	Observation	CE	T	ObservationInterpretation	CNE		
411	observationMedia	1..1			Мультимедийные данные	Act	Observation Media	H				
412	classCode	1..1	O	O2	Класс роли	Act	CS	T	OBS	CNE		
413	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		
414	id	0..*			Идентификатор	Act	SET<II>	T				
415	languageCode	0..1			Язык	Act	CS	T	HumanLanguage	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.6.121»
416	value	1..1		O2	Значение	Observation	ED	T				
417	organizer	1..1			Органайзер	Act	Organizer	H				Элемент organizer может быть источником ссылки на другие компоненты подраздела или внешние действия, но не мо-жет содержать ссылку entryRelationship
418	classCode	1..1	O	O2	Класс роли	Act	CS	T	x_ActClassDocumentEntryOrganizer	CNE		
419	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		
420	id	0..*			Идентификатор	Act	SET<II>	T				
421	code	0..1			Код	Act	CD	T	ActCode	CWE		

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
422	statusCode	1..1		O2	Код статуса	Act	CS	T	ActStatus	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.14»
423	effectiveTime	0..1			Время действия	Act	IVL<TS>	T				
424	component	0..*			Компонент подраздела	Act	SET<Component4>	H				
425	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	T	COMP	CNE		По умолчанию: COMP
426	context ConductionInd	1..1	O	O2	Код управления контекстом	ActRelationship	BL	T				По умолчанию: true
427	sequenceNumber	0..1			Порядковый номер	ActRelationship	INT	T				
428	seperatableInd	0..1			Признак независимого использования	ActRelationship	BL	T				
429	clinicalStatement	1..1			Клиническое утверждение	ActRelationship	clinicalStatement | Act | Encounter | Observation | ObservationMedia | Organizer | Procedure | RegionOfInterest | SubstanceAdministration | Supply	P			Да	
430	procedure	1..1			Процедура	Act	Procedure	H				
431	classCode	1..1	O	O2	Класс роли	Act	CS	T	PROC	CNE		
432	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	x_Document ProcedureMood	CNE		

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
433	id	0..*			Идентификатор	Act	SET<II>	T				
434	code	0..1			Код	Act	CD	T	ActCode	CWE		
435	negationInd	0..1			Признак отрицания	Act	BL	T				
436	text	0..1			Текст	Act	ED	T				
437	statusCode	0..1			Код статуса	Act	CS	T	ActStatus	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.14»
438	effectiveTime	0..1			Время исполнения	Act	IVL<TS>	T				
439	priorityCode	0..1			Срочность	Act	CE	T	ActPriority	CWE		
440	languageCode	0..1			Язык	Act	CS	T	HumanLanguage	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.6.121»
441	methodCode	0..*			Код метода	Procedure	SET<CE>	T	ProcedureMethod	CWE		
442	approachSiteCode	0..*			Место применения	Procedure	SET<CD>	T	ActSite	CWE		
443	targetSiteCode	0..*			Место цели применения	Procedure	SET<CD>	T	ActSite	CWE		
444	regionOfInterest	1..1			Область интереса	Act	RegionOfInterest	H				
445	classCode	1..1	O	O2	Класс роли	Act	CS	T	ROIOVL	CNE		
446	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	EVN	CNE		
447	id	1..*		O2	Идентификатор	Act	SET<II>	T				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
448	code	1..1		O2	Код	Act	CS	T	ROIOverlayShape	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.4»
449	value	1..*		O2	Значение	Observation	LIST<INT>	T				
450	substance Administration	1..1			Применение субстанции	Act	SubstanceAdministration	H				
451	classCode	1..1	O	O2	Класс роли	Act	CS	T	SBADM	CNE		
452	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	x_DocumentSubstanceMood	CNE		
453	id	0..*			Идентификатор	Act	SET<II>	T				
454	code	0..1			Код	Act	CD	T	SubstanceAdministrationActCode	CWE		
455	negationInd	0..1			Признак отрицания	Act	BL	T				
456	text	0..1			Текст	Act	ED	T				
457	statusCode	0..1			Код статуса	Act	CS	T	ActStatus	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.14»
458	effectiveTime	0..1			Продолжительность или частота применения	Act	GTS	T				
459	priorityCode	0..1			Срочность	Act	CE	T	ActPriority	CWE		
460	repeatNumber	0..1			Число повторений	Act	IVL<INT>	T				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
461	routeCode	0..1			Способ применения	Substance Administration	CE	T	RouteOfAdministration	CWE		
462	approachSiteCode	0..*			Место применения	Substance Administration	SET<CD>	T	ActSite	CWE		
463	doseQuantity	0..1			Дозировка	Substance Administration	IVL<PQ>	T				
464	rateQuantity	0..1			Скорость введения	Substance Administration	IVL<PQ>	T				
465	maxDoseQuantity	0..1			Максимальная доза	Substance Administration	RTO<PQ,PQ>	T				
466	administration UnitCode	0..1			Единицы применения	Substance Administration	CE	T	AdministrableDrugForm	CWE		
467	consumable	1..1			Способ изготовления	Act	Consumable	H				
468	typeCode	1..1	O	O2	Код типа объекта	Participation	CS	T	CSM	CNE		По умолчанию: CSM
469	manufactured Product	1..1			Готовая форма	Participation	ManufacturedProduct	H				
470	classCode	1..1	O	O2	Класс роли	Role	CS	T	MANU	CNE		По умолчанию: MANU

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
471	id	0..*			Идентификатор	Role	SET<II>	Т				
472	manufactured DrugOrOther Material	1..1			Выбор готового лекарства или другого материала	Role	LabeledDrug | Material	Н			Да	
473	manufactured LabeledDrug	1..1			Готовое лекарство	Material	LabeledDrug	Н				
474	classCode	1..1	О	О2	Класс роли	Entity	CS	Т	MMAT	CNE		По умолчанию: MMAT
475	determinerCode	1..1	О	О2	Вид объекта или экземпляр объекта	Entity	CS	Т	KIND	CNE		По умолчанию: KIND
476	code	0..1			Код	Entity	CE	Т	DrugEntity	CWE		
477	name	0..1			Наименование	Entity	EN	Т				
478	manufactured Material	1..1			Готовый материал	Material	Material	Н				
479	classCode	1..1	О	О2	Класс роли	Entity	CS	Т	MMAT	CNE		По умолчанию: MMAT
480	determinerCode	1..1	О	О2	Вид объекта или экземпляр объекта	Entity	CS	Т	KIND	CNE		По умолчанию: KIND
481	code	0..1			Код	Entity	CE	Т	MaterialEntityClass-Type	CWE		
482	name	0..1			Наименование	Entity	EN	Т				
483	lotNumberText	0..1			Текстовый номер партии	ManufacturedMaterial	ST	Т				
484	manufacturer Organization	0..1			Производитель	Role	Organization	И				

Продолжение таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
485	supply	1..1			Отпуск лекарства или материала	Act	Supply	H				
486	classCode	1..1	O	O2	Класс роли	Act	CS	T	SPLY	CNE		
487	moodCode	1..1	O	O2	Код завершенности	Act	CS	T	x_DocumentSubstanceMood	CNE		
488	id	0..*			Идентификатор	Act	SET<II>	T				
489	code	0..1			Код	Act	CD	T	ActCode	CWE		
490	text	0..1			Текст	Act	ED	T				
491	statusCode	0..1			Код статуса	Act	CS	T	ActStatus	CNE		По умолчанию: @codeSystem=» 2.16.840.1. 113883.5.14»
492	effectiveTime	0..1			Время отпуска	Act	GTS	T				
493	priorityCode	0..*			Срочность	Act	SET<CE>	T	ActPriority	CWE		
494	repeatNumber	0..1			Номер повторения	Act	IVL<INT>	T				
495	independentInd	0..1			Признак независимого использования	Act	BL	T				
496	quantity	0..1			Количество	Supply	PQ	T				
497	expectedUseTime	0..1			Ожидаемое время применения	Supply	IVL<TS>	T				
498	product	0..1		O2	Лекарство или субстанция	Act	Product	H				
499	typeCode	1..1	O	O2	Код типа объекта	Participation	CS	T	PRD	CNE		По умолчанию: PRD

Окончание таблицы 151

№	Имя информационного объекта	Кратность	Обязательность	Соответствие	Описание	Класс RIM	Тип данных	Источник	Домен	Степень кодирования	Абстрактный	Примечания
500	manufactured Product	1..1		O2	Произведенный продукт	Participation	Manufactured-Product	И				
501	component	0..*			Компонент документа	Act	SET<Component5>	Н				
502	typeCode	1..1	O	O2	Код типа объекта	ActRelationship	CS	Т	COMP	CNE		По умолчанию: COMP
503	context ConductionInd	1..1	O	O2	ActRelationship	ActRelationship	BL	Т				По умолчанию: true
504	section	1..1			Раздел	ActRelationship	Section	Р				

6.2 Графическое представление модели АКД R-MIM

Графическое представление модели АКД R-MIM показано на рисунке 5.

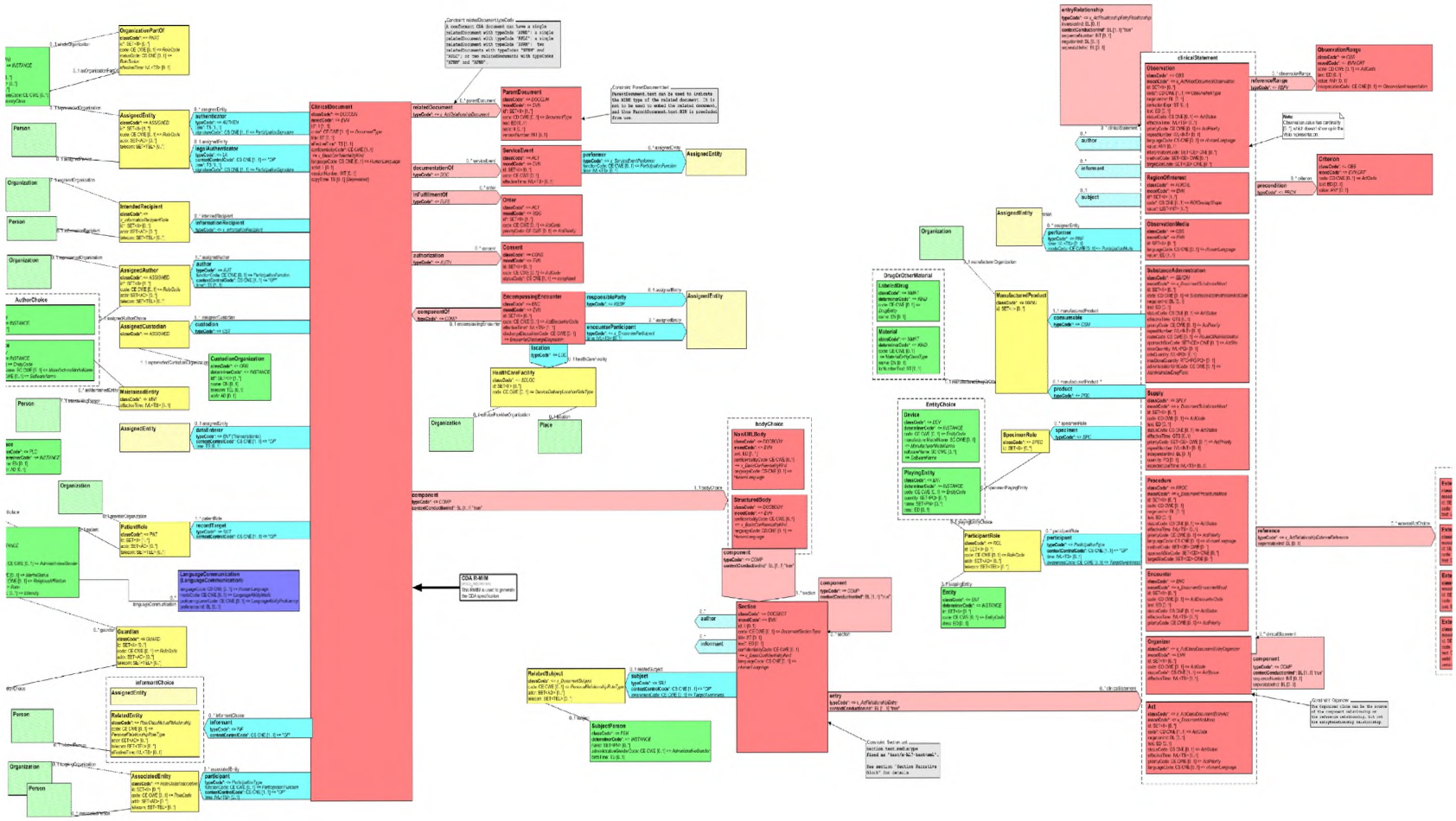


Рисунок 5 — Графическое представление модели АКД RMIM

Приложение А
(справочное)

Эталонная информационная модель HL7

Версия: январь 2009 г.
Ответственная рабочая группа комитета HL7: Modeling and Methodology Work Group.

А.1 Введение

А.1.1 Примечание для голосования в ИСО

Эта версия модели RIM предназначена для обеспечения интерпретации стандарта ISO 27932 — HL7 Clinical Document Architecture, Release 2 (Архитектура клинических документов HL7, выпуск 2). Эта модель HL7 RIM версии 2.07 служит основой стандарта Архитектуры клинических документов (АКД). Она произведена от стандарта ИСО/ HL7 21731:2006.

А.1.2 Основные сведения

Эталонная информационная модель RIM (Reference Information Model) комитета Health Level Seven (HL7) представляет собой статическую модель информации о здоровье и медицинской помощи в соответствии с областями применения деятельности по развитию стандартов HL7. Она является информационной точкой зрения, согласованной комитетом HL7 и его национальными филиалами. RIM служит основным источником, из которого производится все содержание стандартных спецификаций протокола HL7 версии 3, связанное с информацией.

А.1.2.1 История разработки модели RIM

Развитие модели RIM началось в апреле 1996 г. Первый выпуск модели RIM был принят Техническим руководящим комитетом HL7 (Technical Steering Committee) на заседании рабочей группы в январе 1997 г. Этот выпуск был известен как проект RIM версии 0.80.

На следующих двух заседаниях рабочей группы основное внимание уделялось ознакомлению с проектом модели RIM и реализации процесса получения и согласования предложений по развитию модели. Процесс сопровождения модели RIM стал известен как «гармонизация модели RIM». Между 1998 г. и 2000 г. было девять заседаний, посвященных гармонизации, на которых рассматривались предложения по улучшения проекта RIM. Кульминацией этой работы стала модель HL7 RIM версии 1.0, обнародованная на заседании рабочей группы в январе 2001 г. На заседаниях по гармонизации основное внимание уделялось следующим пяти основным темам:

- гарантия охвата стандарта HL7 версии 2.x. Этот набор предложенных изменений привносил в проект модели все информационное содержание стандарта HL7 версии 2.x;
- устранение из модели содержания, не имеющего реализации. Этот набор предложенных изменений концентрировался на удалении той информации из проекта модели, которую проблемный технический комитет (steward technical committee) не инициировал и для сохранения которой в стандарте не мог найти достаточных оснований;
- объединенная модель обслуживания. Этот набор предложенных изменений представлял краткий, точно определенный набор структур и словарей, отвечавший информационным потребностям широкого круга клинических сценариев. Эта коллекция предложений, известная как USAM (Unified service action model), явилась плодом объединенных усилий ряда технических комитетов;
- обеспечения качества. Этот набор предложенных изменений был предназначен для устранения несогласованностей проекта модели и конфликтов между моделью и руководством по стилю моделирования. Он положил начало практике записи и отслеживания открытых проблем модели;
- внимание к «левой части» модели. Этот набор предложенных изменений был предназначен для определения развитых структур и словарей, рассчитанных на неклинические части модели (управление движением пациентов, финансы, ведение расписаний). Как и объединенная модель обслуживания, эти предложения явились плодом усилий ряда технических комитетов.

Процесс гармонизации и темы особого внимания позволили получить модель RIM, которая является надежной, ясной, всеобъемлющей, лаконичной и непротиворечивой. Ее структура является и гибкой, и расширяемой, а ее содержание может быть достаточно легко отображено на стандарт передачи медицинских данных HL7 2.x и другие широко используемые спецификации структур данных в здравоохранении.

Содержание настоящего документа представляет собой выпуск 1.22 модели HL7 RIM. Тонкая подгонка этой модели продолжается по мере ее использования техническими комитетами для все более расширяющейся области применения семейства разрабатываемых стандартов HL7, основанных на применении моделей. Это семейство получило название HL7 версия 3.0. В настоящее время модель RIM является весьма стабильной. В то же время процесс гармонизации обеспечивает ее постоянную релевантность и соответствие информационным потребностям разрабатываемым стандартам HL7

А.1.2.2 Использование RIM в комитете HL7

Модель RIM является критичным компонентом процесса развития стандартов HL7 Версии 3. Она положена в основу всех информационных моделей и структур, разрабатываемых в процессе развития Версии 3.

Разработка стандартов HL7 Версии 3 ведется в соответствии с методологией, ориентированной на применение моделей, согласно которой для описания статических и динамических аспектов требований и разработки стандартов HL7, а также для описания соответствующей семантики и правил деятельности разрабатывается комплекс взаимосвязанных моделей.

Модель RIM представляет собой статическую точку зрения на информационные потребности стандартов HL7 Версии 3. Она включает в себя классы информационных объектов и диаграммы перехода состояний, дополняемые моделями вариантов использования, моделями взаимодействия, моделями типов данных, моделями терминологии и другими типами моделей, необходимых для обеспечения полной точки зрения на требования и архитектуру стандартов HL7. Классы, атрибуты, переходы состояний и ассоциации, описанные в RIM, используются для производства предметно-ориентированных информационных моделей, которые с помощью ряда процессов уточнения ограничений преобразуются в статическую модель содержания информации, используемую в стандарте HL7.

Процесс разработки стандарта HL7 Версии 3 определяет правила производства предметно-ориентированных информационных моделей из модели RIM и уточнения этих моделей в спецификациях стандартов HL7. Правила требуют, чтобы все информационные структуры, описанные в производных моделях, могли быть прослежены обратно к модели RIM и чтобы их семантика и соответствующие правила деятельности не противоречили тем, которые определены в модели RIM. Поэтому модель RIM является основным источником всего информационного содержания стандартов HL7 Версии 3.

Модель RIM используется международными филиалами HL7 для адаптации стандартов HL7 Версии 3 к местным особенностям. С помощью процесса, называемого локализацией, спецификации стандарта Версии 3 могут быть расширены, используя модель RIM как источник нового информационного содержания. Такая новая информация производится из модели RIM и уточняется тем же способом, что и применяется для создания исходной спецификации.

А.1.2.3 Использование RIM вне HL7

В основном модель RIM предназначена для использования комитетом HL7 и его международными филиалами. Однако эта модель может использоваться и вне структур комитета HL7. Первые разработчики, принявшие на вооружение методологию разработки стандартов Версии 3, использовали RIM для конструирования спецификаций сообщений, похожих на описанные в стандарте HL7, в своих собственных средах. К этим разработчикам относятся производители информационных систем, большие сети интегрированных медицинских организаций и правительственные агентства как в США, так и в других странах. Эти же самые первые разработчики крайне активны в комитете HL7 и вносят практические предложения по усовершенствованию RIM и других аспектов процесса разработки Версии 3.

Некоторые организации-участники комитета HL7 сообщили об использовании модели RIM как источника разработки корпоративной информационной архитектуры или как исходной точки для системного анализа и проектирования. Модель RIM действительно может использоваться для таких целей, однако комитет HL7 не дает никаких гарантий, что она полезна не только как эталонная модель, предназначенная для разработки стандартов HL7.

RIM — только одна модель информационных потребностей здравоохранения. Абстрактный стиль модел иRIM и возможность ее расширения с помощью спецификаций словарей данных позволяет применять модель RIM для любого мыслимого сценария обмена данными между информационными системами здравоохранения. На самом деле она концептуально применима для любой предметной области, в которой определенные сущности выполняют роли и участвуют в действиях.

Универсальная применимость модели RIM делает ее особенно полезной для организаций, подобных комитету HL7, которые должны учитывать потребности большого и разнообразного членства. Стиль модели RIM делает ее чрезвычайно стабильной, что является еще одним свойством, важным для комитета HL7. Процесс разработки стандартов HL7 состоит в создании предметно-ориентированных моделей, являющихся производными от модели RIM, и последовательного уточнения этих моделей, позволяющего получить модели, специфичные для данной предметной области. Эти специфичные модели предметной области конкретизируют модель RIM и задают ограничения значений атрибутов и ассоциаций между классами, применимые для конкретных случаев. Внешним организациям, рассматривающим возможность использование модели HL7 RIM, рекомендуется принять подобный процесс построения производных моделей в форме преобразования модели RIM.

А.1.3 Утверждение модели RIM

Утверждение модели RIM в соответствии с принятыми процедурами голосования позволило вывести эту модель за рамки внутренних артефактов комитета HL7. Как утвержденный стандарт она будет представлена для принятия в качестве стандарта Американского института стандартов (American National Standards Institute — ANSI), а затем, возможно, для принятия в качестве стандарта ИСО. Другие организации-разработчики стандартов, аккредитованные в ANSI или ИСО, в том числе организация ASC X12 и ТК 251 «Медицинская информатика» Европейского комитета по стандартизации (CEN TC 251), выразили интерес в использовании модели RIM или в ссылках на эту модель при разработке своих собственных стандартов. Появление модели RIM в качестве стандарта значительно облегчает ее использование другими организациями-разработчиками стандартов (standards development organizations — SDO). Наличие общей эталонной информационной модели, используемой разными разработчиками стандартов информатизации здоровья, сулит значительные выгоды.

Утверждение модели RIM имеет большое значение для местной стандартизации, осуществляемой национальными филиалами комитета HL7. Чтобы филиалы могли разрабатывать местные расширения стандартов HL7

в соответствии с процессом разработки стандарта HL7 Версии 3, сама модель RIM должна быть стандартом. Это нередко требуется и местными юридическими нормами, например, местные стандарты должны быть расширениями уже принятых общих стандартов.

При проведении мероприятий, в которых применимы стандарты, органы государственного управления на федеральном уровне, уровне штата и муниципальном уровне в первую очередь рассматривают аккредитованные национальные стандарты. Поскольку модель RIM стандартизует информационное содержание стандартов интероперабельности, разрабатываемых комитетом HL7, органы государственного управления США будут рассматривать ее в контексте решений, обеспечивающих интероперабельность информационных систем здравоохранения.

A.1.3.1 Нормативные части модели RIM

Модель RIM состоит из классов, включенных в один или несколько пакетов предметных областей. Атрибуты, отношения и машины состояний связаны с классами.

Каждый класс модели RIM представляет информацию о понятии, которое должно быть передано в сфере здравоохранения. Имена классов произведены от обычного (английского) языка, но по необходимости они ограничены «пространством имен» модели RIM. Значение этих классов полностью охвачено определением класса и определениями его свойств (атрибутов и ассоциаций) этого класса. Например, для понимания значения класса Role (роль) достаточно ознакомиться с его определением и определениями его свойств. В контексте пространства имен RIM не являются релевантными определения имени, взятые из другого контекста или из словаря.

Модель RIM представлена на языке UML, расширенном с помощью тегов, специфичных для стандартов HL7 и включенных в метаданные элементов UML-модели. Все стандартные значения метаданных элементов модели UML нормативны, но нормативны также и следующие расширения HL7:

- Class.stateAttribute;
- Class.classCode;
- Attribute.mandatoryInclusion;
- Attribute.cardinality;
- Attribute.vocabDomain;
- Attribute.vocabStrength.

Остальные расширения, предложенные комитетом HL7, предназначены только для целей управления и не являются нормативной частью спецификации модели RIM.

A.1.3.2 Нормативные ссылки

A.1.3.3 Смысл нормативности

A.1.4 Понимание модели RIM

В RIM использован очень абстрактный стиль моделирования. Ее ядром служат базовые классы и их структурные атрибуты. Понимание этих классов и атрибутов существенно для понимания RIM.

A.1.4.1 RIM как абстрактная модель

RIM содержит 6 «базовых» классов:

- класс Act (действие), представляющий действия, которые выполняются и должны быть документированы при оказании медицинской помощи;
- класс Participation (участие), представляющий контекст действия, а именно, кто выполнил действие, для кого оно было выполнено, где было выполнено и т. д.;
- класс Entity (сущность), представляющий физические предметы и существа, которые используются при оказании медицинской помощи и принимают участие в ее оказании;
- класс Role (роль), представляющий роли, выполняемые сущностями, участвующими в действиях по оказанию медицинской помощи;
- класс ActRelashionship (связь действий), представляющий связь одного действия с другим, например, связь между направлением на исследование и состоявшимся исследованием;
- класс RoleLink (связь ролей), представляющий отношения между отдельными ролями.

Три из этих классов — Act, Entity и Role (действия, сущности и роли) — детализируются в виде множества классов-специализаций, или подтипов. В представлении стандарта HL7 подтипы добавляются к модели RIM только в том случае, если требуется определить один или более атрибутов или ассоциаций, которые не могут быть унаследованы от родительского класса. Классы, описывающие отдельными понятия, не нуждающиеся ни в каких дальнейших атрибутах или ассоциациях, представлены исключительно как уникальный код в контролируемом словаре. Поэтому эти три базовых класса имеют следующие кодируемые атрибуты, используемые для дальнейшего определения моделируемого понятия:

- classCode (в классах Act, Entity и Role), уточняющий назначение класса или соответствующего ему понятия независимо от того, представлен ли он как класс в иерархии RIM;
- moodCode (в классе Act) и determinerCode (в классе Entity), с помощью которых можно указать, представляет ли класс экземпляр или разновидность класса Act или Entity. Если класс является специализацией класса Act, то атрибут moodCode позволяет указать, описывает ли экземпляр класса Act свершившееся или планируемое действие;
- code (в классах Act, Entity и Role), обеспечивающий более детальную классификацию значения атрибута classCode, например, конкретный вид исследования в классе Observation (исследование).

Другие три базовых класса модели RIM — Participation, ActRelationship и RoleLink — не представлены иерархиями обобщения-специализации. Тем не менее эти классы представляют разнообразные понятия, например, разные формы участия или разные типы отношений между действиями. Эти отличия представлены атрибутом typeCode, который должен быть определен в каждом из этих классов.

А.1.4.2 Представление структуры класса в модели RIM

Как упоминалось ранее, модель RIM сконструирована с использованием подмножества семантики языка UML. Она представляет собой набор UML-классов, содержащих один или более атрибутов, которым присвоены типы данных, основанные на независимой спецификации типов данных Версии 3. Классы связаны рядом отношений ассоциации, идентифицируемых уникальными именами ролей, или отношениями обобщения-специализации.

Каждый из этих элементов имеет текстовое определение. Внешнее представление атрибутов и ассоциаций управляется кратностью и другими связанными ограничениями, применяемыми к атрибутам и ролям, привязывающим ассоциации к классам.

А.1.4.3 Представление контролируемого словаря

Ряд атрибутов в модели RIM имеют тип данных CS. Это означает, что множество значений такого атрибута должно быть выбрано из множества кодов, определенных в стандарте HL7. Упомянутые ранее атрибуты classCode и typeCode являются примерами атрибутов с типом данных CS.

Все множества кодированных значений этих атрибутов являются частью настоящего документа и принимаются в соответствии с теми же принципами голосования, что и классы модели RIM. Каждое множество кодов представлено как словарный домен, т. е. множество всех понятий, которые могут использоваться как допустимые значения кодированного поля или атрибута. Важно отметить, что словарный домен состоит из множества понятий, а не слов или кодов.

А.1.4.4 Связанные спецификации

Как отмечено ранее, каждому атрибуту в модели RIM присвоен тип данных. Формальной спецификацией этих типов данных служат нормативная спецификация «HL7 V3 Data Types Implementable Technology Specification for XML» (Реализуемая технологическая спецификация типов данных Версии 3 для XML) и справочный документ «HL7 Data Types Abstract Specification» (Абстрактная спецификация типов данных HL7. Оба этих документа в настоящее время находятся в процессе утверждения комитетом HL7. Справочная таблица свойств релевантных типов данных включена в Приложение В.

А.1.5 Спецификация модели RIM и диаграммы

А.1.5.1 Предметные области и классы модели RIM

Содержание следующих разделов настоящего документа составляет спецификацию предметных областей и классов модели RIM. Отдельные предметные области и классы «помечены» пиктограммами, обозначающими, какие элементы являются нормативными, а какие — справочными. (Интерпретацию этих пиктограмм можно найти в сопутствующих документах.)

А.1.5.2 Нормативный словарь структурных атрибутов модели RIM

Словарные домены HL7 детализированы в отдельном документе. Избранное подмножество этих таблиц является частью нормативной спецификации модели RIM. Оно образовано из таблиц значений структурных атрибутов (тех, что имеют тип данных «CS») нормативных классов. В спецификации модели RIM приводятся гиперссылки на эти таблицы в тех местах, где они упоминаются. Кроме того, ниже приведен полный перечень ссылок на эти таблицы.

ActClass	ActStatus	ParticipationType
ActMood	ContextControl	RelationshipConjunction
ActRelationshipCheckpoint	EntityClass	RoleClass
ActRelationshipJoin	EntityDeterminer	RoleLinkType
ActRelationshipSplit	EntityStatus	RoleStatus
ActRelationshipType	ManagedParticipationStatus	

А.1.5.3 Общие диаграммы модели RIM

Модель RIM представлена диаграммами классов для большинства предметных областей и диаграммами состояний перехода экземпляров классов. Кроме того, полная диаграмма классов модели RIM предоставлено в виде «плаката» в файле формата PDF, обеспечивающего удобные возможности ее увеличения. Ниже приведены диаграммы классов для отдельных предметных областей и диаграммы перехода состояний.

А.1.5.4 Графические диаграммы нормативного содержания модели RIM

Диаграммы классов, включенных в нормативное содержание RIM, представлены на рисунках А.—А.4.

А.1.5.4.1 Предметная область FoundationClasses

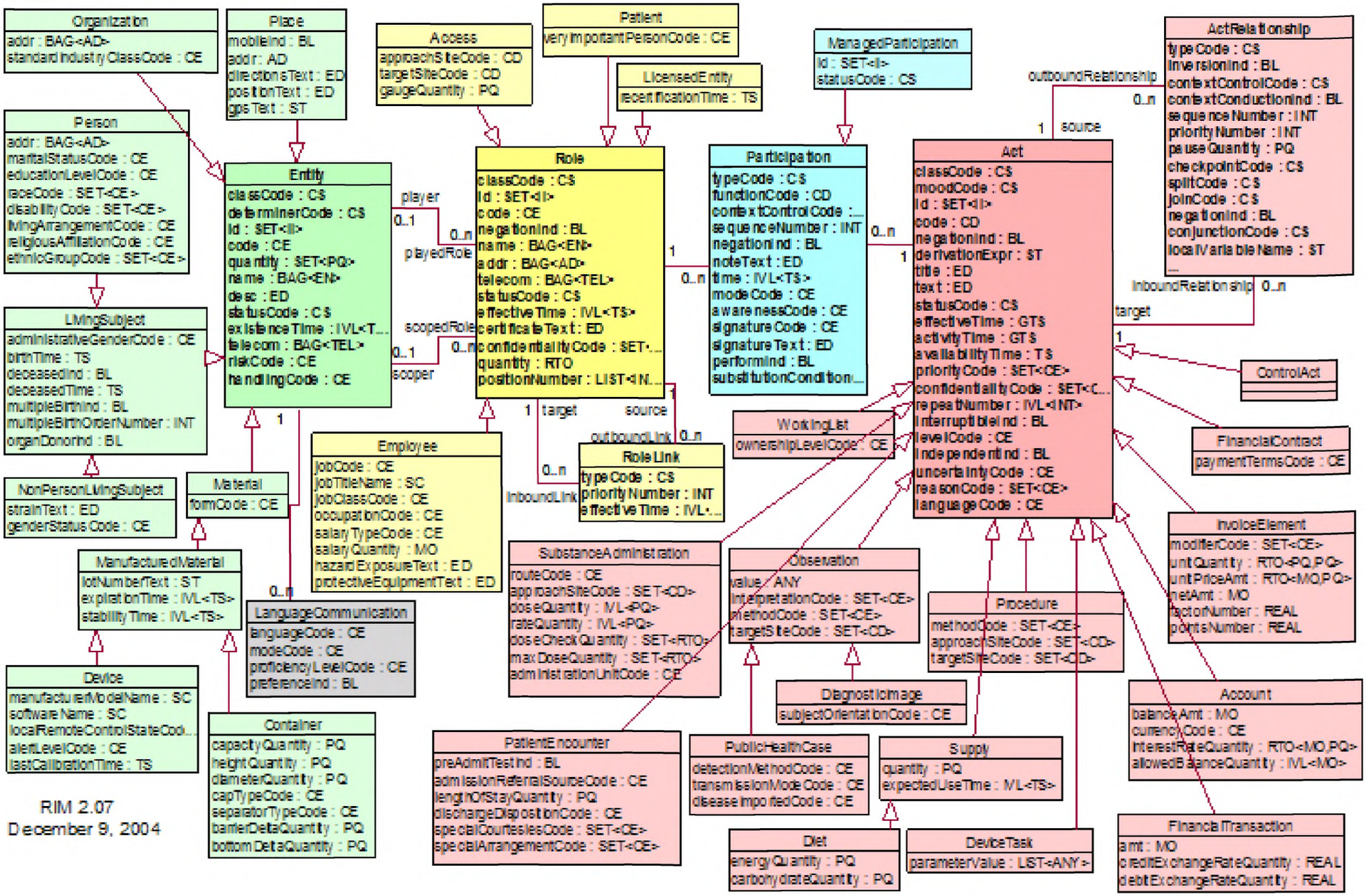


Рисунок А.1 — Диаграмма классов предметной области FoundationClasses

A.1.5.4.2 Предметная область Acts (действия)

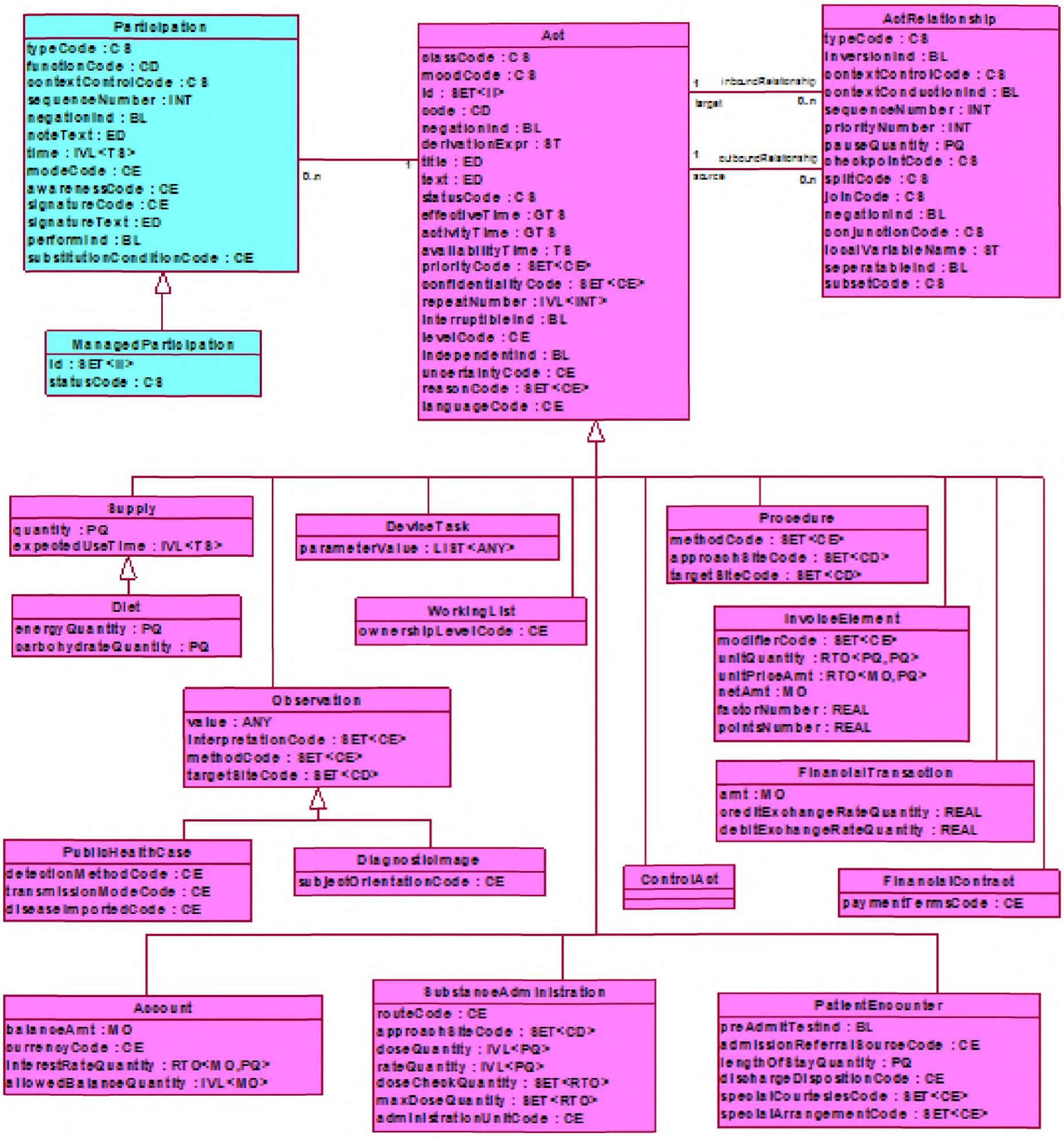


Рисунок А.2 — Диаграмма классов предметной области Acts (действия)

А.1.5.4.3 Предметная область Entities (сущности)

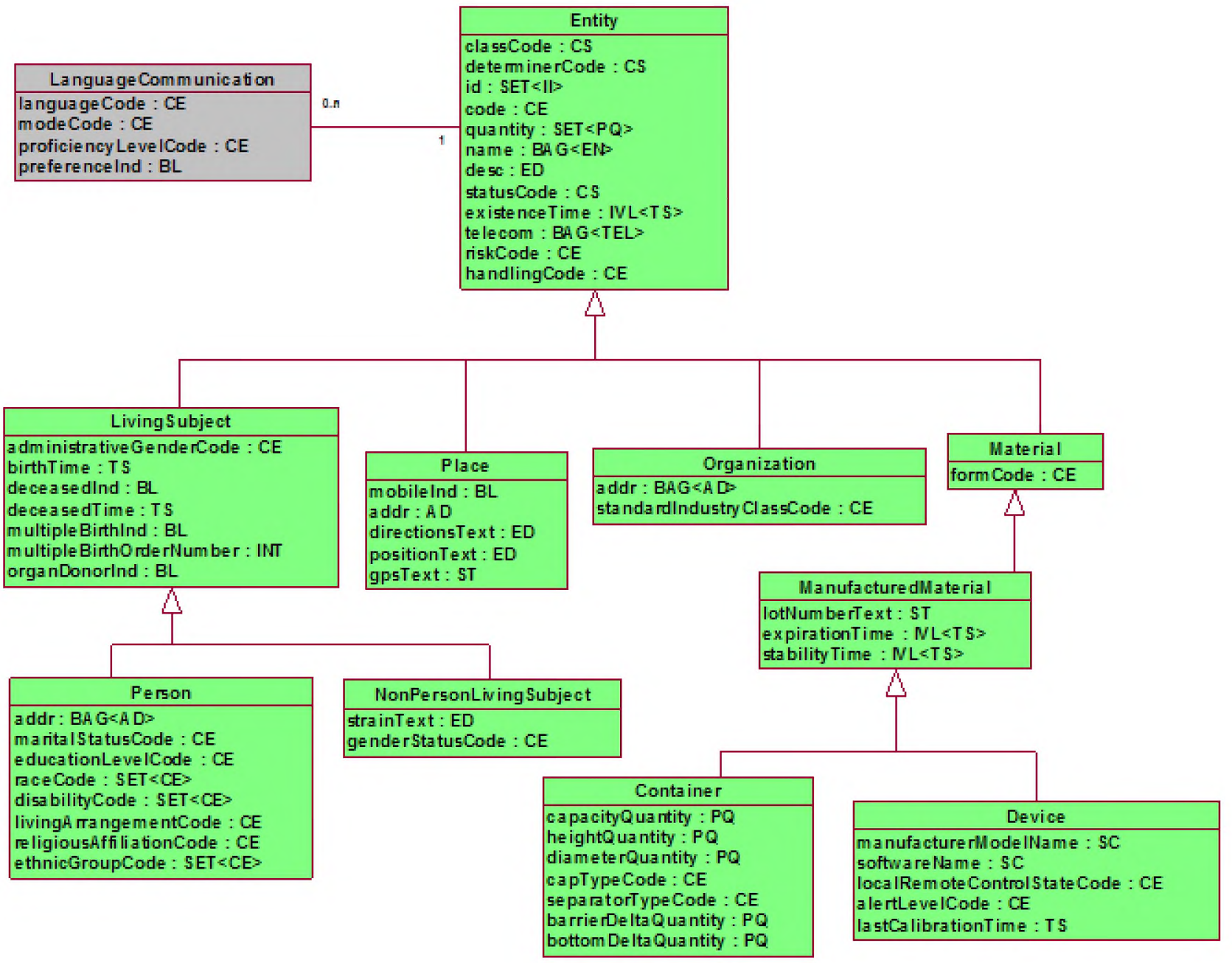


Рисунок А.3 — Диаграмма классов предметной области Entities (сущности)

A.1.5.4.4 Предметная область Roles (роли)

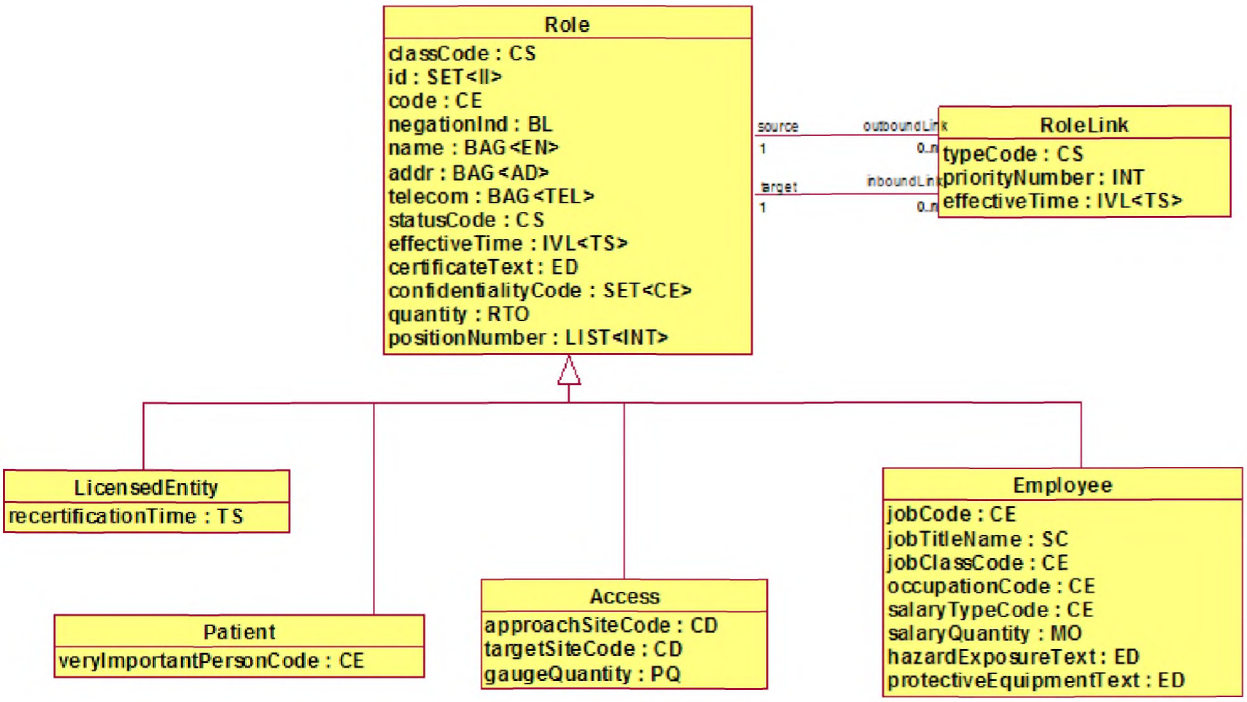


Рисунок А.4 — Диаграмма классов предметной области Roles (роли)

A.1.5.4.5 Диаграмма перехода состояний класса Act (действие)
Диаграмма перехода состояний класса Act (действие) показана на рисунке А.5.

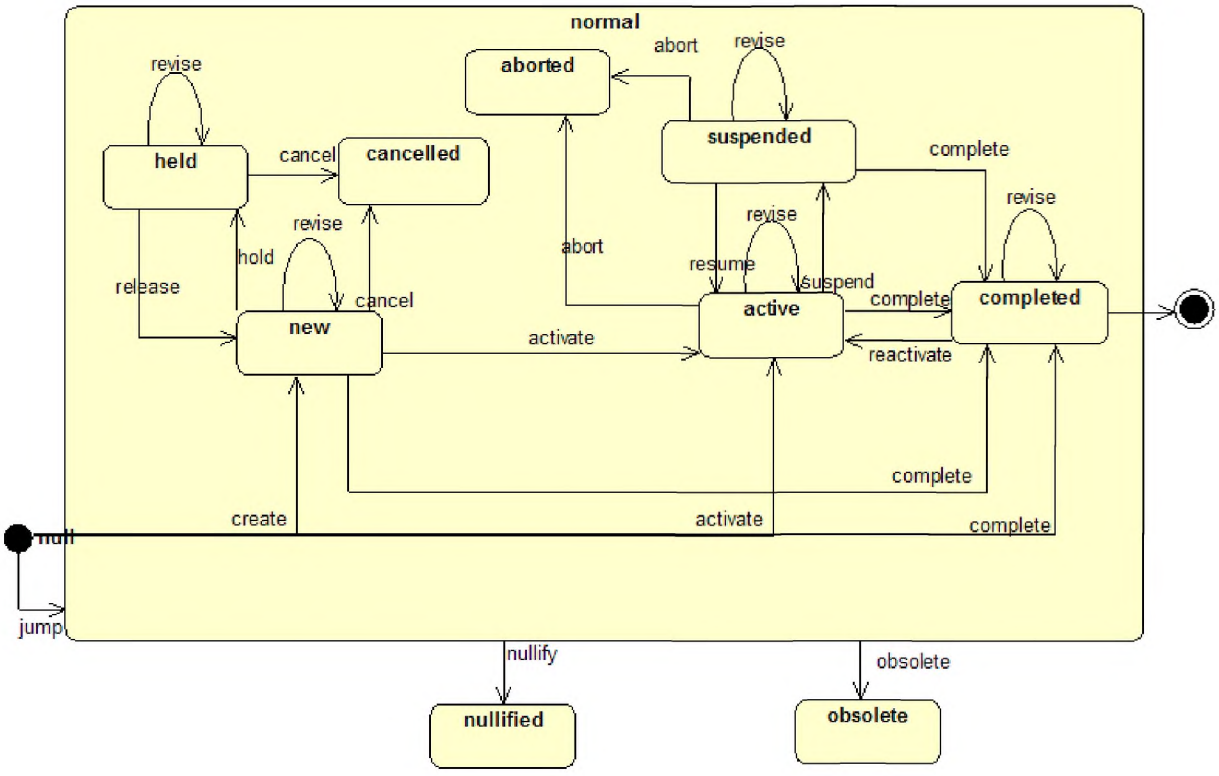


Рисунок А.5 — Диаграмма перехода состояний класса Act (действие)

А.1.5.4.6 Диаграмма перехода состояний класса Entity (сущность)
Диаграмма перехода состояний класса Entity (сущность) показана на рисунке А.6.

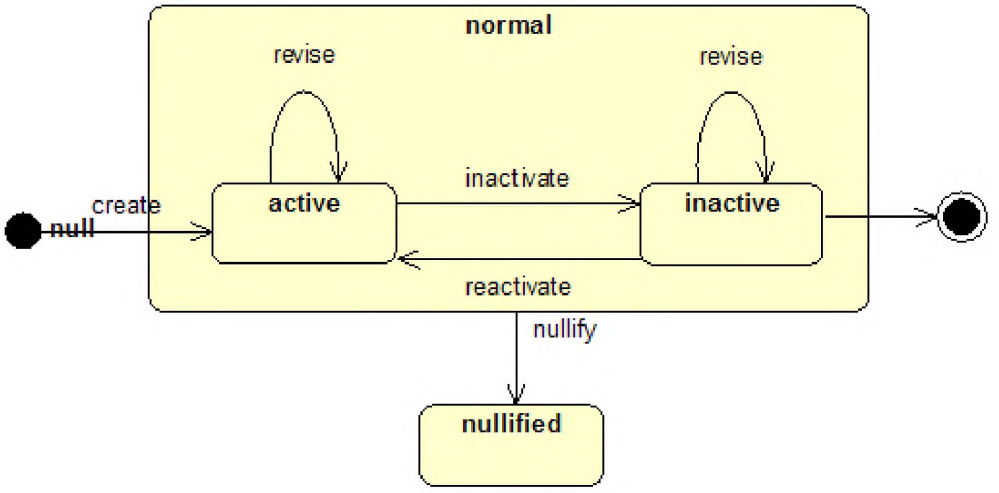


Рисунок А.6 — Диаграмма перехода состояний класса Entity (сущность)

А.1.5.4.7 Диаграмма перехода состояний класса ManagedParticipation (управляемое участие)
Диаграмма перехода состояний класса ManagedParticipation (управляемое участие) показана на рисунке А.7.

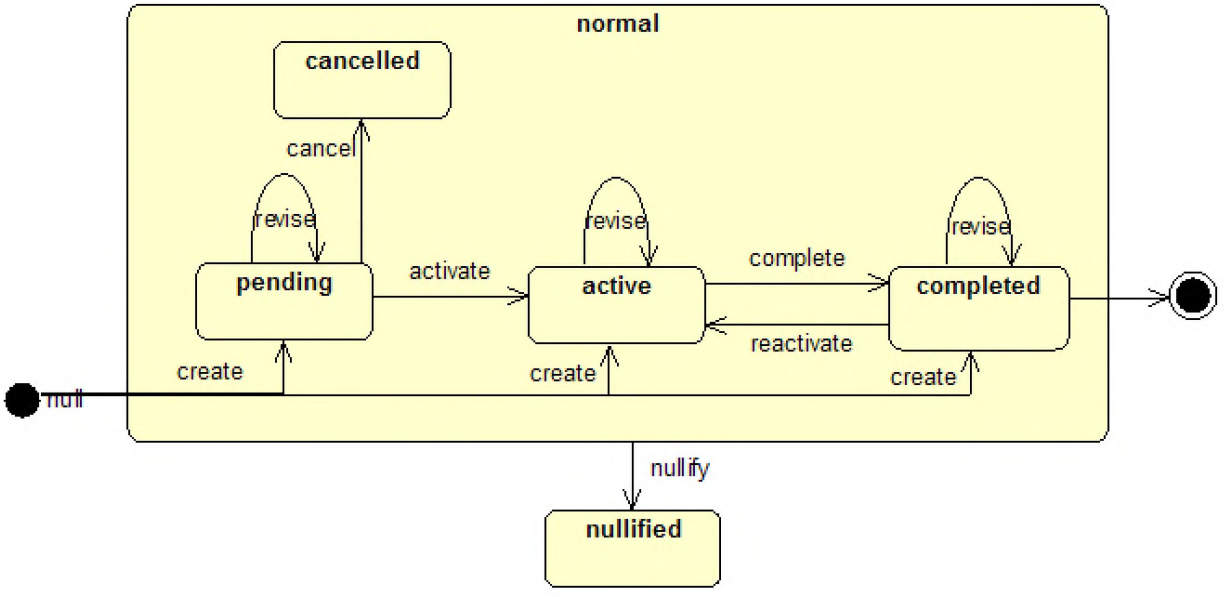


Рисунок А.7 — Диаграмма перехода состояний класса ManagedParticipation
(управляемое участие)

A.1.5.4.8 Диаграмма перехода состояний класса Role (роль)
Диаграмма перехода состояний класса Role (роль) показана на рисунке A.8.

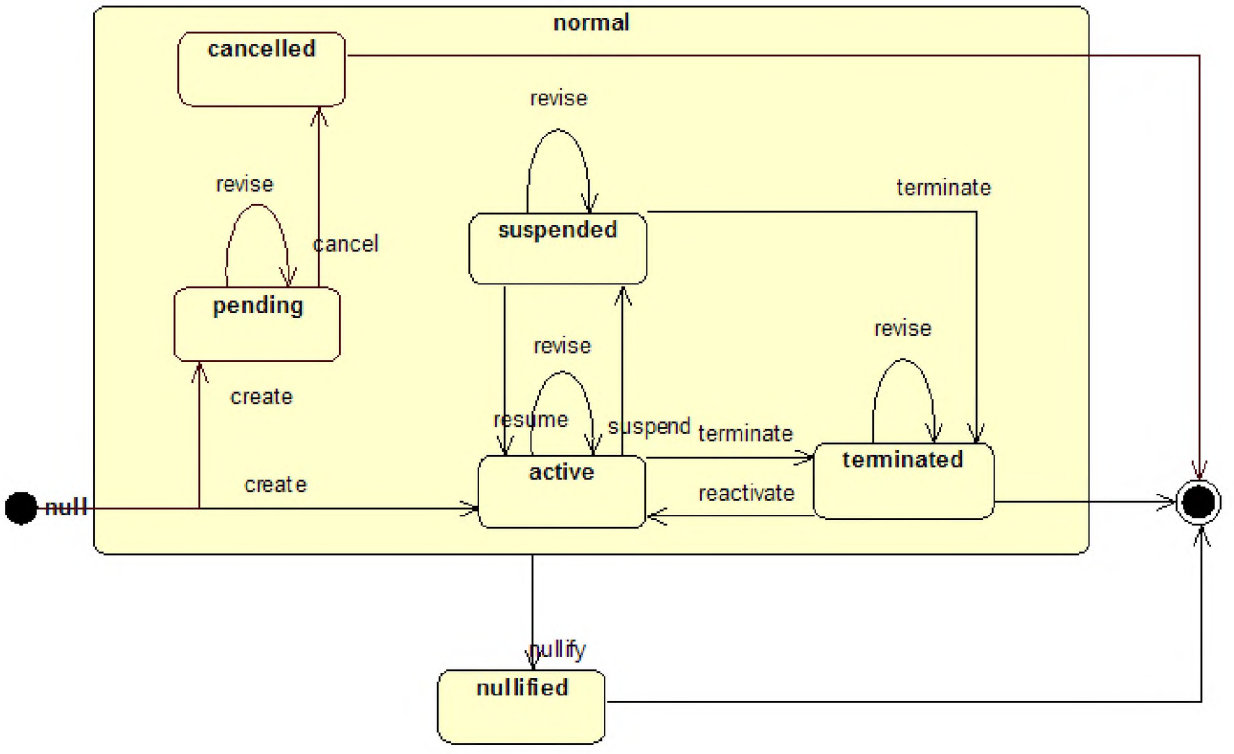


Рисунок A.8 — Диаграмма перехода состояний класса Role (роль)

A.1.5.5 Диаграммы классов инфраструктурных предметных областей модели RIM
Диаграммы классов инфраструктурных предметных областей модели RIM показаны на рисунках A.9—A.13.
A.1.5.5.1 Предметная область StructuredDocuments (структурированные документы)

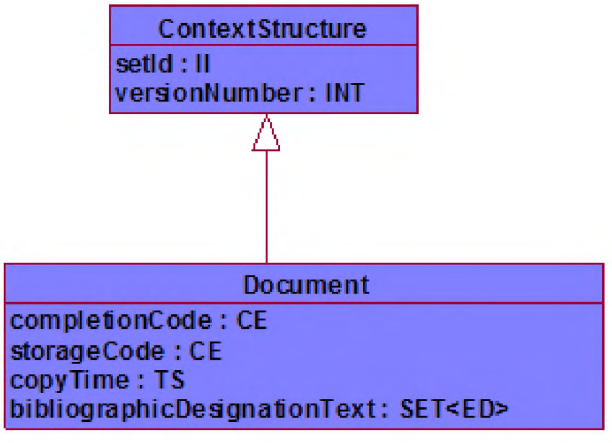


Рисунок A.9 — Диаграмма классов предметной области StructuredDocuments (структурированные документы)

A.1.5.5.2 Предметная область MessageCommunicationsControl (управление передачей сообщений)

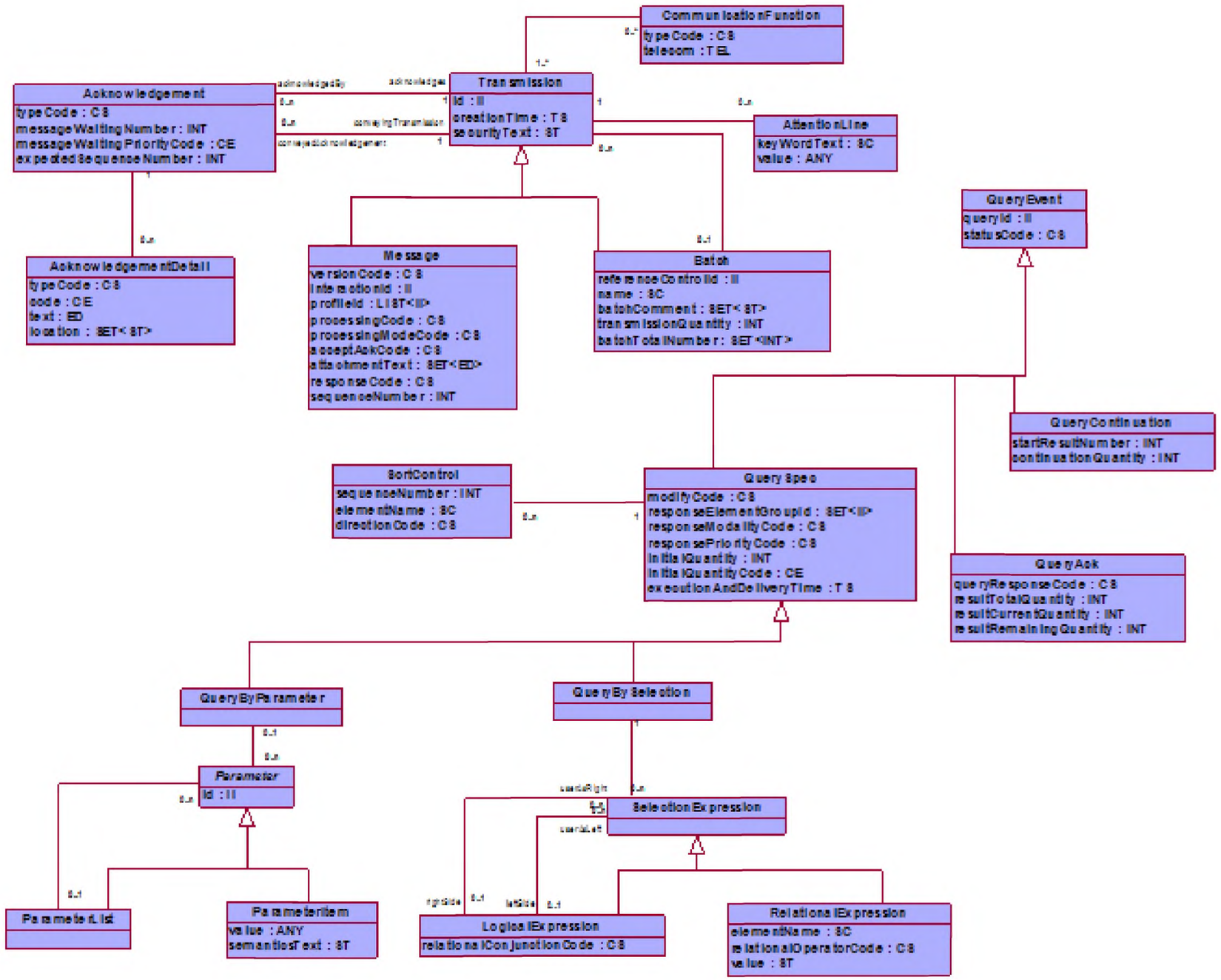


Рисунок А.10 — Диаграмма классов предметной области MessageCommunicationsControl (управление передачей сообщений)

A.1.5.5.3 Предметная область MessageControl (управление сообщением)

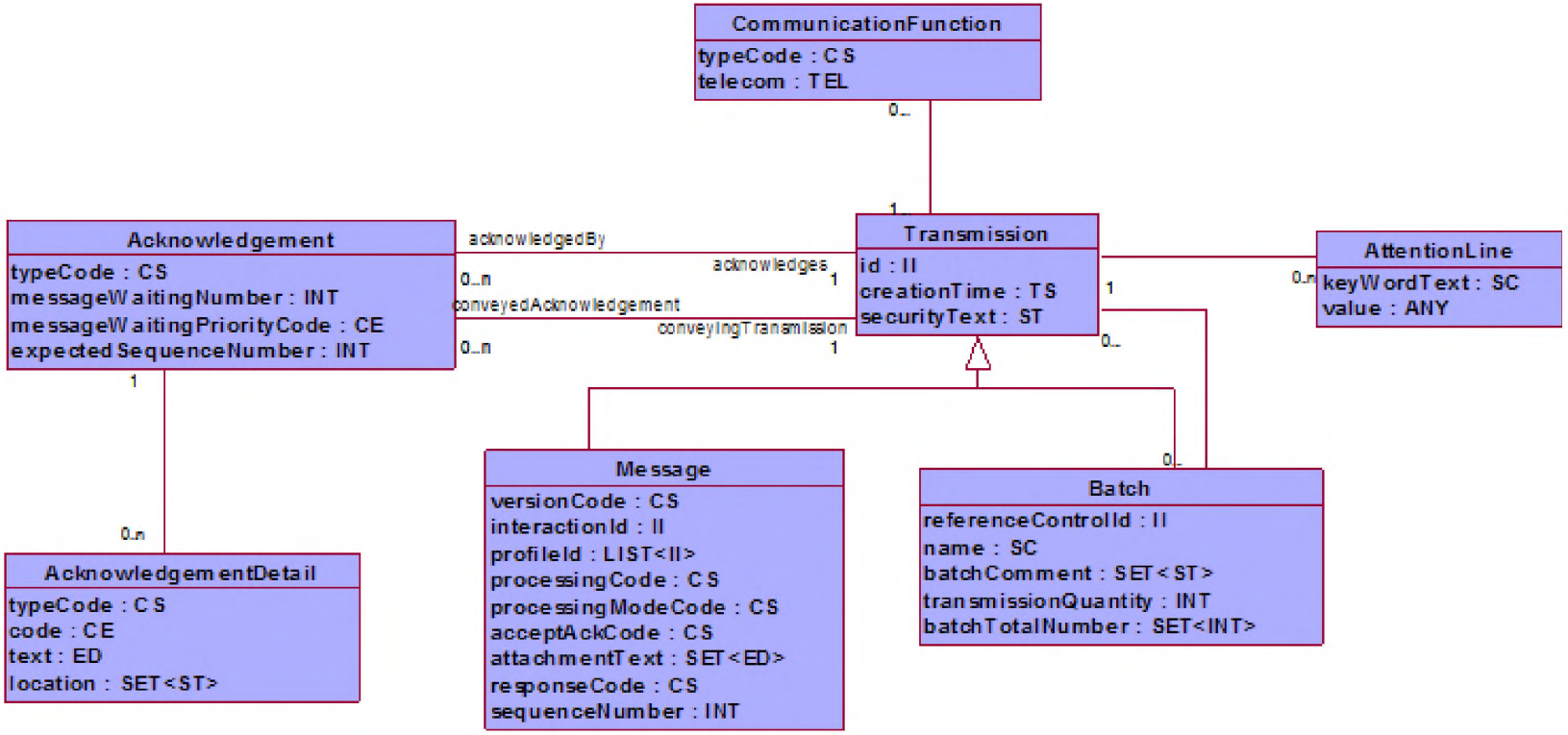


Рисунок А.11 — Диаграмма классов предметной области MessageControl (управление сообщением)

А.1.5.5.4 Предметная область QueryControl (управление запросами)

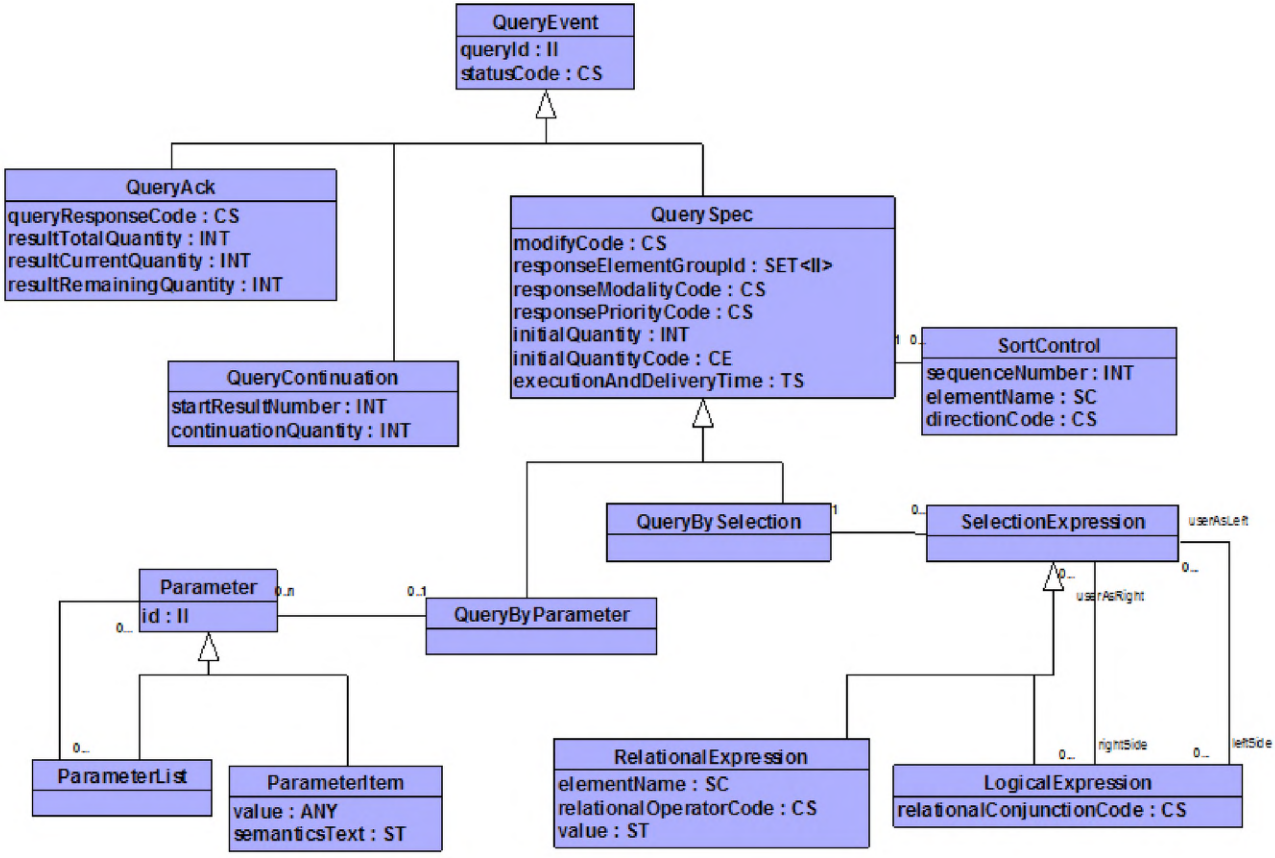


Рисунок А.12 — Диаграмма классов предметной области QueryControl (управление запросами)

A.1.5.5.5 Предметная область CoreInfrastructure (базовая инфраструктура)

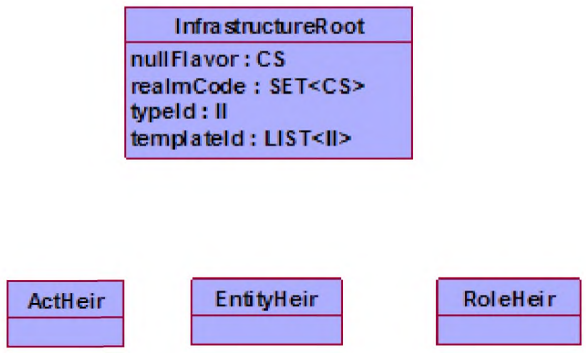


Рисунок A.13 — Диаграмма классов предметной области CoreInfrastructure (базовая инфраструктура)

A.1.5.5.6 Диаграмма перехода состояний класса QueryEvent (событие запроса)

Диаграмма перехода состояний класса QueryEvent (событие запроса) показана на рисунке A.14.

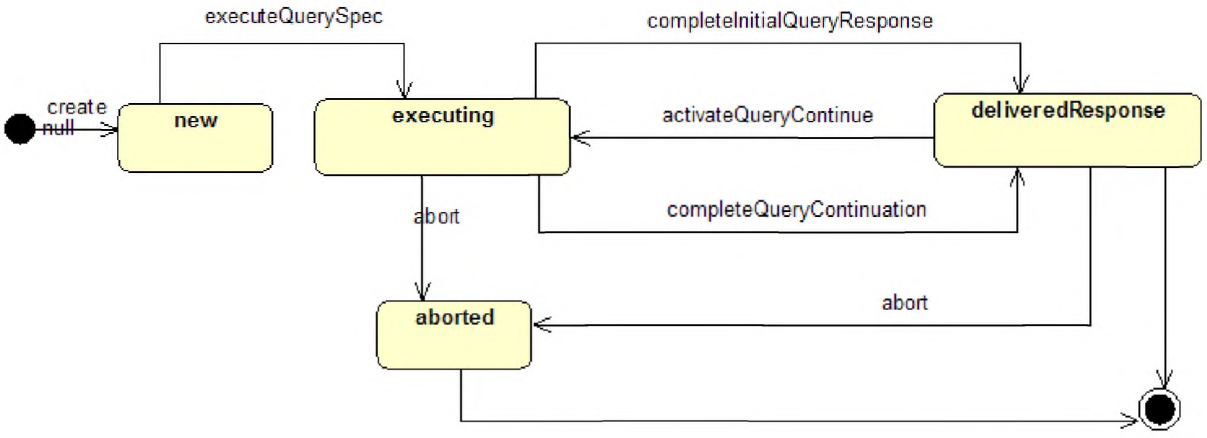


Рисунок A.14 — Диаграмма перехода состояний класса QueryEvent (событие запроса)

A.1.6 Примечания к версии

В настоящей версии модели RIM отражены изменения, одобренные в процессе гармонизации по состоянию на 18 ноября 2004 года, дополненные двумя раундами технической коррекции исходной публикации. Эта версия является седьмой по счету, охватывающей содержание, которое должно стать частью Выпуска 2 нормативной модели RIM.

Вопросы по содержанию следует адресовать сопредседателям комитета Methodology and Modeling (методология и моделирование) или посылать их в список рассылки этого комитета по адресу mnm@lists.hl7.org.

A.2 Предметные области

A.2.1 Предметная область FoundationClasses (в базовой модели)

Эта совокупность классов и их ассоциаций представляет «нормативное содержание» модели HL7 RIM. Содержание данной предметной области утверждается в комитете HL7 как нормативный документ.

Диаграмма классов этой предметной области показана на рисунке A.1.

Предметная область FoundationClasses содержит следующие предметные области:

- Acts (действия),
- Entities (сущности),
- Roles (роли).

A.2.1.1 Предметная область Acts (в предметной области FoundationClasses)

Совокупность классов, включая класс Act и его специализации. Они относятся к действиям и событиям, связанным с оказанием медицинской помощи.

Диаграмма классов этой предметной области показана на рисунке А.2.

Предметная область Acts содержит следующие классы:

- Account (счет);
- Act (действие);
- ActRelationship (связь действий);
- ControlAct (управляющее действие);
- DeviceTask (функция устройства);
- DiagnosticImage (диагностическое изображение);
- Diet (диета);
- FinancialContract (контракт);
- FinancialTransaction (финансовая операция);
- InvoiceElement (элемент счета-фактуры);
- ManagedParticipation (управляемое участие);
- Observation (исследование);
- Participation (участие);
- PatientEncounter (контакт с пациентом);
- Procedure (процедура);
- PublicHealthCase (событие общественного здоровья);
- SubstanceAdministration (лекарственное назначение);
- Supply (предоставление материала);
- WorkingList (рабочий лист).

А.2.1.2 Предметная область Entities (в предметной области FoundationClasses)

Совокупность классов, содержащая класс Entity, его специализации и связанные с ними квалифицирующие классы. Эти классы представляют участников медицинской помощи и другие сущности здравоохранения.

Диаграмма классов этой предметной области показана на рисунке А.3.

Предметная область Entities содержит следующие классы:

- Container (контейнер);
- Device (устройство);
- Entity (сущность);
- LanguageCommunication (вербальное общение);
- LivingSubject (живой организм);
- ManufacturedMaterial (произведенный материал);
- Material (материал);
- NonPersonLivingSubject (нечеловеческое живое существо);
- Organization (организация);
- Person (лицо);
- Place (место).

А.2.1.3 Предметная область Roles (в предметной области FoundationClasses)

Совокупность классов, содержащая класс Role и его специализации. Эти классы представляют роли участников процесса оказания медицинской помощи.

Диаграмма классов этой предметной области показана на рисунке А.4.

Предметная область Roles содержит следующие классы:

- Access (доступ);
- Employee (работник);
- LicensedEntity (лицензированный субъект);
- Patient (пациент);
- Role (роль);
- RoleLink (связь роли).

А.2.2 Предметная область CommunicationInfrastructure (в базовой модели)

Эта совокупность предметных областей представляет техническую инфраструктуру стандартов HL7, включая сообщения, структурированные документы и компоненты.

Предметная область CommunicationInfrastructure содержит следующие предметные области:

- CoreInfrastructure (базовая инфраструктура),
- MessageCommunicationsControl (управление передачей сообщений),
- StructuredDocuments (структурированные документы).

А.2.2.1 Предметная область CoreInfrastructure (в предметной области CommunicationInfrastructure)

Эта предметная область содержит классы, являющиеся базовыми элементами коммуникационной инфраструктуры HL7.

Диаграмма классов этой предметной области показана на рисунке А.13.

Предметная область CoreInfrastructure содержит следующие классы:

- ActHeir (наследование действия),
- EntityHeir (наследование сущности),

- InfrastructureRoot (корень инфраструктуры),
- RoleHeir (наследование роли).

A.2.2.2 Предметная область MessageCommunicationsControl (в предметной области CommunicationInfrastructure)

Эта предметная область содержит классы, относящиеся к техническому определению и управлению коммуникациями, основанными на обмене сообщениями, в стандартах HL7.

Диаграмма классов этой предметной области показана на рисунке A.10.

Предметная область MessageCommunicationsControl содержит следующие предметные области:

- MessageControl (управление сообщениями);
- QueryControl (управление запросами).

A.2.2.2.1 Предметная область MessageControl (в предметной области MessageCommunicationsControl)

Эта предметная область содержит те элементы модели RIM, что имеют отношение к управлению, передаче и подтверждению сообщений.

Диаграмма классов этой предметной области показана на рисунке A.11.

Предметная область MessageControl содержит следующие классы:

- Acknowledgement (подтверждение),
- AcknowledgementDetail (детальная информация подтверждения),
- AttentionLine (идентификация адресата сообщения),
- Batch (пакет сообщений),
- CommunicationFunction (функция коммуникации),
- Message (сообщение),
- Transmission (передача).

A.2.2.2.2 Предметная область QueryControl (в предметной области MessageCommunicationsControl)

Эта предметная область содержит те элементы модели RIM, что имеют отношение к формулировке и передаче запросов, а также к ответам на запросы.

Диаграмма классов этой предметной области показана на рисунке A.12.

Предметная область QueryControl содержит следующие классы:

- LogicalExpression (логическое выражение),
- Parameter (параметр),
- ParameterItem (компонент параметра),
- ParameterList (список параметров),
- QueryAck (подтверждение запроса),
- QueryByParameter (запрос с параметром),
- QueryBySelection (запрос с фильтрацией),
- QueryContinuation (продолжение запроса).
- QueryEvent (событие запроса),
- QuerySpec (спецификация запроса),
- RelationalExpression (реляционное выражение),
- SelectionExpression (формулировка фильтра),
- SortControl (управление сортировкой).

A.2.2.3 Предметная область StructuredDocuments (в предметной области CommunicationInfrastructure)

Эта предметная область содержит классы, относящиеся к передаче документов в стандартах HL7, представленных Архитектурой клинических документов.

Диаграмма классов этой предметной области показана на рисунке A.9.

Предметная область StructuredDocumentscодержит следующие классы:

- ContextStructure (структура контекста),
- Document (документ).

A.3 Классы

Далее перечислены все классы модели RIM. Порядок сортировки основан на следующих трех критериях:

- основные классы, за которыми следует коммуникационное и инфраструктурное содержание;
- имя основной предметной области (в алфавитном порядке);
- имя класса (в алфавитном порядке).

A.3.1 Классы предметной области Acts

A.3.1.1 Класс Account (в предметной области Acts)

Свойства класса Account

Атрибуты класса Account:

- balanceAmt :: MO
- currencyCode :: CE
- interestRateQuantity :: RTO<MO,PQ>
- allowedBalanceQuantity :: IVL<MO>

Класс Account является специализацией класса Act.

Определение класса Account

Специализация класса Act, представляющая категорию финансовых операций, которые проводятся на отдельном балансе.

Обсуждение. Этот класс может использоваться для представления накопленной общей суммы, полученной за товары или услуги, оплаченной товары или услуги, а также для представления счетов дебита и кредита, между которыми осуществляются операции.

Примеры — Лицевые счета пациентов, счета за случаи обращения; центры затрат; счета дебиторов.

Атрибуты класса Account

A.3.1.1.1 Account.balanceAmt:: MO (0..1)

Сумма операций по дебету и кредиту счета (остаток) с момента его открытия.

Обсуждение. Остаток счета обычно сообщается в валюте, указанной в атрибуте Account.currencyCode. Однако сумму остатка можно сообщать в альтернативных валютах.

A.3.1.1.2 Account.currencyCode:: CE (0..1)

Словарный домен: Currency

Указывает валюту счета.

Обсуждение. Конкретные суммы можно сообщать в другой валюте, однако данный атрибут должен представлять валюту, используемую по умолчанию для операций по этому счету.

A.3.1.1.3 Account.interestRateQuantity:: RTO<MO,PQ> (0..1)

Отношение, указывающее процентную ставку остатка по этому счету, и период, для которого она действует.

Обсуждение. В этом атрибуте может быть указана процентная ставка для платежей (например, по ссудам, просроченным счетам) или для поступлений (по вложениям капитала и т. д.) в зависимости от типа счета.

Примеры — 0.10/1a (10% в год); 0.0005895/1d (0,05895% в день).

Ограничения. Должна иметься возможность перевода единиц измерения знаменателя, имеющего тип данных PQ, в секунды. (Т. е. знаменателем должно быть время.)

A.3.1.1.4 Account.allowedBalanceQuantity:: IVL<MO> (0..1)

Интервал, описывающий минимально и максимально допустимое значение остатка счета.

Обсуждение. Этот интервал не обязательно задает «жесткие» пределы (т. е. остаток может быть и выше, и ниже указанных пределов). Он указывает такой «целевой» диапазон остатка счета, выход из которого может повлечь за собой определенные последствия. Для остатка счета не обязательно одновременно задавать и верхний, и нижний предел (или какой-либо из них).

Примеры — Пределы для «прекращения потерь»; пределы для кредита.

A.3.1.2 Класс Act (в предметной области Acts)

Свойства класса Act

Атрибуты класса Act:

- classCode :: CS,
- moodCode :: CS,
- id :: SET<II>,
- code :: CD,
- negationInd :: BL,
- derivationExpr :: ST,
- title :: ED,
- text :: ED,
- statusCode :: CS,
- effectiveTime :: GTS,
- activityTime :: GTS,
- availabilityTime :: TS,
- priorityCode :: SET<CE>,
- confidentialityCode :: SET<CE>,
- repeatNumber :: IVL<INT>,
- interruptibleInd :: BL,
- levelCode :: CE,
- independentInd :: BL,
- uncertaintyCode :: CE,
- reasonCode :: SET<CE>,
- languageCode :: CE.

Ассоциации класса Act:

- outboundRelationship::(0..*) ActRelationship::source::(1..1) (ассоциация с классом ActRelationship, роль source — источник);

- inboundRelationship::(0..*) ActRelationship::target::(1..1) (ассоциация с классом ActRelationship, роль target — цель);
- participation::(0..*) Participation::act::(1..1) (ассоциация с классом Participation, роль act — действие).

Класс Act является обобщением следующих классов:

- Account,
- ControlAct,
- DeviceTask,
- FinancialContract,
- FinancialTransaction,
- InvoiceElement,
- Observation,
- PatientEncounter,
- Procedure,
- SubstanceAdministration,
- Supply,
- WorkingList,
- ActHeir,
- ContextStructure.

Диаграмма перехода состояний класса Act приведена на рисунке А.5.

Определение класса Act

Запись о чем-то, что делается, что было сделано, что может быть сделано, что планируется или что требуется сделать.

Видами действий, типичными для здравоохранения, являются:

1) клиническое исследование;
2) оценка состояния здоровья (например, жалобы и диагнозы);
3) цели медицинской помощи;
4) услуги лечения (например, лекарственная терапия, хирургическое лечение, физиотерапия и психотерапия);
5) ассистирование, мониторинг, ведение пациента;
6) санитарное просвещение пациентов и их близких лиц;
7) услуги нотариуса (например, упреждающие указания или отказ от искусственного поддержания жизни);
8) редактирование и обработка документов, и т. д.

Обсуждение и обоснование. Классы действий являются краеугольным камнем RIM; информация всех предметных областей и процессов, в основном, сосредоточена в классах действий. Любая профессия или деятельность, включая здравоохранение, состоит из намеренных действий, выполняемых и регистрируемых ответственными лицами. Экземпляр класса действия представляет собой запись о таком намеренном действии. Намеренные действия отличаются от тех, что происходят в силу естественных причин (естественные события). Такие естественные события не являются действиями как таковыми, но могут регистрироваться как экземпляры класса Observation (исследования).

Соединение классов действий Act с классами сущностей Entity осуществляется с помощью классов участия Participation и ролей Role. Соединение одного класса действий Act с другим классом Act осуществляется с помощью класса связи действий ActRelationship. Классы участия Participation представляют авторов, исполнителей и другие ответственные стороны, а также субъектов и бенефициариев (в том числе инструменты и материал, используемые при выполнении действия, которые также являются субъектами). Использование атрибута moodCode позволяет различить фактически выполненные действия, запланированные и затребованные действия, а также другие варианты определенности действия.

Каждый класс действия Act имеет (в крайнем случае, косвенно) связь с одним из классов Participation, представляющим основного автора, ответственного за выполнения действия и «владеющего» действием. Ответственность за действие означает ответственность за то, что оно означает, и как это записано. А владелец действия имеет право его изменить в рабочем порядке. Ответственность за действие и владение действием, представленным классом Act, не идентичны ответственности и владению по отношению к тому, что соответствует экземпляру этого класса в реальном мире. Одна и та же деятельность, происходящая в реальном мире, может быть описана с точки зрения двух людей, каждый из которых является автором своего экземпляра класса Act, содержащего такое описание. Но один из этих людей может быть свидетелем деятельности, а другой — ее основным исполнителем. Исполнитель несет ответственность за физические действия; свидетель отвечает только за достоверное описание этих действий в силу своих способностей. Эти два экземпляра класса Act могут даже не быть согласованы между собой, но поскольку у каждого из них есть свой автор, такие разногласия могут сосуществовать и рассматриваться получателем этих экземпляров класса Act.

В этом отношении экземпляр класса Act представляет собой «утверждение» в терминах Rector и Nowlan (1991) [Foundations for an electronic medical record. Methods Inf Med. 30]. Rector и Nowlan подчеркнули важность понимания медицинской карты не как собрания фактов, а как «достоверного отчета о том, что слышали, видели, думали и делали медицинские работники». Rector и Nowlan продолжают эту мысль, заявляя, что «другие требования, предъявляемые к ведению медицинской карты, например, что она должна иметь определенную принадлежность и

обладать свойством постоянства, естественно вытекают из этой точки зрения». Действительно, класс Act как раз и представляет собой такое «утверждение», содержащее атрибуты, и правила обновления экземпляров класса Act (обсуждаемые в модели перехода состояний, см. атрибут Act.statusCode), применяемые вместо создания новых экземпляров этого класса, разработаны в соответствии с принципом постоянства и принадлежности.

Rector и Nowlan описывают электронную медицинскую карту как совокупность утверждений, которые имеют определенную принадлежность, но ограничиваются изложением фактов. Однако класс Act выходит за это ограничение, относящееся к фактическим утверждениям определенной принадлежности, представляя то, что известно в лингвистике и философии как «речевые акты». Понятие речевого акта означает, что речевое выражение, кроме изложения факта, имеет определенное прагматическое значение, и что в реальном мире речевые выражения используются для изменения состояния дел, в том числе, чтобы непосредственно инициировать физическую деятельность. Например, заказ исследования представляет собой речевой акт, который (при условии, что оно адекватно) вызывает физическое выполнение заказанного действия. Кульминацией теории речевых актов является плодотворная работа Austin (1962) [How to do things with words. Oxford University Press].

В реальном мире отдельная деятельность может развиваться от ее формулировки до выполнения, проходя этапы планирования и указаний. Отражение этих этапов находит свое представление с помощью атрибута moodCode класса Act. Хотя такую деятельность и можно рассматривать в развитии от плана до выполнения, это развитие должно представляться в форме нескольких экземпляров класса Act, каждый из которых имеет ровно одно значение атрибута moodCode, которое не меняется в течение всего срока жизни этого экземпляра. Это связано с тем, что принадлежность и содержание речевых актов в процессе деятельности могут быть различными, и нередко очень важно вести достоверную и постоянно хранящуюся регистрацию этого процесса. Спецификация указаний, обещаний или планов не должна заменяться на спецификацию того, что было выполнено на самом деле, чтобы иметь возможность сравнить результат с более ранними спецификациями. Экземпляры класса Act, являющиеся отражением развития одной деятельности, должны быть связаны между собой с помощью экземпляров класса ActRelationship, у которых атрибут typeCode имеет значение SEQL (продолжение).

Экземпляры класса Act, представляющие утверждения или речевые акты, являются единственным способом представления фактов реального мира или процессов в модели HL7 RIM. Правда о реальном мире конструируется только с помощью комбинаций (или выбора) таких утверждений, имеющих определенную принадлежность, и в модели RIM нет ни одного класса, экземпляры которого представляли бы «реальное положение дел» или «реальные процессы» независимо от этих утверждений. В силу этого обстоятельства не проводится никаких различий между деятельностью и ее описанием. Каждый экземпляр класса Act характеризует и то, и другое в различной степени. Примером может служить фактическое утверждение о недавних (но уже завершенных) действиях, сделанное (и подписанное) исполнителем этих действий, обычно известное как протокол или первичная документация (например, протокол хирургической операции, дневниковые записи и т. д.). А изменение состояния текущей деятельности, документируемое исполнителем (или непосредственным наблюдателем), рассматривается как сбор информации об этой деятельности (которая позже заменяется полным протоколом процедуры). Однако и обновление состояния, и протокол процедуры представляются экземплярами класса Act одного рода, которые можно различить по значениям атрибутов наклонения (moodCode) и состояния (statusCode), а также по завершенности информации.

Атрибуты класса Act

A.3.1.2.1 Act.classCode:: CS (1..1) Mandatory

Словарный домен: ActClass

Определение.

Код, устанавливающий главный тип класса Act, представляющий данный экземпляр этого класса.

Ограничения. Домен classCode представляет собой строго контролируемый словарь, который не может быть внешним или пользовательским.

Каждый экземпляр класса Act должен иметь атрибут classCode. Если этот класс не специализирован, то атрибут Act.classCode должен иметь наиболее общее значение (ACT).

Значение атрибута Act.classCode должно быть обобщением определенного понятия (например, заданного значением атрибута Act.code), другими словами, понятия, моделируемые классом Act и передаваемые в его экземпляре, должны быть специализациями класса понятий, заданного значением атрибута Act.classCode. В частности, атрибут classCode не является «модификатором» атрибута кода класса Act.code. (Для сравнения см. описание атрибута Act.code).

A.3.1.2.2 Act.moodCode:: CS (1..1) Mandatory

Словарный домен: ActMood

Код, позволяющий различить, что именно представляет данный экземпляр класса Act: фактическое утверждение, команду, возможность, цель и т. д.

Ограничения. Экземпляр класса Act должен иметь одно и только одно значение атрибута moodCode.

Значение атрибута moodCode конкретного экземпляра класса Act никогда не изменяется. Стадия деятельности, характеризуемая этим атрибутом, не является состоянием объекта.

Чтобы описать развитие конкретной деятельности от ее плана до выполнения, необходимо создать несколько экземпляров классов Act, имеющих разные значения атрибута moodCode, и связать их между собой с помощью экземпляров класса ActRelationship, у которых атрибут typeCode имеет значение SEQL (продолжение). (См. описание атрибута ActRelationship.typeCode.)

Обсуждение. Значение атрибута Act.moodCode описывает следующие понятия: (1) событие, т. е. фактическое описание произошедших действий; (2) определение возможных действий и планов действия (на уровне нормативно-справочного файла); (3) намерение, т. е. план действий, который для пациента выражен в форме плана лечения или направления; (4) цель, т. е. желаемый результат, приложенный к проблемам пациента и планам; и (5) критерий, т. е. предикат, используемый для вычисления логического выражения.

Значение атрибута Act.moodCode контролируемым образом изменяет смысл класса Act подобно тому, как в естественном языке грамматическая форма глагола определенным образом изменяет смысл предложения. Например, если значение атрибута Act.moodCode является признаком фактического события, то весь экземпляр класса Act представляет известный факт. Если оно является признаком плана (намерения), то весь экземпляр класса Act представляет описание того, что должно быть сделано. Значение атрибута Act.moodCode не меняет каким-либо особым способом конкретные свойства класса Act.

Поскольку значение атрибута Act.moodCode определяет смысл экземпляра класса Act, то оно должно быть всегда известно. Это означает, что всякий раз, когда создается экземпляр класса Act, его атрибуту moodCode должен быть присвоен допустимый код, который не может меняться в течение всего срока жизни этого экземпляра.

Поскольку смысл экземпляра класса Act задается кодом, присвоенным атрибуту moodCode, то значение этого кода влияет на интерпретацию всего этого экземпляра, включая каждое его свойство (атрибут или ассоциацию). Коль скоро значение атрибута moodCode влияет на интерпретацию экземпляра класса, то смысл этого экземпляра, в свою очередь, влияет на смысл его атрибутов. Однако значение атрибута moodCode не может оказать непосредственное влияние на смысл отдельного атрибута.

Классы Act имеют два типа свойств действий — инертные и описательные. Смысл инертных свойств не зависит от значения атрибута Act.moodCode, а интерпретация описательных свойств зависит. Например, у класса Act есть атрибут Act.id, который обеспечивает уникальную идентификацию экземпляра этого класса. Уникальная идентификация объекта никоим образом не зависит от значения атрибута Act.moodCode. Поэтому «интерпретация» идентификатора Act.id является инертной по отношению к атрибуту Act.moodCode.

Напротив, большинство из других атрибутов класса Act является описательным по отношению к утверждению, передаваемому в форме экземпляра класса Act. Эти атрибуты дают ответы на вопросы, кто выполнил действие, для кого, где, что использовал, как и когда было выполнено это действие. Ответы на вопросы, кто, для кого, где, что использовал, передаются в описательных атрибутах классов Participation, а ответы на вопросы, как и когда — в описательных атрибутах и экземплярах классов ActRelationship. Интерпретация описательного атрибута зависит от интерпретации всего экземпляра и управляется значением атрибута moodCode.

Обоснование. Понятие «наклонения» (mood) заимствовано из грамматики естественного языка, а именно, наклонения глагола (лат. modus verbi).

Понятие наклонения также напоминает различные расширения логики фактов в модальной логике и логики с модальностями, в том смысле, что значение атрибута moodCode задает модальность (факт, возможность, намерение, цель и т. д.), по которой можно судить, является ли утверждение, описываемое экземпляром класса Act, подходящим или неприемлемым.

Примеры. Для иллюстрации влияния атрибута moodCode ниже рассмотрены экземпляры класса Act, относящиеся к процессу определения сахара крови.

Экземпляр специализации класса Act, у которого атрибут moodCode имеет значение DEF (definition — описание), содержит справочное описание процесса «определение сахара крови». Связанные с ним экземпляры классов Participation содержат характеристики субъектов, которые должны участвовать в этом процессе, и требуемых для него объектов, например, образец, подразделение, лабораторное оборудование и т. д. Значение атрибута Observation.value указывает абсолютный диапазон значений (домен) результата анализа (например, «15—500 мг/дл»).

Если атрибут moodCode имеет значение INT (intent — намерение), это означает, что автор, указанный в экземпляре класса, намерен назначить анализ концентрации сахара в крови («надо определить сахар крови»). Связанные с ним экземпляры классов Participation содержат информацию о тех субъектах и объектах, которые фактически или предположительно участвуют в этом назначении, в первую очередь об авторе намерения или о любом отдельном лице при групповом намерении, а также о передаваемом образце, о требованиях к лабораторному оборудованию и т. д. Атрибут Observation.value в этом случае обычно отсутствует, поскольку речь идет о намерении провести анализ концентрации сахара, а не измерить концентрацию сахара в указанном диапазоне значений. (Иная ситуация описана ниже, когда атрибут moodCode имеет значение GOL.)

Экземпляр специализации класса Act, у которого атрибут moodCode имеет значение RQO (request — требование, что можно рассматривать как разновидность намерения), содержит направление на анализ концентрации сахара в крови («определите сахар крови»). Связанные с ним экземпляры классов Participation содержат информацию о субъектах и объектах, которые фактически или предположительно должны участвовать в процессе выполнения анализа, в первую очередь о заказчике анализа и о выбранном исполнителе, а также о передаваемом образце, о требованиях к лабораторному оборудованию и т. д. Атрибут Observation.value в этом случае обычно отсутствует, поскольку речь идет о намерении провести анализ концентрации сахара, а не измерить концентрацию сахара в указанном диапазоне значений.

Экземпляр специализации класса Act, у которого атрибут moodCode имеет значение EVN (event — событие), содержит результат определения сахара крови («сахар крови определен»). Связанные с ним экземпляры классов

Participation содержат информацию о субъектах и объектах, фактически участвовавших в процессе определения (включая образец, подразделение, лабораторное оборудование). Атрибут Observation.value содержит измеренное значение (например, «80 мг/дл» или «<15 мг/дл»).

Если атрибут moodCode имеет значение EVN.CRT (event-criterion — критерий события), это означает, что автор, указанный в экземпляре класса, рассматривает некоторый класс процессов «определения сахара крови», возможно, с определенным критерием оценки (диапазоном). Связанные с ним экземпляры классов Participation содержат критерий, применяемый, например, к пациенту. Атрибут Observation.value содержит диапазон значений критерия (например, «>180 мг/дл» или «200—300 мг/дл»).

Экземпляр специализации класса Act, у которого атрибут moodCode имеет значение GOL (goal — цель, что можно рассматривать как разновидность критерия), содержит информацию о цели, которую требуется достичь («целью является определенный уровень (диапазон) концентрации сахара в крови»). Связанные с ним экземпляры классов Participation содержат информацию, близкую к той, что была указана в намерении определения сахара крови, в первую очередь сведения об авторе цели и о пациенте, по отношению к которому эта цель поставлена. Атрибут Observation.value содержит целевой диапазон значений (например, «80—120 мг/дл»).

A.3.1.2.3 Act.id:: SET<II> (0..*)

Определение. Уникальный идентификатор экземпляра класса Act.

A.3.1.2.4 Act.code:: CD (0..1)

Словарный домен: ActCode

Определение. Код, указывающий конкретный вид действия, представленного экземпляром класса Act.

Ограничения. Для указания вида действия (например, физикальное исследование, определение калия в сыворотке крови, госпитализация, финансовая транзакция по оплате лечения и т. д.) используется код, который берется из какой-либо (обычно внешней) системы кодирования. Система кодирования зависит от конкретной специализации класса Act, например, для класса Observation, описывающего исследования, может использоваться система кодирования LOINC, и т. д.

Концептуально значение атрибута Act.code должно быть специализацией значения атрибута Act.classCode. Поэтому структура словарного домена ActClass должна найти свое отражение на верхнем уровне структуры словарного домена ActCode и отдельные коды или внешние словари должны быть подчинены структуре словарного домена ActClass.

Атрибуты Act.classCode и Act.code не являются модификаторами друг для друга, однако понятие, передаваемое в атрибуте Act.code, должно логически вытекать из понятия, передаваемого в атрибуте Act.classCode. Негативным примером служит использование атрибута Act.code для передачи понятия «калий» одновременно в экземпляре класса Observation, у которого атрибут Act.classCode имеет значение SPCOBS (specimen observation — лабораторное исследование образца), чтобы он означал «лабораторное исследование содержания калия», и в экземпляре класса Medication, у которого атрибут Act.classCode имеет значение SBADM (substance administration — лекарственное назначение), чтобы он означал «замещение калия». Такое взаимное изменяющее использование сочетаний атрибутов Act.classCode и Act.code не допускается.

Обсуждение. Атрибут Act.code не является обязательным в классе Act. Вместо конкретизации вида действия с помощью атрибута Act.code можно воспользоваться атрибутом Act.classCode и другими атрибутами и свойствами класса Act. Более общий и чаще встречающийся прием состоит в задании вида действия с помощью экземпляра класса Act, в котором атрибут Act.moodCode имеет значение DEF, связанного с другим экземпляром класса Act с помощью экземпляра класса ActRelationship. Вид действия без труда можно указать и без такой привязки к его определению, используя другие атрибуты, а также классы ActRelationship и Participation. Например, вид лекарственного назначения, передаваемого в экземпляре класса SubstanceAdministarion, можно указать с помощью ассоциации ActRelationship с экземпляром класса Entity, содержащим информацию о конкретном лекарстве.

Act.negationInd:: BL (0..1)

Определение. Признак, указывающий, что к описательным атрибутам утверждения, передаваемого в экземпляре класса Act, должно быть применено отрицание.

Пример — Использование этого признака в экземпляре класса Observation, у которого атрибут Act.moodCode имеет значение «EVN» (event — событие), позволяет передать утверждение «у пациента НЕТ боли в груди». А в экземпляре класса Observation, у которого атрибут Act.moodCode имеет значение «EVN.CRT» (criterium — критерий), использование идентификатора отрицания позволяет передавать критерии выполнения действия вида «если у пациента НЕТ боли в груди в течение 3 дней...», или «если систолическое давление НЕ находится в пределах 90—100 мм рт. ст...».

Обсуждение. Атрибут negationInd используется как отрицание квантора существования. Его смысл лучше всего объяснять для экземпляров класса Act, у которых атрибут Act.moodCode имеет значение «EVN.CRT» (criterium — критерий). Если критерий используется без признака отрицания, то обычно в экземпляре класса Act надо передать только несколько тех критичных атрибутов и связей (параметров), которые нужны для проверки критерия. Чем больше параметров указано, тем более специфичным (ограниченным) будет критерий. Например, для задания критерия «систолическое давление находится в пределах 90—100 мм рт. ст.» в экземпляре класса Observation достаточно указать описательные атрибуты Act.code, указывающий «измерение кровяного давления», и Observation.value, указывающий диапазон значений «90—100 мм рт. ст.». Если будет указан еще и атрибут

effectiveTime, скажем, со значением «вчера», то критерий станет более узким. Если при таком критерии атрибут negationInd будет иметь значение TRUE (истина), то критерий приобретет следующий смысл: «не существует вчерашнего значения систолического кровяного давления в диапазоне 90—100 мм рт. ст.» (вне зависимости от того, измерялось вчера кровяное давление или нет).

Значение атрибута negationInd воздействует указанным выше образом на описательные атрибуты класса Act (включая Act.code, Act.effectiveTime, Observation.value, Act.doseQty, и т. д.), и на любые их компоненты. Инертные свойства, например, Act.id, Act.moodCode, Act.confidentialityCode, и особенно ассоциация с классом Participation, имеющая роль автора, остаются неизменными. Эти инертные свойства всегда имеют одно и то же значение: т. е., автор остается и автором отрицательного исследования. Кроме того, атрибут negationInd не воздействует и на большинство связей ActRelationship (за исключением компонентов).

Например, крайне конфиденциальное указание, записанное д-ром Джонсом в форме «не применять сукцинилхолин» в связи (класс ActRelationship) с имевшейся злокачественной гипертермией (класс Observation), является отрицанием положительного указания «применить сукцинилхолин» (атрибут Act.code), но тем не менее остается указанием, написанным д-ром Джонсом для пациента Джона Смита, и причина этого указания — имевшаяся у пациента злокачественная гипертермия.

Однако дополнительные детали, передаваемые в описательных атрибутах, будут частью отрицания, ограничивая его воздействие. Например, если в указании не применять субстанцию присутствует атрибут дозы doseQuantity, то это означает, что нельзя давать эту конкретную дозу субстанции (но любая другая доза могла бы оставаться допустимой).

Экземпляр класса Act, у которого атрибут negationInd имеет значение TRUE, тем не менее, остается утверждением об определенном факте, описанном в этом экземпляре. Например, отрицание утверждения «1 июля выявлена одышка» означает, что его автор определенно отрицает, что 1 июля была одышка, и что он несет ту же самую ответственность за это утверждение и те же самые требования к его доказательству, как если бы он не использовал отрицание. И наоборот, признак отрицания, переданный в атрибуте negationInd, никоим образом не отрицает того, что факт был подтвержден или что утверждение имело место. Это равным образом относится ко всем наклонениям утверждения, задаваемым атрибутом moodCode, например, применение отрицания к направлению является указанием не делать того, что в нем написано, а вовсе не лапидарное утверждение, что такого направления нет.

A.3.1.2.6 Act.derivationExpr:: ST (0..1)

Определение. Строка символов, содержащая выражение на формальном языке, указывающее, каким образом значения атрибутов экземпляра класса Act выводятся из входных параметров, связанных с этим экземпляром отношениями «произведен из».

Обсуждение. Производный результат исследования, передаваемый в экземпляре класса Observation, может быть определен с помощью ассоциаций ActRelationship, у которых атрибут typeCode имеет значение DRIV (is derived from — произведен из) и которые связывают данный экземпляр с другими экземплярами класса Observation. Например, для определения производного исследования «среднее содержание гемоглобина в эритроците» (MCH) надо связать исследование MCH с исследованием концентрации гемоглобина (Hb) и абсолютным содержанием эритроцитов (RBC). В этом случае производное выражение должно задаваться формулой MCH = Hb / RBC.

Производное выражение представляется в виде строки символов.

Примечание — Синтаксис такого выражения должен быть полностью определен. Следует выбрать единственный стандартный язык выражений вместо необязательного выбора из нескольких языков. Синтаксис может быть основан на фактическом стандарте многих объектно–ориентированных языков, например, C++, Java, OCL и т. д. Конкретная спецификация такого языка выражений разрабатывается и ее проект ожидается в 2003 году.

A.3.1.2.7 Act.title:: ED (0..1)

Определение. Слово или фраза, под которыми данный экземпляр класса Act известен людям.

Пример — Название научного исследования (например, «Scandinavian Simvastatin Study»), название судебного дела (например, «Brown v. Board of Education»), наименование другого типа рабочего проекта или действия. Для действий, представляющих документы, в этом атрибуте передается название документа.

Ограничения. До версии 2.05 модели RIM этот атрибут имел тип данных ST. Начиная с версии 2.05, его тип данных был расширен до ограниченного типа данных ED. Наложенные ограничения идентичны тем, что присущи типу данных ST, только атрибут типа среды (mediaType) должен иметь значение «text/x-hl7-title+xml». Это было сделано, чтобы обеспечить возможность использования достаточного ассортимента разметки, позволяющего передать семантику научных фраз, например, названия химических компонентов. Такая разметка не должна использоваться для указания простых предпочтений отображения.

Обсуждение. Значение этого атрибута является скорее человеко-читаемым общим названием, а не формальным идентификатором. Однако оно похоже на атрибут id тем, что отночится к конкретному экземпляру класса Act, а не к «виду» действия. (в экземпляре класса Act, у которого атрибут moodCode имеет значение DEF (определение), значение атрибута title относится к этому конкретному определению, а не к широкой категории действий, задаваемой значением атрибута Act.code.)

Примечание — Этот атрибут не входил в нормативное содержание, утвержденное в первом выпуске стандарта RIM. Он будет рассмотрен при подготовке второго выпуска модели RIM. В настоящее время он используется в других стандартах HL7 Версии 3.

A.3.1.2.8 Act.text:: ED (0..1)

Текстовое или мультимедийное описание действия (либо ссылка на такое описание).

Пример — В экземпляре класса Act, у которого атрибут moodCode имеет значение DEF (определение), атрибут Act.text может содержать справочную информацию о действии. В экземпляре класса Act, у которого атрибут moodCode имеет значение RQO (направление), атрибут Act.text будет содержать специфические инструкции, применяемые только к этому направлению.

Ограничения. Никаких ограничений ни на содержание описания, ни на его длину не накладывается.

Содержание этого описания не считается частью функциональной информации, передаваемой между компьютерными системами. Если предполагается, что информация, переданная в экземпляре класса Act, предназначена, в том числе, для людей (читателей и исполнителей), то автоматизированная система должна показать пользователю поле Act.text либо каким-то образом уведомить его о том, что такое поле существует, и показать его по требованию.

Описания в форме свободного текста используются, чтобы человек мог интерпретировать содержание и контекст действия, но вся информация, необходимая для автоматизированных функций, должна быть передана, используя надлежащие атрибуты и ассоциированные объекты.

A.3.1.2.9 Act.statusCode:: CS (0..1)

Словарный домен: ActStatus

Определение. Код, описывающий состояние действия.

Рекомендация разработчикам: в исходной версии модели RIM этот атрибут был определен как повторяющийся, чтобы можно было отразить наличие вложенных подсостояний, указанных на диаграмме перехода состояний. На практике, однако, необходимость передачи нескольких значений состояния никогда не возникала. Поэтому комитетам рекомендуется ограничить кратность этого атрибута до 1 во всех конструкциях сообщений.

A.3.1.2.10 Act.effectiveTime:: GTS (0..1)

Определение. Выражение, указывающее «эффективное время» — момент или оперативное время действия, основное время осуществления действия, планируемое время действия.

Примеры

1 Для клинических исследований атрибут effectiveTime содержит время проведения исследования (эффективное время) пациента.

2 Для контрактов атрибут effectiveTime указывает срок действия контракта.

3 Для информированных согласий атрибут effectiveTime указывает срок действия согласия.

4 Для лекарственных назначений атрибут effectiveTime указывает время, в течение которого надо принимать лекарство, включая режим приема (например, трижды в день в течение 10 дней).

5 Для хирургической процедуры (операции) атрибут effectiveTime указывает время, значимое для пациента, например, время между разрезом и последним швом.

6 Для действий транспортировки атрибут effectiveTime указывает время в пути.

7 Для посещений пациентов атрибут effectiveTime указывает «административное время», т.е. даты начала и завершения посещения, определяемые в соответствии с административным регламентом, в отличие от фактического времени оказания медицинской помощи при данном посещении.

Обсуждение. Понятие «эффективного времени» называется также «основным» временем (в стандарте Arden Syntax) или «биологически значимым временем» (в стандарте HL7 v2.x). Этот атрибут надо отличать от атрибута activityTime, описывающего время деятельности.

Для исследований время деятельности может быть много позже времени наблюдения. Например, при анализе газов, содержащихся в артериальной крови, результат всегда будет получен через несколько минут после взятия биоматериала, а за это время физиологическое состояние пациента может значительно измениться.

Для существенно физической деятельности (хирургические процедуры, транспортировка и т. д.) эффективным временем является время целевого действия, например, коль скоро целью транспортировки является доставка груза из пункта А в пункт Б, то эффективным временем является время в пути. Однако действие обычно также включает в себя дополнительную работу, которая необходима для достижения цели действия, но не является существенной для этой цели.

Например, время, необходимое для приезда на место погрузки А и последующего возвращения на свою автобазу из пункта разгрузки Б, включается в физическую деятельность, но не включается во время транспортировки полезного груза. Другой пример — рабочие часы, считающиеся эффективным временем, могут быть с 8 часов утра до 5 часов вечера независимо от того, тратит ли человек на дорогу 10 минут или 2 часа. Приход на работу необходим, но на рабочие часы не влияет.

A.3.1.2.11 Act.activityTime:: GTS (0..1)

Определение. Выражение, указывающее время деятельности, т. е. когда происходило действие, описанное в экземпляре класса Act, либо, в зависимости от значения атрибута Act.moodCode, когда оно должно было происходить по предварительному плану, по текущему плану, когда оно могло происходить, и т. д. Время деятельности указывает, длительность действия, описанного в экземпляре класса Act. Атрибут activityTime содержит полное время деятельности, в том числе время, затрачиваемое на ее подготовку и на необходимые действия после ее завершения. В этих случаях значения атрибутов activityTime могут использоваться в функциях управления/планирования процедур и лекарственных назначений, поскольку более полно описывают затраты времени по сравнению с длительностью самого действия.

Обсуждение. Атрибут activityTime в большей степени предназначен для использования в административных, а не клинических целях. Клинически релевантное время передается в атрибуте effectiveTime. Если передается результат исследования ранее выявленного симптома, то атрибут activityTime содержит время исследования симптома, в отличие от атрибута effectiveTime, в котором должно передаваться время появления симптома. Поэтому значение атрибута activityTime может существенно отличаться от значения атрибута effectiveTime того же самого действия. Но даже без учета клинических аспектов разработчики должны рассматривать значение атрибута effectiveTime как наиболее релевантное время действия.

Значение атрибута activityTime указывает время деятельности, описанной в экземпляре класса Act, а не время регистрации деятельности. Во многих приложениях учитывается время регистрации результата исследования, а не время, в течение которого осуществлялось это исследование. В этих случаях должно использоваться время, передаваемое в атрибуте Participation.time (например, время. указанное автором). Регистрируемые результаты могут относиться к исследованиям, выполненным в процесее посещения пациента, и время посещения нередко является достаточно информативным, поэтому значение атрибута activityTime не является клинически релевантным.

A.3.1.2.12 Act.availabilityTime:: TS (0..1)

Определение. Момент времени, когда информация об экземпляре класса Act (в зависимости от значения атрибута Act.moodCode) впервые становится доступной системе, воспроизводящей информацию о действии, описанном в этом экземпляре.

Пример — В экземпляре класса Act может передаваться информация о том, что 3 часа назад у пациента произошел инфаркт миокарда правого желудочка (см. описание атрибута Act.effectiveTime), но информация об этом необычном случае поступила только несколько минут назад (значение атрибута Act.availabilityTime). Соответственно, с момента Act.effectiveTime до момента Act.availabilityTime любые вмешательства, скорее всего, предпринимались, исходя из предположения о более распространенном инфаркте миокарда левого желудочка. Новое знание может объяснить, почему эти вмешательства (например, назначение нитрата) могли оказаться не подходящими.

Обсуждение. Значение атрибута availabilityTime — субъективная вторичная часть информации, добавленная (или измененная) системой, информирующей о действии. Она не может быть приписана автору действия (и не должна включаться в информацию, которую автор засвидетельствует своей подписью). Система, информирующая о действии, нередко отличается от системы, зарегистрировавшей действие, т. е. получает информацию о действии из другой системы. Время получения этой информации должно быть присвоено атрибуту availabilityTime, поскольку, начиная с этого момента, пользователи системы-получателя должны получить возможность узнать об этом экземпляре класса Act.

При передаче атрибута availabilityTime другой системе время доступности экземпляра А класса Act передается в экземпляре В этого класса, который содержит экземпляр А или ссылку на него. Например, если выписка из медицинской карты подготовлена для отчета о непредвиденной побочной реакции, то автор отчета становится и автором значений availabilityTime, указанных в этом отчете.

A.3.1.2.13 Act.priorityCode:: SET<CE> (0..*)

Словарный домен: ActPriority

Определение. Код или последовательность кодов (описывающих, например, обычную или экстренную срочность действия), указывающих срочность, с которой действие случилось, может случиться, происходит, планируется или требуется.

Обсуждение. Этот атрибут используется в направлениях для обозначения требуемой срочности действия, а в документации о совершенном действии он указывает фактическую срочность действия. В экземплярах класса Act, у которых атрибут moodCode имеет значение DEF, в этом атрибуте указаны разрешенные значения срочности.

A.3.1.2.14 Act.confidentialityCode:: SET<CE> (0..*)

Словарный домен: Confidentiality

Определение. Код, контролирующий раскрытие информации о данном действии независимо от значения атрибута moodCode.

Обсуждение. Важно отметить, что необходимая конфиденциальность медицинской карты не может быть достигнута только с помощью кодов конфиденциальности, предназначенных для маскировки отдельных частей этой карты от определенных типов пользователей. При обеспечении конфиденциальности на основе кодов конфиденциальности, присваиваемых отдельным частям карты, возникают две проблемы: одна связана с возможностью логического вывода, а другая — с риском отсутствия доступа к информации, которая может быть критичной в

определенных случаях оказания медицинской помощи. Возможность вывода означает, что фильтрованная чувствительная информация может быть выведена из другой, не отфильтрованной информации. Простейшей формой вывода может служить пример, когда из факта наличия направления на анализ иммуноблота или на анализ количества Т4- и Т8-клеток с большой вероятностью можно сделать вывод, что пациент ВИЧ-инфицирован, даже если результат анализа не известен. Очень часто диагнозы могут быть выведены из лекарственных назначений, например, назначение зидовудина свидетельствует о лечении ВИЧ. Проблема сокрытия отдельных частей медицинской карты становится особенно трудной при наличии текущих лекарственных назначений, поскольку должно быть обеспечено продолжение приема лекарств.

Чтобы ослабить некоторые риски вывода, следует считать, что агрегированные данные имеют уровень конфиденциальности самого конфиденциального действия в агрегате.

A.3.1.2.15 Act.repeatNumber:: IVL<INT> (0..1)

Определение. Диапазон целых чисел, задающий минимальное и максимальное число повторений действия.

Пример — После удаления зуба хирург-стоматолог может дать следующий совет пациенту: «Меняйте тампон каждый час от одного до трех раз, пока кровотечение не остановится полностью». Этот совет преобразуется в значение атрибута repeatNumber с нижней границей 1 и верхней границей 3.

Обсуждение. Этот атрибут принадлежит к совокупности атрибутов управления рабочим процессом.

Число повторений дополнительно ограничивается временем. Действие будет повторяться, по меньшей мере, минимальное число раз и, самое большее, максимальное число раз. Повторения также закончатся, если время превысит максимальное значение атрибута Act.effectiveTime, если это случится раньше.

Использование: в экземпляре класса Act, у которого атрибут moodCode имеет значение «EVN» (event — событие), значение атрибута repeatNumber обычно равно 1. Если оно превышает 1, то этот экземпляр представляет собой сводную информацию о нескольких событиях, произошедших в течение интервала времени, указанного в атрибуте effectiveTime.

Чтобы различать экземпляры однотипных повторяющихся действий, используйте атрибут ActRelationship. sequenceNumber.

A.3.1.2.16 Act.interruptibleInd:: BL (0..1)

Определение. Признак, указывающий, может ли действие прерываться асинхронными событиями.

Обсуждение. Этот атрибут принадлежит к совокупности атрибутов управления рабочим процессом. Активные действия могут быть прерваны разными способами. Различаются следующие события прерывания: (1) когда получено прямое требование прекращения действия; (2) когда истекло время, выделенное для выполнения этого действия (тайм-аут); (3) когда «условие» выполнения действия перестает быть истинным (см. описание атрибута ActRelationship.checkpointCode); (4) когда экземпляр класса Act является компонентом, у которого атрибут joinCode имеет значение «K» (kill — прекращение) и все другие компоненты этой же группы завершены (см. описание атрибута Act.joinCode), и (5) когда прерывается объемлющий экземпляр класса Act.

Если действие получило прерывание и само оно может быть прервано, но в настоящее время имеет активные компоненты, которые не могут быть прерваны, то действие будет прервано тогда, когда все его активные не прерываемые компоненты будут завершены.

A.3.1.2.17 Act.levelCode:: CE (0..1)

Словарный домен: ActContextLevel

Определение. Код, определяющий уровень в иерархической структуре составного действия и тип контекста составных действий («контейнеров»), распространяемый на компоненты действия в пределах этих контейнеров. Значение атрибута levelCode обозначает положение в этой иерархии включения и применяемые ограничения.

Обсуждение. Следует иметь в виду, что этот признак может быть объявлен «устаревающим» в следующем нормативном выпуске модели HL7 RIM. Вместо него рассматривается альтернативное понятие уровня, использующее иерархию значений атрибута classCode. Если это изменение будет принято, то согласно процедурам сопровождения модели HL7 RIM атрибут levelCode будет объявлен «устаревающим» в следующем выпуске модели, а затем «устаревшим» в выпуске после этого. Прежде чем использовать этот атрибут, пользователям рекомендуется проверить самые последние внутренние определения, которые комитет HL7 использует в модели RIM.

Понятия уровня определены в целях удовлетворения специфичных требований к передаче медицинских карт. Хотя эти понятия и применимы к некоторым другим типам транзакций, они не образуют полностью закрытый список. Существуют варианты других наборов ортогональных уровней, которые должны удовлетворять деловым требованиям (например, сообщения о нескольких пациентах можно подразделить с помощью вышестоящего уровня предметных областей).

Пример — Экземпляры класса Act, находящиеся на «уровне выписки из медицинской карты» (значение атрибута Act.levelCode равно «EXTRACT») и «уровне папки» (значение атрибута Act.levelCode равно «FOLDER») должны содержать данные о единственном лице, в то время как на «уровне нескольких субъектов» эти экземпляры могут содержать данные более чем об одном лице. В то время как «выписка из медицинской карты» может быть сделана из нескольких источников, «папка» должна содержать данные из одного источника. Уровень «композиции» (значение атрибута Act.levelCode равно «COMPOSITION») обычно имеет единственного автора.

Ограничения. Ограничения, применимые к специфическому уровню, могут включать разные требования к участникам (например, к пациенту, к организации-источнику, к автору или другому лицу, подписывающему данные), к ассоциациям или включениям других экземпляров класса Act, к документам или к использованию шаблонов. Ограничения, применимые к уровню, могут также определить допустимые уровни экземпляров, которые могут быть компонентами этого уровня. Несколько вложенных уровней с тем же самым значением атрибута levelCode могут быть допустимыми, запрещенными (или ограниченными). Экземпляры класса Act следующего подчиненного уровня обычно разрешены на каждом уровне, но некоторые уровни могут быть опущены в модели, и допускается пропустить несколько уровней.

A.3.1.2.18 Act.independentInd:: BL (0..1)

Определение. Признак, указывающий, можно ли управлять данным экземпляром класса Act независимо от других экземпляров, или управление этим экземпляром возможно только из вышестоящего составного действия, для которого данный экземпляр является компонентом. По умолчанию атрибут independentInd должен иметь значение TRUE.

Пример — Определению действия иногда присваивается значение атрибута independentInd = FALSE, если деловые правила не разрешают назначать это действие, не назначая группу действий, которая его содержит.

Назначение может иметь компонент, который нельзя прервать независимо от других компонентов.

A.3.1.2.19 Act.uncertaintyCode:: CE (0..1)

Словарный домен: ActUncertainty

Код, указывающий, было ли в целом утверждение, передаваемое в экземпляре класса Act, объявлено как недостаточно точное.

Пример — Пациенту могли в прошлом сделать операцию холецистэктомии (однако он в этом не уверен).

Ограничения. Отсутствие точности, объявленное с помощью этого атрибута, относится к объединенному смыслу утверждения, передаваемого в экземпляре класса Act с помощью всех описательных атрибутов (например, Act.code, Act.effectiveTime, Observation.value, SubstanceAdministration.doseQuantity и т. д.), и к смыслу всех компонентов.

Обсуждение. Этот атрибут не предназначен для замены или конкуренции с отсутствием точности значения атрибута Observation.value или других отдельных атрибутов класса. Такие точечные указания отсутствия точности должны быть определены с помощью расширения типов данных PPD, UVP или UVN, применяемых к конкретному атрибуту. В частности, если отсутствие точности относится к значению количественного измерения, то его надо указать с помощью присваивания этому значению типа данных PPD<PQ>, а не с помощью атрибута uncertaintyCode. Если, к примеру, дифференциальные диагнозы перенумерованы или им присвоены вероятностные веса, то надо использовать типы данных UVP<CD> или UVN<CD>, а не атрибут uncertaintyCode. Использование атрибута uncertaintyCode возможно только в том случае, если точность всего действия и зависящих от него действий подвергаются сомнению.

Можно было бы подумать, что если точность совсем уж мала, то вместо атрибута uncertaintyCode надо воспользоваться атрибутом отрицания negationInd, однако эти два понятия совершенно независимы. Можно быть очень неуверенным в том, что событие имело место, но это не означает уверенности в его отрицании.

A.3.1.2.20 Act.reasonCode:: SET<CE> (0..*)

Словарный домен: ActReason

Определение. Код, указывающий мотивацию, причину, или логическое обоснование действия, если такое обоснование не было представлено с помощью ассоциации ActRelationship, у которой атрибут typeCode имеет значение «RSON» (has reason — имеет причину) и которая связывает данное действие с другим.

Пример — Примерами причин, которые могли бы заслуживать передачи в этом поле, служат «обычное назначение», «требование сообщить об инфекционном заболевании», «по запросу пациента», «требование закона».

Обсуждение. Большинство причин действий могут быть четко описаны с помощью связывания нового действие с предшествующим, использующего ассоциацию ActRelationship, у которой атрибут typeCode имеет значение «RSON». Такая связь означает, что предшествующее действие служит причиной для нового (см. описание класса ActRelationship). Это предшествующее действие может быть специфичным существующим действием или текстовым разъяснением. Такой подход пригоден для большинства случаев, и чем более специфична причина, тем более следует использовать ассоциацию ActRelationship, а не атрибут reasonCode.

Атрибут reasonCode остается как место для указания общих причин, которые не связаны с предшествующим действием или любыми другими условиями, выраженными с помощью экземпляров класса Act. Примером могут служить указания, что таковы требования закона или что причиной послужил запрос пациента и т. д. Но если требуется более точно сослаться на конкретную статью закона, правила, контракта или запроса пациента, то надо их представить в форме экземпляра класса Act (обычно так и делается) и не использовать атрибут reasonCode.

A.3.1.2.21 Act.languageCode:: CE (0..1)

Словарный домен: HumanLanguage

Определение. Основной язык, на котором описано данное действие, в особенности язык значения атрибута Act.text.

A.3.1.2.22 Переходы состояний экземпляра класса Act (атрибутом состояния является statusCode)

Диаграмма перехода состояний класса Act показана на рисунке А.5. Действие может иметь следующие состояния:

- aborted (прервано) — подсостояние состояния normal: активный объект услуги был неожиданно завершен;
- active (активно) — подсостояние состояния normal: объект услуги активен;
- cancelled (отменено) — подсостояние состояния normal: объект услуги был отменен до того, как стал активным;
- completed (завершено) — подсостояние состояния normal: объект услуги завершен;
- held (отложено) — подсостояние состояния normal: объект услуги, все еще находящийся на подготовительной стадии. Он не может стать активным, пока не будет выведен из этого состояния;
- new (новое) — подсостояние состояния normal: объект услуги, который готовится стать активным;
- normal (нормальное): охватывает все ожидаемые состояния объекта услуги, за исключением nullified и obsolete, которые представляют необычные терминальные состояния жизненного цикла;
- nullified (аннулировано): объект услуги не должен был создаваться, поэтому он аннулирован;
- obsolete (устарело): объект услуги заменен новым объектом;
- suspended (приостановлено) — подсостояние состояния normal: объект активной услуги временно приостановлен.
- Между состояниями действия возможны следующие переходы:
- abort (прервать) — из состояния active в состояние aborted;
- revise (пересмотреть) — из состояния active в состояние active;
- complete (завершить) — из состояния active в состояние completed;
- suspend (приостановить) — из состояния active в состояние suspended;
- reactivate (активировать заново) — из состояния completed в состояние active;
- revise (пересмотреть) — из состояния completed в состояние completed;
- cancel (отменить) — из состояния held в состояние cancelled;
- revise (пересмотреть) — из состояния held в состояние held;
- release (освободить) — из состояния held в состояние new;
- activate (активировать) — из состояния new в состояние active;
- cancel (отменить) — из состояния new в состояние cancelled;
- complete (завершить) — из состояния new в состояние completed;
- hold (задержать) — из состояния new в состояние held;
- revise (пересмотреть) — из состояния new в состояние new;
- nullify (аннулировать) — из состояния normal в состояние nullified;
- obsolete (сделать устаревшим) — из состояния normal в состояние obsolete;
- activate (активировать) — из начального (пустого) состояния в состояние active;
- complete (завершить) — из начального (пустого) состояния в состояние completed;
- create (создать) — из начального (пустого) состояния в состояние new;
- jump (перейти) — из начального (пустого) состояния в состояние normal;
- abort (прервать) — из состояния suspended в состояние aborted;
- resume (возобновить) — из состояния suspended в состояние active;
- complete (завершить) — из состояния suspended в состояние completed;
- revise (пересмотреть) — из состояния suspended в состояние suspended.

A.3.1.3 Класс: ActRelationship (в предметной области Acts)

Атрибуты класса ActRelationship:

- typeCode:: CS,
- inversionInd:: BL,
- contextControlCode:: CS,
- contextConductionInd:: BL,
- sequenceNumber:: INT,
- priorityNumber:: INT,
- pauseQuantity:: PQ,
- checkpointCode:: CS,
- splitCode:: CS,
- joinCode:: CS,
- negationInd:: BL,
- conjunctionCode:: CS,
- localVariableName:: ST,

- seperatableInd:: BL,
- setCode :: CS.

Ассоциации класса ActRelationship:

- source :: (1..1) Act :: outboundRelationship :: (0..*) (ассоциация с классом Act, роль outboundRelationship — исходящая связь),
- target :: (1..1) Act :: inboundRelationship :: (0..*) (ассоциация с классом Act, роль inboundRelationship — входящая связь).

Определение класса ActRelationship

Класс ActRelationship моделирует направленную ассоциацию между классом-источником Act и классом-целью Act. Ассоциация класса ActRelationship с классом-источником Act моделирует «исходящую» часть направленной ассоциации, а ассоциация класса ActRelationship с классом-целью Act моделирует «входящую» часть направленной ассоциации. Смысл и назначение экземпляра класса ActRelationship указаны в атрибуте ActRelationship. typeCode.

Примеры

Компонентами панели электролитов являются натрий, калий, pH и бикарбонат. Тогда экземпляр класса Act, описывающий панель электролитов, имел бы четыре «исходящие» ассоциации с экземплярами классов ActRelationship, у которых атрибут ActRelationship.typeCode имеет значение «COMP» (has component — имеет компонент).

Панель электролитов исследуется по направлению на лабораторный анализ. Тогда экземпляр класса Act, содержащий результат исследования, имел бы «исходящую» ассоциацию с экземпляром класса ActRelationship, у которого атрибут ActRelationship.typeCode имеет значение «FLFS» (fulfills — выполняет) и который имеет «входящую» ассоциацию с экземпляром класса Act, содержащим направление на анализ.

Операция «холецисэктомия» выполнена в связи с результатом исследования «желчекаменная болезнь». Тогда экземпляр класса Act, содержащий сведения об этой операции, имел бы «исходящую» ассоциацию с экземпляром класса ActRelationship, у которого атрибут ActRelationship.typeCode имеет значение «RSON» (has reason — имеет причину) и который имеет «входящую» ассоциацию с экземпляром класса Observation, в котором указан диагноз «желчекаменная болезнь».

Обсуждение. Каждый экземпляр класса ActRelationship можно рассматривать как стрелку с наконечником (упирающимся в целевой объект) и хвостовиком (упирающимся в объект-источник). Функции (иногда называемые «ролями»), которые выполняют экземпляры класса Act при таком связывании, определяются для каждого типа экземпляров класса ActRelationship по-разному. Когда экземпляр класса ActRelationship обеспечивать связь «композиция», то источник является составным объектом, а цель — его компонентом. Когда экземпляр класса ActRelationship обеспечивать причинно-следственную связь, то источником может быть любой экземпляр класса Act, а целью — экземпляр класса Act, описывающий причину появления источника.

Связи, ассоциированные с экземпляром класса Act, должны рассматриваться как свойства экземпляра-источника. Это означает, что автор экземпляра класса Act считается автором всех связей с этим экземпляром, имеющих его в качестве источника. Из этого правила нет исключений.

Более подробные сведения о различных типах экземпляров класса ActRelationship см. в описании атрибута ActRelationship.typeCode.

Класс ActRelationship используется для представления планов действий и клинических выводов или суждений о связи действий. Предшествующие действия могут быть связаны с вновь появившимися в качестве их причин. Известные факты могут быть связаны с текущими клиническими гипотезами в качестве их доказательства. Списки диагнозов и другие совокупности связанных суждений о клинических событиях могут быть образованы с помощью экземпляров класса ActRelationship.

Одним из наиболее распространенных применений класса ActRelationship является описаний композиций и декомпозиций действий, в которых экземпляры этого класса используются с атрибутом typeCode, имеющим значение «COMP» (has component — имеет компонент). Этот тип связи позволяет описывать действия с разной степенью детализации.

Связь композиции («COMP») позволяет группировать действия в «панели», например, панели LYTES, CHEM12 или CBC, с помощью которых можно сделать групповой заказ нескольких рутинных лабораторных анализов. Некоторые группировки, скажем, CHEM12, представляются более случайными, другие — более обоснованными, например, измерение артериального давления естественным образом состоит из систолического и диастолического давления.

Отношения композиции могут быть организованы в последовательности, чтобы формировать временные и условные (не временные) ряды планов действий (например, план лечения, критичные пути, протоколы клинических испытаний, протоколы медикаментозной терапии). Как в классе Act, так и в классе ActRelationship есть группа атрибутов, называемая «комплексом атрибутов рабочих процессов», с помощью которых можно формировать детальные спецификации планов выполняемых действий. Они имеют следующие функции:

- с помощью атрибута ActRelationship.sequenceNumber можно упорядочить компоненты экземпляра класса Act в форме последовательной или параллельной коллекции, выражая логические ветвления, а также парал-

лельные задачи (задачи, выполняемые в одно и то же время). С помощью атрибута ActRelationship.splitCode и ActRelationship.joinCode можно описывать выбор ветвей или параллельное выполнение задач;

- с помощью атрибутов Act.activityTime и ActRelationship.pauseQty можно явным образом задавать время выполнения планируемых действий. С помощью атрибута Act.repeatNumber можно задать повторение действия (цикл);

- с помощью экземпляров класса ActRelationship, у которых атрибут ActRelationship.typeCode имеет значение «PRCN» (has precondition — имеет предусловие), можно задавать условия выполнения шагов плана в зависимости от состояния или результата предшествующих действий. С помощью атрибута ActRelationhsip.checkpointCode можно указать, когда проверяется предусловие действия в процессе передачи управления.

С помощью экземпляров класса ActRelationship, описывающих отношения композиции, можно организовывать многие уровни вложения, позволяющие полностью обеспечить управление рабочими процессами. Такое вложение и такое использование атрибутов рабочих процессов сконструировано по аналогии с конструкциями языков структурного программирования, поддерживающих параллелизм (ветвление, соединение, прерывания) и не требующих применения операторов перехода GOTO. Важно отметить, что ВСЕ планы описываются с помощью упорядоченных компонентов (шагов) составного действия (блока), которые могут быть отображены с помощью диаграмм Насси-Шнейдерман (известных также как «щелевые схемы» (Chap Charts) или структограммы), а не в форме цепочек связанных действий, как в блок-схемах.

С помощью отношений композиции детали действия могут быть представлены на разных уровнях для разных целей, не требуя переупорядочения структуры иерархии классов Act. Это позволяет отображать разные точки зрения на один и тот же процесс деятельности. Например, с позиции платежной системы панель лабораторных анализов может считаться простой оплачиваемой услугой. С клинической позиции та же самая панель лабораторных анализов является совокупностью отдельных анализов независимо от того, как они были заказаны. Точка зрения лаборатории на эту панель может быть более детальной и включать в себя такие шаги, которые никогда не сообщаются клиницисту (центрифугирование, отбор, аликвотирование, использование определенных устройств и т. д.). Лабораторная точка зрения обеспечивает исчерпывающую спецификацию планов действий (которые могут быть автоматизированы). При составлении такой спецификации будут описаны все более и более вложенные подчиненные действия. Тем не менее действие остается тем же самым, только его детали скрываются с помощью декомпозиции до необходимого уровня.

Имея в виду природу варьирования деталей, действия можно назвать «фракталами», и они даже допускают больший уровень декомпозиции, подобно тому как движение руки робота можно представить в виде большого числа точных шагов управления.

Атрибуты класса ActRelationship

A.3.1.3.1 ActRelationship.typeCode:: CS (1..1) Mandatory

Словарный домен: ActRelationshipType

Определение. Код, указывающий смысл и назначение каждого экземпляра класса ActRelationship. Каждое из его значений предполагает определенное ограничение того, какие виды экземпляров класса Act могут быть связаны и каким именно способом.

Обсуждение. Типы экземпляров класса ActRelationship попадают в одну из 5 категорий:

1) Композиция или декомпозиция, с составным объектом (источником) и его компонентом (цель).

2) Продолжение, например, долечивание, выполнение назначения, конкретизация, замена, преобразование и т. д., имеющие то общее, что источником и целью связи являются экземпляры класса Act принципиально того же типа, но отличающиеся значениями атрибута moodCode и других атрибутов, а также то, что цель связи существует до того, как появился источник, и источник содержит ссылку на цель, а цель содержит обратную ссылку на этот источник.

3) Предусловие, триггер, причина, противопоказание, при которых условно выполняемое действие является источником связи, а условие или причина — целью.

4) Постусловие, результат, цель или риск, при которых действие-источник имеет связь с результатом или целью.

5) Множество функциональных связей, например, поддержка, причина, производное, обобщаемых понятием «отношение».

A.3.1.3.2 ActRelationship.inversionInd:: BL (0..1)

Определение. Признак, указывающий, что значение атрибута ActRelationship.typeCode должен быть интерпретировано таким образом, как если бы роли источника и целевых действий поменялись местами. Этот признак реверса используется, когда смысл значения атрибута ActRelationship.typeCode должен быть обращен.

A.3.1.3.3 ActRelationship.contextControlCode :: CS (0..1)

Словарный домен: ContextControl

Определение. Код, указывающий, какой вклад вносит данный экземпляр класса ActRelationship в контекст текущего экземпляра класса Act, и будет ли этот контекст распространяться на действия-потомки, чьи связи разрешают такое распространение (см. описание атрибута ActRelationship.contextConductionInd).

Обоснование. В целях сокращения дублирования люди склонны при интерпретации информации полагаться на ее контекст. Например, читая документ, извлеченный из папки, содержащей медицинскую карту пациента, пользователь сделает вывод, что этот документ относится к данному пациенту, даже если в нем нет прямого ука-

зания на принадлежность к этому пациенту. Однако другие части информации, например, автор папки (больница, в которой ведется эта медицинская карта), могут иногда применяться к содержанию папки (например, документ, составленный врачом этой больницы), а могут не применяться (например, в папку вложена копия документа, полученного из другого медицинского учреждения). Люди вполне хорошо делают необходимые выводы о том, какая часть контекста должна распространяться на элемент данных, а какая — нет. Однако случаются и неправильные выводы (например, в медицинскую карту пациента вложен документ с информацией о его родственнике). Более того, автоматизация подобных выводов представляет собой очень сложную задачу, даже если такие выводы необходимы для систем обеспечения принятия решений.

Обсуждение. С помощью этого атрибута можно точно указать, вносит ли ассоциация дополнение к контексту, связанному с конкретным элементом (например, добавляя дополнительного автора), или заменяет (отменяет) часть контекста, связанного с конкретным элементом (например, указывая единственного автора независимо от прежнего содержания элемента). С его помощью можно также указать, применяется ли ассоциация только к этому действию (не распространяемый контекст), или может применяться также и к производным действиям (распространяемый контекст).

Рассматриваемый атрибут тесно связан с атрибутом ActRelationship.contextConductionInd, указывающий, будут ли ассоциации, маркированные как распространяемые контекст, действительно распространять его на дочернее действие. Например, участие автора могло быть маркировано как распространяемое, но при этом не будет применяться к гиперссылке на внешний документ.

Если этот атрибут не имеет значения (например, у него пустое значение) или у него нет значения по умолчанию, то о контексте нельзя сделать никаких выводов. В системах могут использоваться собственные допущения на основе передаваемых данных. (По этой причине подкомитетам комитета HL7 рекомендуется в своих спецификациях указывать для этого атрибута значение по умолчанию или фиксированное значение, чтобы гарантировать его согласованную интерпретацию.)

Примеры

1 Пусть экземпляр класса Observation имеет исходящую ассоциацию к экземпляру класса ActRelationship, содержащему информацию об участии пациента в исследовании, и при этом атрибут ActRelationship.contextControlCode имеет значение «AP» (additive, propagating — аддитивная, распространяемая). Пусть этот же экземпляр класса Observation имеет связи с компонентами исследования, описанные с помощью экземпляров класса ActRelationship, которые маркированы как распространяющие контекст. Это означает, что участие пациента, описанное для родительского экземпляра класса Observation, распространяется и на эти компоненты исследования.

2 Пусть создано комплексное назначение, содержащее заказ в аптеку, а также заказ нескольких лабораторных анализов. Это комплексное назначение имеет связи участия с пациентом и автором назначения, а также связь с диагнозом, и каждая из этих связей маркирована как «аддитивная, распространяемая». Пусть связь между составным назначением и входящим в него заказом в аптеку маркирована как распространяющая связь (ее атрибут contextConductionInd имеет значение TRUE). При этом заказ в аптеку имеет связь участия с автором, маркированную как «AN» (additive, non-propagating — аддитивная, не распространяемая), и причинно-следственную связь с диагнозом, маркированную как «OP» (overriding, propagating — замещающая, распространяемая). Кроме того, заказ в аптеку имеет связь с информацией об отпуске лекарства, а также связь с протоколом медикаментозной терапии, которая маркирована как не распространяемая (ее атрибут contextConductionInd имеет значение FALSE). Такая совокупность объектов и связей трактуется следующим образом: заказ в аптеку интерпретируется как наследующий пациента от составного назначения. Он имеет двух авторов (одного, унаследованного от составного назначения, и другого, явно указанного автора заказа в аптеку). Диагнозом, в связи с которым сделан заказ в аптеку, будет считаться только тот, что связан с заказом в аптеку, а не с составным назначением. Событие отпуска унаследует пациента от составного назначения и диагноз заказа в аптеку, но не авторов. Протокол медикаментозной терапии не будет связан ни с пациентом, ни с диагнозом, ни с автором.

A.3.1.3.4 ActRelationship.contextConductionInd:: BL (0..1)

Определение. Если этот атрибут имеет значение TRUE, то связи родительского действия распространяются через данный экземпляр класса ActRelationship на дочернее действие.

Обсуждение. Распространяются только те связи, которые были добавлены к контексту родительского действия и были маркированы как «распространяемые» (см. описания атрибута contextControlCode в классах ActRelationship и Participation).

Идентификация экземпляра класса Act как родительского или дочернего (и, следовательно, идентификация направления распространения контекста) определяется перемещением по связи при ее сериализации. Первое обнаруженное действие рассматривается как родительское. Контекст распространяется через экземпляр класса ActRelationship на второе (дочернее) действие.

Обоснование и примеры указаны в описании атрибута ActRelationship.contextControlCode.

A.3.1.3.5 ActRelationship.sequenceNumber:: INT (0..1)

Определение. Целое значение, указывающее относительное положение данной связи среди других связей похожих типов, у которых источником является один и тот же экземпляр класса Act.

Обсуждение. Этот атрибут принадлежит к группе атрибутов управления рабочим процессом. План действия представляет собой составное действие, связанное с действиями-компонентами. В упорядоченном плане каждый экземпляр класса ActRelationship, связывающий составное действие с компонентом, имеет атрибут sequenceNumber, значение которого определяет порядок шагов плана. Если у нескольких компонентов значение атрибута sequenceNumber одинаково, то эти компоненты являются ветвями. Ветви могут быть исключающими (как в переключателе case) или могут допускать параллельное выполнение, что можно указать с помощью атрибута splitCode.

Если атрибут имеет пустое значение, то относительная позиция действия, описываемого целевым экземпляром класса Act, не задана (то есть он может быть на любом месте).

Чтобы указать относительную очередность связанных действий, вместо атрибута sequenceNumber следует использовать атрибут priorityNumber.

A.3.1.3.6 ActRelationship.priorityNumber:: INT (0..1)

Определение. Целое значение, указывающее относительный приоритет данной связи среди других связей похожих типов, у которых источником является один и тот же экземпляр класса Act. Связи с меньшими значениями атрибута priorityNumber рассматриваются раньше и выше тех, что имеют более высокие значения.

Пример — Если указано несколько критериев, то с помощью этого атрибута можно указать, какой из них должен быть рассмотрен раньше других. Если связи с компонентами имеют одно и то же значение атрибута sequenceNumber, то атрибут приоритета позволяет указать, какая из них должна быть рассмотрена раньше других. Если альтернативы или варианты выбираются людьми, то значение priorityNumber указывает предпочтение.

Обсуждение. Упорядочение может быть полным, при котором все значения приоритета уникальны, или частичным, при котором один и тот же приоритет назначается нескольким связям.

A.3.1.3.7 ActRelationship.pauseQuantity:: PQ (0..1)

Определение. Промежуток времени между готовностью действия к выполнению и фактическим началом его выполнения.

Обсуждение. Этот атрибут принадлежит к группе атрибутов управления рабочим процессом. План действия представляет собой составное действие, связанное с действиями-компонентами. В упорядоченном плане каждый экземпляр класса ActRelationship, связывающий составное действие с компонентом, имеет атрибут sequenceNumber, значение которого определяет порядок шагов плана. Если у шага есть предусловия, то его выполнение инициируется в том случае, когда они удовлетворяются. В этот момент запускается таймер со значением pauseQuantity, и действие начинает выполняться, когда пройдет время, указанное в атрибуте pauseQuantity.

A.3.1.3.8 ActRelationship.checkpointCode:: CS (0..1)

Словарный домен: ActRelationshipCheckpoint

Определение. Код, указывающий моменты проверки выполнения предусловия действия, (например, перед тем, как действие начнется впервые, после каждого повторения действия, но не перед первым, или в процессе всего времени действия).

Обсуждение. Этот атрибут принадлежит к группе атрибутов управления рабочим процессом. План действия представляет собой составное действие, связанное с действиями-компонентами. В упорядоченном плане каждый экземпляр класса ActRelationship, связывающий составное действие с компонентом, имеет атрибут sequenceNumber, значение которого определяет порядок шагов плана. Если у шага есть предусловия, то его выполнение инициируется в том случае, когда они удовлетворяются. С помощью атрибута repeatNumber можно указать, что выполнение действия может повторяться. А с помощью атрибута checkpointCode можно указать, когда проверяется предусловие, что аналогично различным условным операторам и циклам в языках программирования: while-do по сравнению с do-while или repeat-until по сравнению с loop-exit.

Для всех значений атрибута checkpointCode, кроме «E» (end — конец), предусловия проверяются в момент завершения предшествующего шага плана при условии, что данный шаг является следующим согласно значению атрибута sequenceNumber.

Если атрибут checkpointCode критерия повторяющегося действия имеет значение «E» (end — конец), то этот критерий проверяется только в конце каждого повторения действия. Если критерий повторения удовлетворен, то следующее повторение действия готово к выполнению.

Если атрибут checkpointCode имеет значение «S» (entry — вход), то критерий проверяется в начале каждого повторения (если таковые имеются), при этом «начало» означает, что критерий проверяется однократно при старте «циклического» повторения.

Если атрибут checkpointCode имеет значение «T» (through — в течение), то оно задает особый случай, когда критерий проверяется в процессе повторения. Как только критерий перестал выполняться, должно быть инициировано событие прерывания действия (см. описание атрибута Act.interruptibleInd) и в принципе действие должно быть завершено.

Атрибут checkpointCode со значением «X» (exit — выход) используется в специальном шаге плана, представляющем выход из цикла. С его помощью можно обеспечить завершение плана действий в связи с выполнением

определенного условия, проверяемого при выполнении этого плана. Такие критерии выхода упорядочены относительно других компонентов плана с помощью атрибута ActRelationship.sequenceNumber.

A.3.1.3.9 ActRelationship.splitCode:: CS (0..1)

Словарный домен: ActRelationshipSplit

Определение. Код, указывающий, какие ветви плана действия выбираются среди других ветвей.

Обсуждение. Этот атрибут принадлежит к группе атрибутов управления рабочим процессом. План действия представляет собой составное действие, связанное с действиями-компонентами. В упорядоченном плане каждый экземпляр класса ActRelationship, связывающий составное действие с компонентом, имеет атрибут sequenceNumber, значение которого определяет порядок шагов плана. Если для нескольких компонентов значение атрибута sequenceNumber одинаково, то эти компоненты являются ветвями. Атрибут splitCode указывает, является ли ветвь исключающей (как в переключателе case) или включающей, т. е. может допускать параллельное выполнение других ветвей.

В дополнение к исключающему и включающему ветвлению с помощью атрибута splitCode можно указать, как проверяется предусловие (называемое также «сторожевым условием») ветви. Сторожевое условие может проверяться однократно при переходе к ветви, и если оно не выполнено, то ветвь отвергается. В другом варианте выполнение ветви может быть отложено, пока это условие не будет выполнено.

При исключающем ветвлении с ожиданием первая ветвь, у которой ее условие ветвления станет выполненным, начнет выполняться, а все остальные ветви будут отвергнуты. При включающем ветвлении с ожиданием некоторые ветви могут уже выполняться, в то время как другие все еще будут ожидать, пока их сторожевое условие не будет выполнено.

A.3.1.3.10 ActRelationship.joinCode:: CS (0..1)

Словарный домен: ActRelationshipJoin

Определение. Код, указывающий способ восстановления синхронизации параллельно выполняемых действий.

Обсуждение. Этот атрибут принадлежит к группе атрибутов управления рабочим процессом. План действия представляет собой составное действие, связанное с действиями-компонентами. В упорядоченном плане каждый экземпляр класса ActRelationship, связывающий составное действие с компонентом, имеет атрибут sequenceNumber, значение которого определяет порядок шагов плана. Если для нескольких компонентов значение атрибута sequenceNumber одинаково, то эти компоненты являются ветвями. Ветви могут выполняться параллельно, если атрибут splitCode указывает, что в одно и то же время может исполняться более одной ветви. В этом случае с помощью атрибута joinCode можно указать, будет ли восстанавливаться синхронизация ветвей, и если да, то каким образом.

Основные способы восстановления синхронизации следующие:

- поток управления ждет, пока выполнение каждой ветви не завершится (ожидание ветвей);
- как только выполнится одна ветвь, выполнение остальных ветвей прекращается (прекращение ветвей);
- синхронизация ветвей не восстанавливается, и они продолжают выполняться (отложенные ветви).

Прекращение ветвей происходит только в том случае, когда имеется, как минимум, одна активная ожидающая (или исключающая ожидающая) ветвь. Если активных ожидающих ветвей нет, то процесс прекращения ветвей не инициируется (а не прекращается после инициации). Поскольку отложенная ветвь не связана с другими ветвями, наличие активных отложенных ветвей не мешает прекращению другой ветви.

A.3.1.3.11 ActRelationship.negationInd:: BL (0..1)

Определение. Признак, указывающий, что к значению связи применяется отрицание.

Пример — Если исходная связь указывает, что у действия А есть компонент В, то ее отрицание означает, что действие В не является компонентом действия А. Если действие В описывает причину действия А, то отрицание означает, что действие В не является причиной действия А. Если действие В описывает предусловие действия А, то отрицание означает, что действие В не является предварительным условием действия А.

Обсуждение. Как показывают примеры, использование этого атрибута довольно ограничено, особенно по сравнению с атрибутом Act.negationInd, с помощью которого фактически можно указать, что описанное действие не существует, не выполняется и т. д., в то время как с помощью атрибута ActRelationship.negationInd можно лишь отрицать данную связь между действием-источником и действием-целью действия, а не изменить смысл каждого действия. Поэтому данный атрибут используется, главным образом, в разъяснительных целях.

Учите также различие между отрицанием и противоположностью. Противопоказание является противоположностью показания (причины). То обстоятельство, что наличие боли в пояснице не является причиной назначения антибиотиков, не означает, что боль в пояснице является противопоказанием для применения антибиотиков.

A.3.1.3.12 ActRelationship.conjunctionCode:: CS (0..1)

Словарный домен: RelationshipConjunction

Определение. Код, указывающий логическое соединение критериев, описанных условными связями между действиями (например, AND («и»), OR («или»), XOR («исключающее или»)).

Ограничения. Все критерии, соединенные с помощью AND, должны выполняться. Если в одну группу входят критерии, соединяемые с помощью OR и AND, то должны выполняться хотя бы один из критериев OR и все крите-

рии AND. Если в одну группу входят критерии, соединяемые с помощью XOR, OR и AND, то должен выполняться ровно один критерий XOR, по крайней мере один из критериев OR и все критерии AND. Другими словами, множества критериев AND, OR и XOR, в свою очередь, объединены логическим оператором AND (все критерии AND, по крайней мере один из критериев OR и ровно один критерий XOR). Если требуется иное, то можно воспользоваться вложенными критериями.

A.3.1.3.13 ActRelationship.localVariableName:: ST (0..1)

Определение. Строка символов, указывающая имя входного параметра, по значению которого экземпляр класса Act, который служит источником ассоциации с данным экземпляром класса ActRelationship, может вычислить некоторые из своих атрибутов. Областью действия имени локальной переменной является значение атрибута Act.derivationExpr, который может содержать это имя в качестве входного параметра.

A.3.1.3.14 ActRelationship.seperatableInd:: BL (0..1)

Определение. Этот индикатор указывает, должно ли действие-источник связи интерпретироваться независимо от целевого действия связи. Его значение не может препятствовать человеку или приложению обрабатывать действия независимо друг от друга, оно лишь обозначает желание и готовность автора заверить содержание действия-источника, если оно будет отделено от содержания целевого действия. Имейте в виду, что типичным значением этого атрибута по умолчанию будет TRUE. Также учтите, что этот атрибут ортогонален механизму наследования контекста и никак с ним не связан. Если контекст действия распространен на вложенные действия, то тем самым предполагается, что эти вложенные действия не могут интерпретироваться без распространенного контекста.

A.3.1.3.15 ActRelationship.subsetCode:: CS (0..1)

Словарный домен: ActRelationshipSubset

Определение. Код, указывающий, что целевое действие связи является фильтрованное подмножество общего связанного множества целевых действий.

Используется для ограничения числа компонентов до подмножества, состоящее из первого, последнего или следующего компонента, а также из суммарного множества, усредненного множества или иным образом отфильтрованного либо вычисленного подмножества.

A.3.1.4 Класс ControlAct (в предметной области Acts)

Ассоциации класса ControlAct:

- payload::(0..*) Message::controlAct::(0..1) (ассоциация с классом Message, роль controlAct — управляющее действие),
- queryEvent::(0..1) QueryEvent::controlAct::(1..1) (ассоциация с классом QueryEvent, роль controlAct — управляющее действие).

Класс ControlAct является специализацией класса Act.

Определение класса ControlAct

Действие, приводящее к изменению состояния другого класса, пользовательское событие (например, запрос) или системное событие (например, выполняемое по таймеру).

Примеры

1 Выписка пациента (состояние класса Encounter (контакт с пациентом) изменяется с Active (активный) на Completed (завершенный)).

2 Прекращение применения лекарственного средства (состояние класса SubstanceAdministration (применение лекарства) изменяется с Active (активное) на Aborted (прекращенное)).

3 Передача ежедневной сводки (событие, выполняемое по таймеру).

Обсуждение. Этот класс воплощает понятие «события, инициирующего взаимодействие» и поэтому должен присутствовать при каждом обмене сообщениемям (в силу кратности 1..1 между событием и взаимодействием). Однако управляющие действия могут также присутствовать в информации, передаваемой в сообщении. Примером может служить последовательность управляющих действий, связанных с лабораторным заказом и идентифицирующих события его обработки (вначале создание, затем пересмотр, приостановка, возобновление и, наконец, завершение).

A.3.1.5 Класс DeviceTask (в предметной области Acts)

Свойства класса DeviceTask

Атрибуты класса DeviceTask:

- parameterValue :: LIST<ANY>.

Класс DeviceTask является специализацией класса Act.

Определение класса DeviceTask

Активность автоматизированной системы.

Обсуждение. Такая активность возникает или по внешней команде или самопроизвольно планируется и выполняется устройством (например, регулярная калибровка или промывка). У экземпляра класса, содержащего команду устройству, атрибут moodCode должен иметь значение «ORD» (order — заказ), у экземпляра с описанием выполнямого или выполненного действия атрибут moodCode должен иметь значение «EVN» (event — событие), у экземпляра с описанием автоматически инициируемой задачи атрибут moodCode должен иметь значение «APT» (выполнение по графику).

Атрибуты класса DeviceTask

A.3.1.5.1 DeviceTask.parameterValue :: LIST<ANY> (0..*)

Определение. Список параметров задачи, передаваемых устройству при выдаче команды (или используемых для планирования самопроизвольно выполняемых задач).

Обоснование. Некоторые параметры задач уникальны для конкретной модели оборудования. Наиболее критичные аргументы задачи (например, привести контейнер в движение, позиционировать его, выполнять действия по графику) указаны в структуре, стандартизованной комитетом HL7, и данный список параметров не должен использоваться для их передачи. Он нужен только в тех случаях, когда передаются не стандартированные параметры, уникальные для конкретной модели оборудования.

Примечание — Тем самым следует, что семантика и интерпретация списка parameterValue могут быть осуществлены только исходя из содержания спецификации данного устройства или его документации. Это содержание не является частью сообщения.

Ограничения. Параметры задаются в этом списке только в том случае, если они отсутствуют в других структурах, определенных в стандарте HL7. Список может содержать любые значения, интерпретируемые устройством. Они должны иметь тип данных из числа определенных в стандарте HL7 (например, коды номинальных значений, к примеру, флагов, типы данных REAL и INT для чисел, TS для моментов времени, PQ для размерных физических величин и т. д.). Однако помимо типизации значений иные характеристики параметров являются прозрачными для стандартов HL7.

A.3.1.6 Класс DiagnosticImage (в предметной области Acts)

Свойства класса DiagnosticImage

Атрибуты класса DiagnosticImage:

- subjectOrientationCode :: CE.

Класс DiagnosticImage является специализацией класса Observation.

Определение класса DiagnosticImage

Исследование, непосредственным и основным результатом которого (постусловие) являются новые данные о субъекте в форме визуализируемых изображений.

Атрибуты класса DiagnosticImage

A.3.1.6.1 DiagnosticImage.subjectOrientationCode :: CE (0..1)

Словарный домен: ImagingSubjectOrientation

Определение. Код, характеризующий пространственное расположение исследуемого объекта по отношению к рентгеновской пленке или детектору.

A.3.1.7 Класс Diet (в предметной области Acts)

Свойства класса Diet

Атрибуты класса Diet:

- energyQuantity :: PQ,
- carbohydrateQuantity :: PQ.

Класс Diet является специализацией класса Supply.

Определение класса Diet

Действие кормления субъекта или предоставления ему питания.

Обсуждение. Детали диеты передаются в экземпляре класса Material, ассоциированного с помощью экземпляра класса Participation, у которого атрибут typeCode имеет значение «PRD» (product — товар). Номер диеты, имеющий медицинское значение, можно передать в атрибуте Diet.code, однако детали пищи, входящей в диету, и различные сочетания блюд должны передаваться в форме экземпляров класса Material.

Примеры — Безглютеновая диета, бессолевая диета.

Атрибуты класса Diet

A.3.1.7.1 Diet.energyQuantity:: PQ (0..1)

Определение. Предоставляемая биологическая энергия (калории) в день.

Обсуждение. Должно существовать преобразование единицы измерения этой физической величины в ккал/день (или кДж/день). Обратите внимание на возможную путаницу между «большой калорией» и «малой калорией». Этикетки на продуктах питания указывают калорийность в «больших калориях». Однако в качестве единицы измерения лучше использовать малую калорию, являющуюся 1/1000 большой калории. Это различие строго проводится в таблицах единиц измерения, включенных в стандарт HL7.

A.3.1.7.2 Diet.carbohydrateQuantity:: PQ (0..1)

Определение. Предоставляемое количество углеводов (грамм) в день.

Обсуждение. Для диабетической диеты типично ограничение количества усваиваемых углеводов в день (например, 240 г/д). Это ограничение может быть передано как значение атрибута carbohydrateQuantity.

A.3.1.8 Класс FinancialContract (в предметной области Acts)

Свойства класса FinancialContract

Атрибуты класса FinancialContract

- paymentTermsCode:: CE.

Класс FinancialContract является специализацией класса Act.

Определение класса FinancialContract

Контракт, имеющий денежное выражение.

Пример — Договор страхования, соглашение о закупках.

Атрибуты класса FinancialContract

A.3.1.8.1 FinancialContract.paymentTermsCode:: CE (0..1)

Словарный домен: PaymentTerms

Определение. Сроки платежа по контракту или обязательствам.

Пример — «В течение 30 дней с даты выставления счета», «по получению счета», «по завершению услуги».

A.3.1.9 Класс FinancialTransaction (в предметной области Acts)

Свойства класса FinancialTransaction

Атрибуты класса FinancialTransaction:

- amt:: MO,
- creditExchangeRateQuantity:: REAL,
- debitExchangeRateQuantity:: REAL.

Класс FinancialTransaction является специализацией класса Act.

Определение класса FinancialTransaction

Действие, представляющее движение денежных сумм между двумя счетами.

Обсуждение. Финансовые операции всегда осуществляются между двумя счетами (дебет и кредит), но могут быть случаи, когда один или оба счета определяются более общей моделью или наследуются от нее.

Если у экземпляра класса FinancialTransaction атрибут moodCode имеет значение ORD (order — заказ), то этот экземпляр трактуется как требование выполнения операции.

Если атрибут moodCode имеет значение «EVN» (event — событие), то этот экземпляр класса FinancialTransaction содержит информацию об уже выполненной операции.

Примеры — Затраты на услугу; оплата услуги; оплата счета.

Атрибуты класса FinancialTransaction

A.3.1.9.1 FinancialTransaction.amt:: MO (0..1)

Определение. Указывает денежную сумму, перемещаемую с дебета на кредит счета.

Обсуждение. Если денежные единицы суммы отличаются от денежных единиц дебита или кредита счета, то должен быть указан используемый обменный курс.

A.3.1.9.2 FinancialTransaction.creditExchangeRateQuantity:: REAL (0..1)

Определение. Десятичное число, указывающее обменный курс между валютой кредита счета и валютой операции.

Обоснование. С помощью этого атрибута операция может быть указана в валюте, отличающейся от валюты кредита и дебета. Это также позволяет вести дебет и кредит в разных валютах.

Пример — При покупке услуг, оцененных в мексиканских песо (MXP) и оплаченных в долларах США (USD) со счета, который ведется в канадских долларах (CAD), в качестве обменного курса валюты кредита должно быть передано десятичное число r, для которого «y (USD) * r = x (CAD)».

A.3.1.9.3 FinancialTransaction.debitExchangeRateQuantity:: REAL (0..1)

Определение. Десятичное число, указывающее обменный курс между валютой дебета счета и валютой операции.

Обоснование. С помощью этого атрибута операция может быть указана в валюте, отличающейся от валюты кредита и дебета. Это также позволяет вести дебет и кредит в разных валютах.

Пример — При покупке услуг, оцененных в мексиканских песо (MXP) и оплаченных в долларах США (USD) со счета, который ведется в канадских долларах (CAD), в качестве обменного курса валюты дебита должно быть передано десятичное число r, для которого «y (USD) * r = x (MXP)».

A.3.1.10 Класс InvoiceElement (в предметной области Acts)

Свойства класса InvoiceElement

Атрибуты класса InvoiceElement:

- modifierCode:: SET<CE>,
- unitQuantity:: RTO<PQ,PQ>,
- unitPriceAmt:: RTO<MO,PQ>,
- netAmt:: MO,
- factorNumber:: REAL,
- pointsNumber:: REAL.

Класс InvoiceElement является специализацией класса Act.

Определение класса InvoiceElement

Действие, представляющее объявление и расшифровка «причитающейся суммы».

Обсуждение. Эта информация является частью «расшифровки» счета. Она часто объединяется с информацией о финансовой операции, представляющей сумму, подлежащую оплате, сумму, согласованную к оплате, или фактически оплаченную сумму.

Чтобы разбить один элемент счета-фактуры, передаваемый в экземпляре класса InvoiceElement, на несколько составляющих элементов, можно использовать рекурсивные отношения.

Если у экземпляра класса InvoiceElement атрибут moodCode имеет значение «DEF» (definition — определение), то этот экземпляр описывает «возможное» согласование строки счета-фактуры при его будущем рассмотрении. Если атрибут moodCode имеет значение «RQO» (request — требование), то этот экземпляр класса InvoiceElement содержит запрос на определение причитающейся суммы. Если атрибут moodCode имеет значение «EVN» (event — событие), то этот экземпляр класса InvoiceElement содержит информацию о сумме, которую должен оплатить конкретный получатель.

Атрибуты класса InvoiceElement

A.3.1.10.1 InvoiceElement.modifierCode:: SET<CE> (0..*)

Словарный домен: InvoiceElementModifier

Определение. Указывает модификатор кода, передаваемого в атрибуте code, для представления дополнительной информации об элементе счета-фактуры.

Обоснование. Этот модификатор не рассматривается как часть пре-координированной классификации с кодом, передаваемым в атрибуте code, поскольку система кодирования значений атрибута modifierCode не обязательно специально является детализацией системы кодирования атрибута code. Это не соответствует смыслу имени modifier (модификатор), поскольку обычно словарный домен модификатора должен быть определен как часть базового системы кодирования или должен быть разработан специально для нее.

Пример — Отдаленная территория, внеурочное обслуживание.

A.3.1.10.2 InvoiceElement.unitQuantity:: RTO<PQ,PQ> (0..1)

Определение. Описание количества товара или услуги, которое включено в счет или подлежит включению в счет.

Примеры — 4 часа, 4 мг, 4 коробки, 15 штук из контейнера, содержащего 1000 штук, и т.д.

Обсуждение. Каждый экземпляр класса InvoiceElement, описывающий элемент, включенный в счет-фактуру или подлежащий включению в счет, идентифицируется кодом товара или услуги, передаваемым в атрибуте InvoiceElement.code. В некоторых случаях этот код берется из пре-координированной классификации и идентифицирует контейнер (например, универсальный код продукта (УКП), присвоенный контейнеру, содержащему 1000 таблеток, и другой УКП-код для контейнера, содержащего 100 таких же таблеток). УКП-код используется при выставлении счетов, но при его применении возникает необходимость указать в форме дроби, что только часть контейнера (например, флакона) подлежит оплате или включению в счет. Если товар, информация о котором передается в экземпляре класса InvoiceElement, не является контейнером, то знаменатель дроби не указывается.

Например, пусть подлежат оплате 15 таблеток из контейнера, содержащего 1000 таблеток. В этом случае числитель дроби может быть указан как «15 {таблеток}» или просто «15», а знаменатель как «1000 {флакон}» или просто «1000» (см. обсуждение, следующее за обоснованием использования описательного текста для исчисляемых величин).

Ограничения. Единицы товара или услуги должны быть ограничены такими измеряемыми единицами, как литры, миллиграммы или часы. Не измеряемые, но исчисляемые единицы, например, короба, пакеты, посещения, таблетки и контейнеры, не должны указываться в компоненте единиц измерения типа данных PQ иначе как в аннотации, указанной в фигурных скобках ({xxx}). См. спецификацию типов данных в документе Data Types Part II Unabridged Specification, Appendix A: Unified Code for Units of Measure.

Указание исчисляемых единиц может быть выполнено с помощью следующих методов:

- определить исчисляемую единицу в атрибуте InvoiceElement.code. В этом случае конкретное значение атрибута InvoiceElement.code будет содержать указание, что предмет, описанный в данном экземпляре класса InvoiceElement, является упаковкой, содержащей 20 элементов. Для упаковки, содержащей 40 элементов, должен использоваться другой код, и т. д.

Например, если значение InvoiceElement.code соответствует упаковке, содержащей 20 элементов, а значение атрибута количества InvoiceElement.unitQuantity = 2 (без единиц измерения), то данный экземпляр класса InvoiceElement описывает 2 упаковки по 20 элементов в каждой, т. е. всего 40 элементов;

2) если требуется указать больше деталей (например, описать состав, упаковку, производителя товара), то эта информация должна быть описана как экземпляр класса Entity, связанный с данным экземпляром класса InvoiceElement с помощью экземпляров классов Role и Participation, при этом атрибут Participation.typeCode должен иметь значение «PRD» (product — товар).

A.3.1.10.3 InvoiceElement.unitPriceAmt:: RTO<MO,PQ> (0..1)

Определение. Стоимость единицы товара или услуги.

Ограничения. При указании отношения числитель должен иметь тип данных MO, а знаменатель — тип данных PQ. Детали указания отношения см. в описании атрибута unitQuantity.

Примеры — $0.20/мг; $250/день; $50.

A.3.1.10.4 InvoiceElement.netAmt:: MO (0..1)

Определение. Указывает полную стоимость элемента счета-фактуры, включая суммы его компонентов.

Обсуждение. Для листовых элементов счета-фактуры эта сумма вычисляется по формуле «unitQuantity*unitPriceAmt[*factorNumber][*pointsNumber]». Для группировок элементов счета-фактуры эта сумма вычисляется как сумма значений атрибутов netAmt всех элементов группы.

A.3.1.10.5 InvoiceElement.factorNumber:: REAL (0..1)

Определение. Указывает коэффициент, используемый при определении полной стоимости оказанных услуг и/или полученных товаров.

Пример — 10 (число услуг в качестве единиц) * $3.00 (стоимость единицы) * 1.5 (коэффициент) = $45.00 (сумма).

Обсуждение. Это понятие часто используется в Европе, чтобы устанавливать разные цены услуги для обязательного и добровольного медицинского страхования.

Простейшая формула для вычисления полной суммы такова: «unitQuantity * unitPriceAmount = netAmt».

С помощью коэффициента можно учесть скидки или доплаты, применяемые к полной сумме. Например, с учетом коэффициента формула для вычисления полной суммы примет следующий вид: «unitQuantity * unitPrice (цена единицы или балла) * factorNumber = netAmt».

См. аналогичное примечание к описанию атрибута InvoiceElement.pointsNumber, когда стоимость вычисляется с помощью баллов. При применении коэффициента и баллов формула для вычисления полной суммы примет следующий вид: «unitQuantity * unitPriceAmt * pointsNumber * factorNumber = netAmt».

A.3.1.10.6 InvoiceElement.pointsNumber:: REAL (0..1)

Определение. Этот атрибут используется в ситуации, когда количество товара или услуги выражается в «баллах», позволяющих задать весовой коэффициент к стоимости товара или услуги (основанный на трудоемкости, стоимости и/или интенсивности ресурса).

Пример — 5 (число услуг в качестве единиц) * 3 (число баллов, присвоенных одной услуге) * $20.00 (стоимость балла) = $300.00 (сумма).

Обсуждение. Такой подход широко используется в системах, где предоставленные услуги оцениваются в «условных единицах трудоемкости» (УЕТ), которым назначается фиксированная цена. В этом случае организация может пересматривать цены услуг, увеличивая или уменьшая стоимость УЕТ, чтобы учесть инфляцию, перегрузку и т. д.

Простейшая формула для вычисления полной суммы такова: «unitQuantity * unitPriceAmount = netAmt».

Понятие баллов может использоваться для расценки услуг и/или товаров, при которой количество услуг или товара измеряется в баллах, а одному баллу назначается определенная стоимость в долларах. В этом случае для расчета полной стоимости используется следующая формула: «unitQuantity * pointsNumber * unitPriceAmt (цена единицы или балла) = netAmt».

См. соответствующее примечание к описанию атрибута factorNumber. При использовании коэффициентов и баллов для вычисления полной суммы применяется следующая формула: «unitQuantity * unitPriceAmt * pointsNumber * factorNumber = netAmt».

A.3.1.11 Класс ManagedParticipation (в предметной области Acts)

Свойства класса ManagedParticipation

Атрибуты класса ManagedParticipation:

- id:: SET<II>,
- statusCode:: CS>.

Класс ManagedParticipation является специализацией класса Participation.

Переходы состояний класса ManagedParticipation описаны в подразделе A.3.1.11.3.

Определение класса ManagedParticipation

Участие, которое может меняться с течением времени, в связи с чем его состоянием и идентичностью надо управлять.

Обоснование. Класс ManagedParticipation определен как подкласс класса Participation, чтобы явно указать, что не все участия не имеют состояния. В общем случае, если о подзадаче, реализуемой с помощью участия, надо иметь больше информации, и этой подзадачей надо управлять, то вместо применения парадигмы участия НАДО моделировать эту подзадачу как компонент основного действия, описываемого классом Act.

Однако в некоторых случаях представление о том, в чем именно состоят эти подзадачи и что именно выполняют участники, является не очень определенным и поэтому их моделирование в форме компонентов оказывается неоднозначным или затруднительным.

Для таких случаев предназначен класс ManagedParticipation, который расширяет базовый класс Participation двумя атрибутами: идентификатором id и кодом состояния statusCode. Классы ManagedParticipation должны при-

меняться с чрезвычайными предосторожностями, чтобы избежать путаницы с действиями, моделируемыми в виде классов Act, и не создавать для участий инфраструктуру управления, подобную той, что уже существует для действий.

Пример — Лечащий врач госпитализированного пациента может уйти в отпуск, поэтому важно зна ть, когда его участие возобновится.

Атрибуты класса ManagedParticipation

A.3.1.11.1 ManagedParticipation.id:: SET<II> (0..*)

Определение. Уникальный идентификатор, с помощью которого можно идентифицировать конкретный экземпляр класса ManagedParticipation среди всех экземпляров этого класса, имеющих ассоциации с тем же самым экземпляром класса Act и с тем же самым экземпляром класса Role. (См. описание класса ManagedParticipation.)

A.3.1.11.2 ManagedParticipation.statusCode:: CS (0..1)

Словарный домен: ManagedParticipationStatus

Определение. Код, указывающий, описывает ли экземпляр класса ManagedParticipation готовящееся, активное, завершенное или отмененное участие. (См. описание класса ManagedParticipation.)

Совет разработчикам: в исходной модели RIM этот атрибут был определен как повтоящийся, чтобы можно было отразить наличие вложенных подсостояний, указанных на диаграмме перехода состояний. На практике, однако, необходимость передачи нескольких значений состояния никогда не возникала. Поэтому комитетам рекомендуется ограничить кратность этого атрибута до 1 во всех конструкциях сообщений.

A.3.1.11.3 Переходы состояний класса ManagedParticipation

Диаграмма перехода состояний класса ManagedParticipation показана на рисунке .7–. Управляемое участие может иметь следующие состояния:

- active (активно) — подсостояние состояния normal: это состояние отражает тот факт, что участие продолжается;
- cancelled (отменено) — подсостояние состояния normal: участие было отменено до того, как стало активным;
- completed (завершено) — подсостояние состояния normal: участие успешно завершено;
- normal (нормальное): «типичное» состояние. Исключает состояние nullified, которое указывает, что экземпляр участия был создан по ошибке;
- nullified (аннулировано): это состояние является терминальным состоянием экземпляра участия, созданного по ошибке;
- pending (готовящееся) — подсостояние состояния normal: это состояние отражает тот факт, что участие еще не стало активным.

Между состояниями действия возможны следующие переходы:

- revise (пересмотреть) — из состояния active в состояние active;
- complete (завершить) — из состояния active в состояние completed;
- reactivate (активизировать заново) — из состояния completed в состояние active;
- revise (пересмотреть) — из состояния completed в состояние completed;
- nullify (аннулировать) — из состояния normal в состояние nullified;
- create (создать) — из начального (пустого) состояния в состояние active;
- create (создать) — из начального (пустого) состояния в состояние completed;
- create (создать) — из начального (пустого) состояния в состояние pending;
- activate (активизировать) — из состояния pending в состояние active;
- cancel (отменить) — из состояния pending в состояние cancelled;
- revise (пересмотреть) — из состояния pending в состояние pending.

A.3.1.12 Класс Observation (в предметной области Acts)

Свойства класса Observation

Атрибуты класса Observation:

- value :: ANY,
- interpretationCode :: SET<CE>,
- methodCode :: SET<CE>,
- targetSiteCode :: SET<CD>.

Класс Observation является специализацией класса Act.

Класс Observation является генерализацией следующих классов:

- DiagnosticImage,
- PublicHealthCase.

Определение класса Observation

Действие исследования, т. е. получения информации о субъекте и сообщения этой информации. Непосредственным и основным результатом исследования (постусловие) является новая информация о субъекте. Исследования нередко осуществляются с помощью измерений или других сложных методов, но иногда сводятся к простым утвердительным заявлениям.

Обсуждение. Структура многих результатов исследований может быть представлена в форме пар «имя-значение», при этом атрибут Observation.code (унаследованный от класса Act) — это имя свойства, а атрибут

Observation.value — значение свойства. Такая конструкция часто называется «переменной» (т. е. именованное свойство, которому может быть присвоено значение). Таким образом, класс Observation всегда используется для передачи общих пар «имя-значение», т. е. переменных, даже если вычисление переменной и не является результатом сложного метода исследования. Оно может быть просто ответом на вопрос, или утверждением, или присваиванием значения параметру.

Как и все другие специализации класса Act, класс Observation используется для описания того, что сделано, и в случае класса Observation такое описание указывает, что было в действительности выявлено («результаты» или «ответы»), и эти «результаты» или «ответы» являются частью исследования и не расщепляются на объекты других типов.

Исследование может состоять из нескольких других исследований (компонентов), у каждого из которых свои значения атрибутов Observation.code и Observation.value. В этом случае составное исследование может не иметь собственного атрибута Observation.value. Например, дифференцированный подсчет белых клеток крови включает в себя отдельные результаты подсчета гранулоцитов, лейкоцитов и других нормальных или аномальных клеток крови (к примеру, незрелых клеток). Поэтому экземпляр класса Observation, описывающий составное исследование (подсчет белых клеток крови), может не иметь собственного значения (хотя его и можно получить, просуммировав все счетчики-компоненты). Таким образом, всякое действие, которое по своей природе является действием получения информации о субъекте и сообщения этой информации, представляется в форме экземпляра класса Observation независимо от того, имеет ли оно собственное простое значение или состоит из нескольких других исследований.

Диагнозы, наблюдения, симптомы и тому подобное также представляются классами Observation. Атрибут Observation.code (см. определение класса Observation) задает вид диагноза (например, «основные жалобы» или «диагноз при выписке», а атрибут Observation.value содержит код диагноза или симптома.

Атрибуты класса Observation

A.3.1.12.1 Observation.value:: ANY (0..*)

Определение. Информация, которая создана или определена действием исследования.

Ограничения. Если иное не указано, то атрибут Observation.value может иметь любой тип данных.

Тип данных, который может принимать значение атрибута Observation.value, зависит от вида исследования и обычно указывается в описании исследования или определяется с помощью простого правила по виду исследования, передаваемому в атрибуте Observation.code.

Ниже приведены общие рекомендации по выбору подходящего типа данных:

1) в количественных результатах, в основном, используется тип данных физической величины (PQ). Этот тип данных, по существу, представляет собой вещественное число с единицей измерения. Это общие предпочтения для всех числовых значений. Некоторые исключения описаны далее.

Числовые значения не должны быть передаваться в форме простой строки символов (тип данных ST);

2) для титра (например, 1:64) и очень немногих других отношений используется тип данных RTO. У титров числителем и знаменателем отношения являются целые числа (например, 1:128). В других отношениях могут использоваться разные количественные типы данных, например, «цена», определенная как физическая величина с типом данных MO.

Иногда по местным соглашениям для титров передаются только знаменатели (например, 32 вместо 1/32). Такие соглашения могут вносить путаницу и в сообщениях стандарта HL7 вместо них должны использоваться правильные отношения;

3) Для передачи значений индексов (чисел без единицы измерения) используется вещественный тип данных (REAL). Если величина не имеет подходящей единицы измерения, то ее можно передать как вещественное число. В качестве альтернативы можно использовать тип данных PQ с безразмерной единицей (например, 1 или %). Целое число можно передавать только в том случае, если согласно определению исследования результатами могут являться только целые числа, что является редким случаем, например, если результат исследования имеет перечислимые значения (см. ниже);

4) диапазоны (например, <3; 12—20) должны быть представлены в форме диапазонов физических количеств (тип данных IVL<PQ>) или в форме интервалов значений других количественных типов данных.

Иногда такие интервалы используются, чтобы указать неопределенность измеренного значения. Однако на этот случай имеются специальные расширения типов данных;

5) для передачи перечислимых значений (например, +, ++, +++; или I, IIa, IIb, III, IV) используется тип данных CO (coded ordinal — кодированное перечислимое значение);

6) номинальные результаты («таксоны», например, тип организма) передаются с помощью любого из кодируемых типов данных (CD, CE), которые указывают, по меньшей мере, код, систему кодирования и необязательный исходный текст, преобразование в другую систему кодирования, а иногда — еще и квалификаторы;

7) для представления мультимедийных данных используется тип данных ED (Encapsulated Data — инкапсулированные данные). С его помощью можно передать изображение (например, рентгенограмму грудной клетки) или видеозапись (например, результат коронарной ангиографии или эхокардиограмму) в виде вложенных двоичных данных или ссылки на внешние адреса, откуда они могут быть загружены по запросу;

8) записи биосигналов можно передавать, используя шаблоны коррелируемых последовательностей результатов (Correlated Observation Sequences), обеспечивающих передачу всей необходимой информации в структуре,

предложенной в стандарте HL7. Эту информацию можно также передать, вложив ее в значение типа ED. Это позволяет использовать другие форматы цифрового кодирования биосигналов, кроме предложенных в стандарте HL7, а также передавать ссылки на внешние адреса, откуда оцифрованные записи биосигналов могут быть загружены по требованию;

9) иногда для передачи формализованных выражений, для которых не подходит ни один из указанных выше типов данных, может использовать строковый тип данных. Однако строковый тип данных не должен использоваться, если значение может быть представлено одним из существующих типов данных;

10) временны́е метки не должны передаваться как значения результатов исследования, если для них можно найти более подходящие места, например, атрибут Act.effectiveTime некоторого действия (к примеру, «время получения биоматериала лабораторией» можно передать в атрибуте effectiveTime действия, описывающего транспорт биоматериала в лабораторию, а не как результат исследования);

11) множества значений любого типа данных, перечисляемые данные, а также интервалы часто используются в экземплярах класса Observation, описывающих критерии (т. е. в экземплярах, у которых атрибут moodCode имеет значение «EVN.CRT»), чтобы указать «референтные пределы» или «пороговые значения» (для тревожных сигналов) и т. д.;

12) для последовательностей результатов (повторяемые измерения того же свойства в течение относительно короткого времени) используется тип данных LIST (list — список)). Дополнительные детали см. в спецификации коррелируемых последовательностей результатов (Correlated Observation Sequences);

13) степень неопределенности значений можно указать, используя расширения типа данных вероятности и распределения вероятности (UVP, PPD). Для передачи статистики абсолютных частот категорий вполне пригоден тип мультимножества BAG.

A.3.1.12.2 Observation.interpretationCode:: SET<CE> (0..*)

Словарный домен: ObservationInterpretation

Определение. Один или несколько кодов, указывающих грубую качественную интерпретацию исследования, например, «норма», «патология», «ниже нормы», «возрастает», «устойчивый», «чувствительный» и т. д.

Обсуждение. Эти коды интерпретации иногда называют «флагами аномалий», однако оценка нормы представляет собой только один пример грубой интерпретации и часто не применима. Например, оценка чувствительности микроорганизмов не имеет никакого отношения к «норме», и при любом исследовании патологического состояния нет смысла говорить о норме, так как патологические состояния никогда не считаются «нормальными».

A.3.1.12.3 Observation.methodCode:: SET<CE> (0..*)

Словарный домен: ObservationMethod

Определение. Код, указывающий дополнительную информацию о методике или способе исследования.

Примеры — Способы измерения давления крови (артериальная пункция, сфигмоманометр (Riva-Rocci), сидя, лежа на спине и т. д.).

Ограничения. При всех исследованиях метод их выполнения частично определяется по значению атрибута Act.code. В этом случае атрибут methodCode не должен был бы использоваться вообще. Однако в тех случаях, когда метод исследования надо указать с большей точностью, нежели это позволяет значение атрибута Act.code, необходимую дополнительную информацию можно передать в атрибуте methodCode. При этом обработка информации об исследовании, осуществляемая информационной системой или процессом, не должна зависеть от детальных данных, переданных в атрибуте methodCode в дополнение к информации о методе исследования, вытекающей из значения атрибута Act.code.

Если атрибут methodCode используется для указания деталей метода, который идентифицируется по значению атрибута Act.code, то значение атрибута methodCode не должно противоречить этому методу.

Обсуждение. При всех исследованиях метод их выполнения частично определяется по виду исследования (определению исследования, атрибут Act.code) и эту косвенную информацию о методе не надо явным образом указывать в атрибуте Observation.methodCode. Например, в классификации LOINC многим видам исследований присвоены разные коды, если методики исследования различаются и это может практически повлиять на интерпретацию результатов исследования. Например, в этой классификации проводится различие между исследованием чувствительности к антибиотикам методом «минимальной ингибирующей концентрации» (МИК) и «методом диффузии в агаре» (Кирби-Бауэр), так что этим видам исследований специально присвоены разные коды. Следовательно, значение атрибута methodCode может лишь служить дополнительным квалификатором, указывающим то, что не выводится непосредственно из значения атрибута Act.code.

Кроме того, некоторые вариации методов могут быть связаны с конкретным используемым устройством. Но значение атрибута methodCode не должно использоваться для указания конкретного устройства или использованного диагностикума. Такую информацию надо связывать с данным экземпляром класса Observation, используя экземпляр класса Participation, у которого атрибут typeCode имеет значение «DEV» (device — устройство).

A.3.1.12.4 Observation.targetSiteCode:: SET<CD> (0..*)

Словарный домен: ActSite

Определение. Код, указывающий анатомическую локализацию или систему организма, являющуюся предметом исследования, если такая информация не вытекает из определения исследования или из значения атрибута Act.code.

Ограничения. Если значение атрибута targetSiteCode присутствует, то оно не должно противоречить той информации об анатомической локализации или системе организма, которая вытекает из определения исследования или из значения атрибута Act.code.

Обсуждение. В большинстве случаев исследуемая анатомическая локализация вытекает из определения исследования, значения атрибута Act.code или значения атрибута Observation.value. Например, «шумы сердца» заведомо относятся к сердцу. Атрибут targetSiteCode используется только в тех случаях, когда анатомическая локализация должна быть детализирована, например, чтобы различать правую и левую сторону, и т. д.

Если предметом исследования не является человек или животное, то этот атрибут используется аналогичным образом для указания структурного ориентира предмета исследования. Например, если предмет — озеро, то исследуемой локализацией может быть приток или исток, и т. д. Если предметом является лимфоузел, то в качестве локализации можно указать «ворота», «периферию» и т. д.

A.3.1.13 Класс Participation (в предметной области Acts)

Свойства класса Participation

Атрибуты класса Participation:

- typeCode :: CS,
- functionCode :: CD,
- contextControlCode :: CS,
- sequenceNumber :: INT,
- negationInd :: BL,
- noteText :: ED,
- time :: IVL<TS>,
- modeCode :: CE,
- awarenessCode :: CE,
- signatureCode :: CE,
- signatureText :: ED,
- performInd :: BL,
- substitutionConditionCode :: CE.

Ассоциации класса Participation:

- act::(1..1)Act::participation::(0..*) (ассоциация с классом Act, роль participation — участие),
- role::(1..1)Role::participation::(0..*) (ассоциация с классом Role, роль participation — участие).

Класс Participation является обобщением следующего класса:

- ManagedParticipation (управляемое участие).

Определение класса Participation

Класс Participation моделирует ассоциацию между классом действия Act и классом сущности Entity, принимающим участие в действии, описанном классом Act, и выполняющим в этом участии роль, описанную классом Role. Каждая пара сущность-роль связана с экземпляром класса Act с помощью одного экземпляра класса Participation. Тип вовлечения этой пары в действие, описываемое экземпляром класса Act, задается с помощью атрибута Participation.typeCode.

Обоснование. Экземпляры класса Participation описывают выполняемую функцию, в то время как экземпляры класса Role описывают компетенцию. Экземпляры класса Participation описывают фактическое участие сущности в определенном действии и, следовательно, вид участия зависит от специфики этого действия. Напротив, роль описывает компетенцию сущности (например, какое принципиальное участие она может принимать в каких типах действий) независимо от конкретного действия.

Например, профессиональные полномочия лица (описанные экземпляром класса Role) могут значительно отличаться от его фактических действий (описанных экземпляром класса Participation). Типичным примером может служить выполнение интерном или ординатором анестезии или хирургической операции под присмотром (более или менее тщательным) ведущего хирурга.

Один и тот же вид участия в действии могут принимать несколько сущностей, что бывает при коллективном сотрудничестве или групповых действиях. Понятие кратных участий частично перекрывается с понятием подчиненных действий (компонентов действий). Когда в действие вовлечено несколько действующих лиц, каждое из них выполняет свою задачу (за крайне редким исключением таких симметричных действий, примером которых служат два человека, которые тянут одну веревку, каждый со своего конца). Поэтому участие нескольких действующих лиц вполне удовлетворительно может быть представлено как составное действие, образованное из нескольких действий, каждое из которых выполняется ровно одним лицом.

Например, в протокол хирургической операции могли быть внесены три лица:

а) лицо, давшее согласие на проведение операции;

б) ведущий хирург;

в) анестезиолог.

Эти лица выполняют разные задачи, которые можно представить в виде трех связанных действий:

а) предоставление согласия,

б) собственно операция,

в) анестезия, выполняемая параллельно с операцией.

Если используются три таких компонента, то тип действующего лица у согласившегося на операцию, хирурга и анестезиолога будет одним и тем же, а именно, «исполнитель». Чем более детальные компоненты действий используются, тем меньше требуется градаций действующих лиц. И наоборот, чем меньше компонентов у действия, тем больше требуется градаций действующих лиц.

Как эмпирическое правило, использование подзадач вместо нескольких действующих лиц полезно в тех случаях, когда каждая подзадача требует специального планирования, или отдельной оплаты, или если полная ответственность за выполнение подзадач различна. Однако в большинстве случаев ресурсы персонала планируются в форме бригад (а не отдельных сотрудников), методы расчетов по оплате лечения имеют тенденцию группировки мелких подзадач в одну строку счета, а полная ответственность часто возлагается на одного лечащего враче, старшую медсестру или заведующего отделением. Такая модель позволяет сочетать подходы моделирования действий как выполняемых несколькими лицами и как состоящих из нескольких задач, что отражает реалии процесса оказания медицинской помощи. При этом наблюдается небольшой уклон в пользу объединения мелких подзадач в одно действие.

Примеры

1 Исполнители действий (хирурги, исследователи, врачи общей практики).

2 Субъекты действий, пациент, устройства.

3 Местонахождения.

4 Автор, поручитель, свидетель, информатор.

5 Адресат, получатель информации.

Атрибуты класса Participation

A.3.1.13.1 Participation.typeCode:: CS (1..1) Mandatory

Словарный домен: ParticipationType

Определение. Код, указывающий вид участия (вовлечения) сущности, выполняющей определенную роль, связанную с этим участием, в действии, описанном в ассоциированном экземпляре класса Act.

Ограничения. Допустимые значения атрибута Participant.typeCode имеют строгие семантические определения, соответствующие области применения стандарта HL7. Это атрибут с кодируемыми значениями без исключений. Альтернативные системы кодирования этого атрибута не допускаются.

A.3.1.13.2 Participation.functionCode:: CD (0..1)

Словарный домен: ParticipationFunction

Определение. Необязательный код, указывающий дополнительную информацию о функции данного участия в действии, если она однозначно не вытекает из значения атрибута Participation.typeCode.

Примеры — Ведущий хирург, второй хирург (или первый ассистент хирурга, стоящий напротив ведущего хирурга), второй ассистент (часто стоящий рядом с ведущим хирургом), потенциально третий ассистент, старшая операционная сестра, операционная сестра, медрегистратор, ведущий анестезиолог, анестезиолог, сестра-анестезистка, санитар, сестра послеоперационного мониторинга, ассистенты, акушерки, студенты и т. д.

Ограничения. Если этот код указан, его значение не должно конфликтовать со значением атрибута Participation.typeCode. В автоматизированных системах не должна присуствовать функциональная зависимость от этого кода.

Не будет написано ни одной спецификации стандарта HL7, технически зависящей от атрибута functionCode. Если возникнет потребность в дополнительных понятиях, то они должны быть определены для значений атрибута Participation.typeCode.

Обсуждение. С помощью этого кода можно указать большое разнообразие и нюансы функций участников в действии. Число и виды этих функций зависят от конкретного типа действий. Например, для каждого вида операции и метода ее проведения может требоваться разное число ассистентов хирурга или медицинских сестер.

Поскольку функции участия характеризуют то, что именно люди выполняют в процессе действия, то в действительности они эквивалентны подзадачам, которые могут выполняться параллельно. Если нужны дополнительные сведения об этих подзадачах, кроме списка их участников, то надо использовать компоненты действия.

A.3.1.13.3 Participation.contextControlCode:: CS (0..1)

Словарный домен: ContextControl

Определение. Код, указывающий, какой вклад вносит данный экземпляр класса Participation в контекст текущего экземпляра класса Act, и будет ли этот контекст распространяться на действия-потомки, чьи связи разрешают такое распространение (см. описание атрибута ActRelationship.contextConductionInd).

Обсуждение. Обоснование, обсуждение и примеры см. в описании атрибута ActRelationship.contextControlCode.

A.3.1.13.4 Participation.sequenceNumber:: INT (0..1)

Определение. Целое значение, указывающее относительный порядок данного участия среди других участий в том же самом действии.

Обоснование. Упорядочение экземпляров класса Participation, описывающих участие в данном действии, может быть предпринято по функциональным причинам или для присвоения участиям приоритета. Примером может

служить указание порядка участия выгодоприобретателей, необходимое для координации возмещений по счету на оплату страхового случая.

A.3.1.13.5 Participation.negationInd:: BL (0..1)

Определение. Значение TRUE этого атрибута указывает, что участие, описанное в экземпляре класса Participation, не имело, не имеет или не должно иметь место (в зависимости от значения атрибута moodCode).

Обоснование. Этот атрибут имеет два основных приложения:

1) чтобы указать, что конкретная роль не имела или не должна иметь участие в действии;

2) чтобы удалить участника из контекста, распространяемого на другие действия.

Обсуждение. Экземпляр класса Participation, у которого атрибут negationInd имеет значение TRUE, обладает преимуществом по отношению к экземпляру класса Participation, у которого атрибут negationInd имеет значение FALSE. Другими словами, если возник конфликт, то отрицание участия имеет больший приоритет.

Примеры — Др. Смит не участвовал; пациент Джонс не подписывал согласие.

A.3.1.13.6 Participation.noteText:: ED (0..1)

Определение. Текстовое или мультимедийное представление комментария к данному участию. Относится только к этому участнику.

A.3.1.13.7 Participation.time:: IVL<TS> (0..1)

Определение. Интервал времени, в течение которого сущность, связанная с действием с помощью конкретного экземпляра класса Participation, была вовлечена в это действие.

Обоснование. Указание времени участия необходимо, если сущность принимала участие в действии не на протяжении всего времени выполнения действия. Время участия также используется для указания времени, в течение которого выполнялись определенные очень общие подзадачи, не заслуживающие упоминания в полном действии.

Примеры

1 Время ввода данных в систему-источник передается в атрибуте Participation.time экземпляра класса Participation, связывающем действие с ролью «ввод данных».

2 Концом времени участия автора считается время подписи данных, передаваемых в экземпляре класса Act.

3 Временем участия поручителя, передаваемым в атрибуте Participation.time, считается момент его подписи.

A.3.1.13.8 Participation.modeCode:: CE (0..1)

Словарный домен: ParticipationMode

Определение. Код, указывающий модальность участия сущности в действии, выполняя определенную роль.

Обоснование. Этот атрибут особенно часть используется для участников «автор» («создатель»), чтобы определить, обеспечивалась ли информация, представленная действием, первоначально устно, письменно (рукописно) или в электронном виде.

Примеры — Физическое присутствие, по телефону, письменная коммуникация.

A.3.1.13.9 Participation.awarenessCode:: CE (0..1)

Словарный домен: TargetAwareness

Определение. Код, указывающий степень осведомленности сущности о действии, в которой она участвует в определенной роли (обычно в качестве целевого субъекта).

Примеры

1 В записях медицинской карты может быть указана степень осведомленности пациента, члена его семьи или другого участника о наличии у пациента смертельного заболевания.

2 Если степень осведомленности, отказ, отсутствие сознания и т. д. — предмет медицинского рассмотрения (например, часть списка проблем), то нужно передавать эту информацию в экземплярах класса Observation, а не полагаться исключительно на этот простой атрибут класса Participation.

A.3.1.13.10 Participation.signatureCode:: CE (0..1)

Словарный домен: ParticipationSignature

Определение. Код, указывающий, заверила ли сущность свое участие подписью, а также указывающий необходимость такой подписи.

Примеры — Протокол хирургической операции (представленный в форме экземпляра класса Procedure) должен быть подписан ответственным оперировавшим хирургом и, возможно, другими участниками операции. (См. также описание атрибута Participation.signatureText.)

A.3.1.13.11 Participation.signatureText:: ED (0..1)

Определение. Текстовое или мультимедийное представление подписи, подтверждающей определенный вид участия сущности в действии (указанный в атрибуте Participation.typeCode) и ее согласие принять на себя связанную с этим ответственность.

Примеры

1 Участник «автор» берет на себя ответственность за достоверность содержания экземпляра класса Act в меру его осведомленности.

2 Получатель информации подтверждает лишь факт ее получения.

Обсуждение. Подпись может быть представлена многими разными способами либо по значению, либо по ссылке в соответствии с типом данных ED. Типичные случаи таковы:

1) подписи бумажных документов: значение, имеющее тип данных ED, может содержать ссылку на некоторый документ или файл, который может быть найден с помощью электронного интерфейса к архиву бумажных документов;

2) электронная подпись: с помощью типа данных ED можно представить фактически любую схему электронной подписи;

3) стандартная электронная подпись: в частности, с помощью типа данных ED можно представить стандартные электронные подписи, например, с помощью ссылки на блок данных подписи, сконструированный в соответствии со стандартом электронной подписи, например, XML-DSIG, PKCS#7, PGP и т. д.

A.3.1.13.12 Participation.performInd:: BL (0..1)

Определение. Указывает, должен ли быть зарезервирован ресурс, необходимый для этого участия, до начала участия (т. е. предоставление ресурса управляется расписанием).

Обоснование. Этот признак используется для очень специфичных нужд в контексте планирования ресурсов. В большинстве описаний участия он не требуется. Чаще всего он применяется при описании участия, для которого требуется конкретное помещение или определенное оборудование, использование которого контролируется расписанием.

A.3.1.13.13 Participation.substitutionConditionCode:: CE (0..1)

Словарный домен: SubstitutionCondition

Определение. Указывает условия, при которых один участвующий предмет может быть заменен другим.

A.3.1.14 Класс PatientEncounter (в предметной области Acts)

Свойства класса PatientEncounter

Атрибуты класса PatientEncounter:

- admissionReferralSourceCode:: CE,
- lengthOfStayQuantity:: PQ,
- dischargeDispositionCode:: CE,
- preAdmitTestInd:: BL,
- specialCourtesiesCode:: SET<CE>,
- specialArrangementCode:: SET<CE>.

Класс PatientEncounter является специализацией класса Act.

Определение класса PatientEncounter

Взаимодействие между пациентом и поставщиком (поставщиками) медицинской помощи с целью предоставления одной или нескольких услуг в сфере здравоохранения. К таким услугам относится в том числе оценка состояния здоровья.

Примеры — Амбулаторное посещение нескольких отделений, медицинская помощь на дому (включая физиотерапию), госпитализация, посещение травмпункта, полевая медицинская помощь (например, при дорожном происшествии), посещение кабинета врачебного приема, производственная терапия, телефонный звонок.

Атрибуты класса PatientEncounter

A.3.1.14.1 PatientEncounter.admissionReferralSourceCode:: CE (0..1)

Словарный домен: EncounterReferralSource

Определение. Источник направления, инициировавшего данное посещение.

A.3.1.14.2 PatientEncounter.lengthOfStayQuantity:: PQ (0..1)

Определение. Общее время, которое субъект проведет или провел в медицинской организации как часть посещения.

Обсуждение. Фактическое число дней пребывания не может быть получено вычитанием даты выписки из даты поступления из-за возможных отпусков (на выходные).

A.3.1.14.3 PatientEncounter.dischargeDispositionCode:: CE (0..1)

Словарный домен: EncounterDischargeDisposition

Определение. Код, указывающий место отправки пациента при выписке (например, выписан домой, умер, выписался вопреки рекомендациям врача, и т. д.). Если посещение все еще «активно» (то есть у него еще нет даты завершения), то этот атрибут должен рассматриваться как ожидаемое место отправки. Если посещение уже «завершено», то в этом атрибуте передается фактическое место отправки.

A.3.1.14.4 PatientEncounter.preAdmitTestInd:: BL (0..1)

Определение. Признак необходимости предварительного выполнения лабораторных анализов перед данным посещением пациента.

A.3.1.14.5 PatientEncounter.specialCourtesiesCode:: SET<CE> (0..*)

Словарный домен: EncounterSpecialCourtesy

Определение. Код, указывающий признак особого обслуживания пациента. Например, обычное обслуживание, особое обслуживание, особое профессиональное обслуживание, обслуживание очень важного лица.

A.3.1.14.6 PatientEncounter.specialArrangementCode:: SET<CE> (0..*)

Словарный домен: SpecialArrangement

Определение. Код, указывающий тип специальных мер, которые должны быть предприняты до посещения пациента (например, каталка, носилки, переводчик, дежурный врач, собака-поводырь). Если атрибут moodCode экземпляра класса PatientEncounter указывает будущее действие, то этот экземпляр может использоваться для указания специальных мер, которые надо предпринять перед ожидаемым поступлением пациента. Это не связано с событием размещения пациента.

A.3.1.15 Класс Procedure (в предметной области Acts)

Свойства класса Procedure

Атрибуты класса Procedure:

- methodCode:: SET<CE>,
- approachSiteCode:: SET<CD>,
- targetSiteCode:: SET<CD>.

Класс Procedure является специализацией класса Act.

Определение класса Procedure

Непосредственным и основным результатом процедуры (постусловие) является измененное физическое состояние субъекта.

Примеры — Изменения физического состояния могут включать в себя нарушение целостности некоторой поверхности тела (например, разрез при хирургической операции), выполнение консервативных процедур, например, вправление вывихнутого сустава, физиотерапию, манипуляции хиропрактика, массаж, бальнеотерапию, иглоукалывание, шиятсу и т. д. За пределами клинической медицины можно привести такие примеры процедур как изменение окружающей среды (например, спрямление рек, осушение болот, возведение дамб), ремонт или модернизация машин и т. д.

Обсуждение. Применительно к клинической медицине процедура — один из нескольких типов клинических действий, например, исследование, лекарственные назначения и коммуникативные воздействия (к примеру, обучение, консультация, психотерапия, представленные в форме экземпляров класса Act без специальных атрибутов). Понятие процедуры не поглощает эти другие понятия и, в свою очередь, не поглощается ими. Примечательно, что оно не включает в себя все действия, целью которых является вмешательство или лечение. Будет ли физическое изменение полезным субъекту или предполагается полезным, не имеет значения, существенно лишь то, что целью действия является изменение физического состояния субъекта.

Выбор между способами представлениями реальных действий в форме экземпляров классов основан на том, применимы ли специфические свойства процедуры и будет ли физическое изменение необходимым постусловием деятельности или ее части. Например, получение рентгеновского изображения иногда называется «процедурой», но в модели RIM оно не отображается на экземпляр класса Procedure, поскольку целью рентгеновского снимка не является изменение физического состояния тела.

При многих видах клинической деятельности отдельные действия, которые по своей природе являются исследованиями и процедурами, объединяются в одно составное действие. Например, при выполнении процедур лечебной радиологии (к примеру, интраартериальный тромболиз) производятся как исследования, так и лечение, и большинство хирургических процедур включает в себя намеренные и документированные шаги исследований. Такие клинические воздействия лучше всего представлять в виде нескольких действий-компонентов, каждый из которых имеет соответствующий тип.

Атрибуты класса Procedure

A.3.1.15.1 Procedure.methodCode:: SET<CE> (0..*)

Словарный домен: ProcedureMethod

Определение. Указывает средства или методику выполнения процедуры.

Обсуждение. У любой процедуры может быть несколько разных методов. Хотя они обеспечивают, в основном, те же самые результаты, для более тщательной интерпретации протокола процедуры может требоваться указание, какой именно метод был применен (например, открытая или лапароскопическая холецистэктомия). Понятия метода могут быть «пре-координированы» в определении действия. Поскольку существует много возможных методов, существенно зависящих от конкретных видов процедур, конструирование словарного домена, который бы охватывал все эти методы, представляется затруднительным. Однако для конкретного типа процедуры такую систему кодирования сконструировать можно. Таким образом, пользователь, направляющий пациента на процедуру, может указать конкретный метод ее выполнения, выбрав требуемый вариант из такой системы кодирования. Возможные варианты методов выполнения могут быть описаны в справочнике процедур. В записях такого справочника, передаваемых с помощью экземпляров класса Procedure, у которых атрибут moodCode имеет значение «DEF» (definition — определение), атрибут methodCode может содержать список методов выполнения процедуры,

используемый для выбора нужного метода при оформлении направления на процедуру или для проверки полученного протокола процедуры.

A.3.1.15.2 Procedure.approachSiteCode:: SET<CD> (0..*)

Словарный домен: ActSite

Определение. Анатомическая локализация или система организма, через которую процедура достигает места своего применения (см. описание атрибута targetSiteCode).

Примеры

1 При нефрэктомии доступ к почке может быть трансабдоминальным или, главным образом, ретроперитонеальным.

2 Целью катетеризации легочной артерии является легочная артерия, но при этом местом доступа обычно является внутренняя яремная вена или подключичная вена, в шее или, соответственно, в подключичной ямке.

3 Для неинвазивных процедур, например, иглоукалывания, место доступа — прокалываемое место кожи.

Обсуждение. Если субъектом процедуры является не человек и не животное, то этот атрибут используется аналогичным образом, чтобы определить структурный ориентир на предмете действия.

Списки мест доступа могут быть «пре-координированы» с описанием процедуры, чтобы исключить неправильный выбор анатомической локализации. Одна и та же информационная структура может использоваться в обоих подходах: пре-координированном и пост-координированном.

A.3.1.15.3 Procedure.targetSiteCode:: SET<CD> (0..*)

Словарный домен: ActSite

Определение. Анатомическая локализация или система организма, являющаяся местом применения процедуры.

Примеры

1 Местом применения нефрэктомии является правая или левая почка.

2 Местом применения катетеризации легочной артерии является легочная артерия.

3 Для неинвазивных процедур, например, иглоукалывания, местом применения является орган/система, на которые рассчитано воздействие (например, «печень»).

Обсуждение. Если субъектом процедуры является не человек и не животное, то этот атрибут используется аналогичным образом, чтобы определить структурный ориентир на предмете действия.

Списки мест применения могут быть «пре-координированы» с описанием процедуры, чтобы исключить неправильный выбор анатомической локализации. Одна и та же информационная структура может использоваться в обоих подходах: пре-координированном и пост-координированном.

A.3.1.16 Класс PublicHealthCase (в предметной области Acts)

Свойства класса PublicHealthCase

Атрибуты класса PublicHealthCase:

- detectionMethodCode:: CE,
- transmissionModeCode:: CE,
- diseaseImportedCode:: CE.

Класс PublicHealthCase является специализацией класса Observation.

Определение класса PublicHealthCase

Описание события, имеющего определенное значение для общественного здоровья, представляется как специализация класса Observation. Обычно это информация об одном или нескольких случаях инфекционного заболевания или другого события, подлежащего регистрации. Описание события общественного здоровья может включать в себя сведения о состоянии здоровья одного лица или сведения о нескольких случаях того же самого заболевания или условия, которые рассматриваются как угроза общественному здоровью. Типичным примером может служить вспышка заболевания. Определение события общественного здоровья (передаваемое в экземпляре класса PublicHealthCase, у которого атрибут moodCode имеет значение «DEF» (definition — определение)) включает в себя описание клинических, лабораторных и эпидемиологических показателей, связанных с заболеванием или условием, влияющим на общественное здоровье. Есть определения событий для условий, которые подлежат регистрации, а также для тех, которые не подлежат. Есть также определения событий вспышек заболеваний. Определение события общественного здоровья используется здравоохранением для статистики событий и не должно использоваться в качестве клинических рекомендаций, предназначенных для лечения. Примерами служат СПИД, синдром токсического шока, сальмонеллез и связанные с ними признаки, используемые при определении угрозы.

Атрибуты класса PublicHealthCase

A.3.1.16.1 PublicHealthCase.detectionMethodCode:: CE (0..1)

Словарный домен: CaseDetectionMethod

Определение. Код, указывающий метод получения санитарно-эпидемиологической службой информации о событии общественного здоровья. Примерами могут служить сообщение поставщика медицинской помощи, само-

стоятельное обращение пациента, сообщение лаборатории, отчет о результате исследования угрозы или вспышки, выявление контактов, активное наблюдение, рутинный физикальный осмотр, дородовое исследование, перинатальное исследование, мониторинг профессиональных заболеваний, анализ медицинских карт и т. д.

A.3.1.16.2 PublicHealthCase.transmissionModeCode:: CE (0..1)

Словарный домен: CaseTransmissionMode

Определение. Код, указывающий механизм приобретения заболевания живым субъектом, вовлеченным в событие общественного здоровья. Примерами служат половой путь, воздушно-капельный, передача через кровь, трансмиссивный путь, передача через продукты питания, зоонозный путь, внутрибольничная инфекция, механический путь, кожный, врожденный, воздействие внешней среды, неопределенный.

A.3.1.16.3 PublicHealthCase.diseaseImportedCode:: CE (0..1)

Словарный домен: CaseDiseaseImported

Определение. Код, указывающий, была ли болезнь, вероятно, приобретена за пределами территории юрисдикции санитарно-эпидемиологической службы, и какова природа межтерриториальных отношений. Примерами возможных значений служат: не занесенная, занесенная из другой страны, занесенная из другого штата, занесенная с другой территории, недостаточно информации для определения.

A.3.1.17 Класс SubstanceAdministration (в предметной области Acts)

Свойства класса SubstanceAdministration

Атрибуты класса SubstanceAdministration:

- routeCode:: CE,
- approachSiteCode:: SET<CD>,
- doseQuantity:: IVL<PQ>:: CE,
- rateQuantity:: IVL<PQ>,
- doseCheckQuantity:: SET<RTO>,
- maxDoseQuantity:: SET<RTO>,
- administrationUnitCode :: CE.

Класс SubstanceAdministration является специализацией класса Act.

Определение класса SubstanceAdministration

Медикаментозное назначение, т. е. действие введения или иного применения субстанции к субъекту.

Обсуждение. Эффект применения субстанции обычно вызывается биохимическими процессами, однако это не обязательно. Например, лучевую терапию в целом можно описать аналогичным образом, особенно соматическую терапию, к примеру, радиоактивным йодом. К этому же классу действий относится применение химического лечения к определенной области.

Примеры — Протокол химиотерапии, рецепт, запись о вакцинации.

Атрибуты класса SubstanceAdministration

A.3.1.17.1 SubstanceAdministration.routeCode:: CE (0..1)

Словарный домен: RouteOfAdministration

Определение. Метод введения терапевтического материала внутрь или на поверхность тела субъекта.

Обсуждение. Путь введения лекарственного средства, место его введения (атрибут approachSiteCode) и устройство, используемое для введения, тесно связаны. Все три (если указаны) должны быть хорошо согласованы между собой. В некоторых случаях используемая система кодирования должна быть пре-координирована с одной или несколькими другими системами.

Когда лекарственное средство применяется к окружающей среде или к некоторому физическому месту, то код способа введения указывает это место в его «теле».

Примеры — Перорально — per os (PO), сублингвально — sublingual (SL), суппозиторий — rectal (PR), ингаляция — per inhalationem (IH), глазные капли — ophtalmic (OP), назальное применение — nasal (NS), ушное применение — otic (OT), пессарий — vaginal (VG), внутрикожно — intra-dermal (ID), подкожно — subcutaneous (SC), внутривенно — intra-venous (IV), внутрисердечно — intra-cardial (IC)

A.3.1.17.2 SubstanceAdministration.approachSiteCode:: SET<CD> (0..*)

Словарный домен: ActSite

Определение. Детальное указание анатомической локализации, куда лекарство вводится или где оно применяется к субъекту.

Обсуждение. Этот атрибут необходим только в том случае, если значение атрибута routeCode требует дальнейшего уточнения. Например, если атрибут routeCode имеет значение «PO» (перорально), то никакая дальнейшая спецификация места введения не требуется. Если, однако, атрибут routeCode имеет значение «IV» (внутривенно) или «IM» (внутримышечно), то в атрибуте approachSiteCode можно указать точное место введения (например, правое предплечье или левая дельтовидная мышца соответственно).

Способ введения, место его применения (approachSiteCode) и устройство, используемое для введения, тесно связаны. Все три (если указаны) должны быть хорошо согласованы между собой. В некоторых случаях используемая система кодирования должна быть пре-координирована с одной или несколькими другими системами.

A.3.1.17.3 SubstanceAdministration.doseQuantity:: IVL<PQ> (0..1)

Определение. Количество терапевтического агента или другой субстанции, введенное за один прием.

Доза может определяться или как физическая величина активного ингредиента (например, 200 мг), или как количество единиц назначения (например, таблетки, капсулы, «штуки»). Какой выбрать подход, зависит от исполнителя «потребляемого» участия (который идентифицирует вводимое лекарство). Если потребляемое лекарство имеет неисчисляемую форму дозировки (например, измеряется в миллиграммах или в литрах), тогда доза должна выражаться в этих единицах. Если потребляемое лекарство имеет исчисляемую форму дозировки (таблетки, капсулы, «штуки»), тогда доза должна быть выражена как безразмерностная величина (т. е. никакая другая единица измерения не указывается).

Единицы измерения должны быть ограничены такими измеряемыми единицами как литры, миллиграммы. Не измеряемые, но исчисляемые единицы, например, таблетки и капсулы, не должны указываться в компоненте единиц измерения типа данных PQ иначе как в аннотации, указанной в фигурных скобках ({xxx}). См. спецификацию типов данных в документе Data Types Part II Unabridged Specification, Appendix A: Unified Code for Units of Measure.

A.3.1.17.4 SubstanceAdministration.rateQuantity:: IVL<PQ> (0..1)

Определение. Указывает скорость введения субстанции в субъект. Выражается как физическая (экстенсивная) величина, введенная за единицу времени (например, 100 мл/ч, 1 г/сут, 40 ммоль/ч, и т. д.).

Обсуждение. Это понятие применимо к непрерывно делимым формам дозировки (например, жидкости, газы). Если значение указанного атрибута является интервалом, то скорость должна быть в этом интервале. Указанный атрибут определяется как физическая (экстенсивная) величина, введенная за единицу времени, т. е. количество, которое передано в атрибуте doseQuantity, деленное на продолжительность введения.

A.3.1.17.5 SubstanceAdministration.doseCheckQuantity:: SET<RTO> (0..*)

Определение. В этом атрибуте указано ожидаемое количество субстанции, которое должно быть введено за период времени. Используется как критерий проверки значений, переданных в других атрибутах.

Обсуждение. Обычно этот атрибут не используется; он предназначен только для специальных целей. В некоторых странах, в частности, в Японии, есть норма, требующая указывать общую суточную дозу в рецепте и сопутствующей документации. Цель этого требования, очевидно, в том, чтобы обеспечить и облегчить контроль общей назначенной дозы во избежание передозировки (или слишком малой дозировки).

Обоснование. Вместо определения атрибута «общая суточная доза», как это было сделано в стандарте HL7 v2.3, здесь введено более общее понятие doseCheckQuantity, имеющее тип данных «отношение» (RTO).

Ограничения: “invariant(SubstanceAdministration med, RTO max) where med.doseCheckQuantity.contains(max) {max.numerator.compares(med.doseQuantity); max.denominator.compares(1 s);}”. Числитель должен быть в единицах, сопоставимых с единицами измерения значения doseQuantity, а знаменатель должен быть мерой времени.

Примеры

1 При назначении эритромицина в дозировке 250 мг (1 таблетка 3 раза в день) общую суточную дозу можно рассчитать по формуле «doseCheckQuantity = doseQuantity (1) * Ingredient.quantity (250 мг) * effectiveTime (3 / сут) = 750 мг/сут».

2 При внутривенном введении ожидаемое количество субстанции в час можно рассчитать по формуле «doseCheckQuantity = doseQuantity (100 мл) * Ingredient.quantity (5 мг/л) / rateQuantity (1 час) = 0.5 мг/ч», что можно преобразовать в суточную дозу по формуле «doseCheckQuantity = 0.5 мг/ч * 24 ч/сут = 12 мг/сут».

A.3.1.17.6 SubstanceAdministration.maxDoseQuantity:: SET<RTO> (0..*)

Определение. Указывает максимальное общее количество терапевтической субстанции, которое может быть введено субъекту за период времени.

Обсуждение. Этот атрибут особенно полезен, если разрешенная дозировка задается в форме диапазона либо режим введения является переменным или по мере необходимости. С его помощью можно указать предел общего количества субстанции, которое может быть применено за период времени. Чтобы указать разные пределы для разных периодов времени, можно указать несколько значений атрибута maxDoseQuantity.

Примеры — 500 мг/сут; 1200 мг/неделя.

Ограничения: “invariant(SubstanceAdministration med, RTO max) where med.maxDoseQuantity.contains(max) {max.numerator.compares(med.doseQuantity); max.denominator.compares(1 s);}”. Числитель должен выражаться в единицах, сопоставимых с единицами измерения значения doseQuantity, а знаменатель должен быть мерой времени.

A.3.1.17.7 SubstanceAdministration.administrationUnitCode :: CE (0..1)

Словарный домен: AdministrableDrugForm

Определение. Код, указывающий, что принимаемое лекарственное средство или медицинское изделие взято из большого целого. Используется, если потребляемое средство или изделие указано как большое целое, но разовая доза, указанная в атрибуте doseQuantity, относится к конкретной порции, а не ко всему потребляемому средству или изделие.

Обоснование. Если атрибут administrationUnitCode отсутствует, то в приведенном примере значение атрибута doseQuantity = 1 означало бы, что весь флакон ингалятора должен быть опустошен за один акт применения. С помощью атрибута administrationUnitCode, содержащего код «дозы аэрозоля» (или «вдоха»), можно указать, что значение атрибута doseQuantity относится не ко всему количеству лекарственного средства, а только к его части.

Ограничения:

1) этот атрибут должен использоваться только в том случае, если вещество, указанное в качестве «исполнителя» в экземпляре класса Role, связанном с классом Psrticipation, описывающим потребляемое участие, представляет собой не готовую форму, применяемую целиком, а большее целое, упаковку и т. д.;

2) если вещество, представленное экземпляром класса SubstanceAdministration, является готовой формой, применяемой целиком, то атрибут administrationUnitCode должен иметь пустое значение («не применим»);

3) если это вещество представляет собой аморфную субстанцию (жидкость, газ, порошок и т. д.), измеряемую объемом, массой и т. д., то атрибут administrationUnitCode также должен иметь пустое значение («не применим»);

4) если экземпляр класса SubstanceAdministration означает контейнер, содержание которого измеряется, объемом, массой и т. д., то атрибут administrationUnitCode должен иметь значение, соответствующее «отмеренной порции».

Пример — В системе ввода заказов может быть присвоен Код только «Дозированному аэрозольному ингалятору с будесонидом», а доза должна измеряться «числом доз аэрозоля».

A.3.1.18 Класс Supply (в предметной области Acts)

Свойства класса Supply

Атрибуты класса Supply:

- quantity:: PQ,
- expectedUseTime:: IVL<TS>.

Класс Supply является генерализацией класса Diet.

Класс Supply является специализацией класса Act.

Определение класса Supply

Действие, включающее в себя передачу товара от одной сущности к другой.

Обсуждение. Информация о передаваемом товаре связывается с экземпляром класса Supply с помощью экземпляра класса Participation, у которого атрибут typeCode имеет значение «PRD» (product — товар). При этом важна точная идентификация товара (производитель, серийный номер и т. д.). Большая часть детальной информации о товаре должна передаваться в экземпляре класса Material. Если требуется отдельно описать планирование доставки, доставку и оплату товара, то с экземпляром класса Supply можно связать экземпляр класса Transportation. Для описания услуги отпуска лекарственного средства используется экземпляр класса Supply, связанный с экземпляром класса SubstanceAdministration. В этом случае экземпляр класса SubstanceAdministration описывает применение лекарства, а экземпляр класса Supply — отпуск.

Примеры — Заказ простыней, отпуск лекарства, отпуск медицинских расходных материалов со склада.

Атрибуты класса Supply

A.3.1.18.1 Supply.quantity:: PQ (0..1)

Определение. Количество товара, которое было предоставлено или должно быть предоставлено (в зависимости от значения атрибута moodCode).

Обсуждение. Этот атрибут может использоваться как альтернатива атрибуту expectedUseTime или вместе с этим атрибутом. Если оба эти атрибута указаны, то значение, указанное в атрибуте quantity, представляет собой количество товара, которое предполагается израсходовать в течение времени, указанного в атрибуте expectedUseTime.

Единицы измерения должны быть ограничены такими измеряемыми единицами как литры, миллиграммы. Не измеряемые, но исчисляемые единицы, например, таблетки и капсулы, не должны указываться в компоненте единиц измерения типа данных PQ иначе как в аннотации, указанной в фигурных скобках ({xxx}). См. спецификацию типов данных в документе Data Types Part II Unabridged Specification, Appendix A: Unified Code for Units of Measure. Тип «исчисляемой» информации определен информацией сущности «продукта».

A.3.1.18.2 Supply.expectedUseTime:: IVL<TS> (0..1)

Определение. Указывает период времени, в течение которое предполагается использовать предоставляемый товар, или продолжительность времени, в течение которого будет длиться его предоставление.

В некоторых случаях этот атрибут может использоваться вместо атрибута Supply.quantity для косвенного указания предоставляемого количества в форме продолжительности предоставления. Например, применение лекарства в течение 90 дней лечения (вычисление количества основано на дозировке лекарства), количество реактивного топлива на 10 часов полета, и т. д.

Примечание — По возможности количество лучше передавать в атрибуте Supply.quantity, поскольку это точнее. Значение атрибута Supply.expectedUseTime всегда будет оценкой количества, подверженной влиянию внешних факторов.

A.3.1.19 Класс WorkingList (в предметной области Acts)

Свойства класса WorkingList

Атрибуты класса WorkingList:

- ownershipLevelCode:: CE.

Класс WorkingList является специализацией класса Act.

Определение класса WorkingList

Динамический список отдельных экземпляров класса Act, составленный для отражения потребностей отдельного сотрудника, бригады или организации по просмотру группы действий, выполняемых в клинических или административных целях.

Обсуждение. Группируемые действия связываются с экземпляром класса WorkingList с помощью экземпляров класса ActRelationship, у которых атрибут typeCode имеет значение «COMP» (component — компонент).

Примеры — Список жалоб, список целей лечения, список аллергий, листы назначений и т. д.

Атрибуты класса WorkingList

A.3.1.19.1 WorkingList.ownershipLevelCode:: CE (0..1)

Словарный домен: ListOwnershipLevel

Определение. Указывает, какому уровню подчинения предназначен данный рабочий лист — отдельному лицу, бригаде или организации.

A.3.2 Классы предметной области Entities

A.3.2.1 Класс Container (в предметной области Entities)

Свойства класса Container

Атрибуты класса Container:

- capacityQuantity:: PQ,
- heightQuantity:: PQ,
- diameterQuantity:: PQ,
- capTypeCode:: CE,
- separatorTypeCode:: CE,
- barrierDeltaQuantity:: PQ,
- bottomDeltaQuantity:: PQ.

Класс Container является специализацией класса ManufacturedMaterial.

Определение класса Container

Класс Container является специализацией класса ManufacturedMaterial, предназначенной для описания контейнера, в который помещаются другие сущности в целях транспортировки или защиты их содержания от утери или повреждения.

Обоснование. Спецификация этого класса появилась в результате сотрудничества комитета HL7 и организации NCCLS. Многие из определений атрибутов позаимствованы из стандарта NCCLS или ссылаются на него. Для хранения аморфных субстанций (жидкости, газы) необходим контейнер. Однако его содержание всегда можно отличить от самого контейнера и относительно легко отделяется от него, в отличие от содержания (ингредиента) микстуры.

Применение. Экземпляр класса Container связан с информацией о веществе, содержащемся в контейнере, с помощью экземпляра класса Role, у которого атрибут classCode имеет значение «CONT» (content — содержание).

Атрибуты класса Container

A.3.2.1.1 Container.capacityQuantity:: PQ (0..1)

Определение. Функциональная емкость контейнера.

A.3.2.1.2 Container.heightQuantity:: PQ (0..1)

Определение. Высота контейнера.

A.3.2.1.3 Container.diameterQuantity:: PQ (0..1)

Определение. Внешний диаметр контейнера.

A.3.2.1.4 Container.capTypeCode:: CE (0..1)

Словарный домен: ContainerCap

Определение. Тип крышки контейнера, с которой могут осуществляться такие автоматизированные манипуляции, как открытие, прокалывание и т. д.

A.3.2.1.5 Container.separatorTypeCode:: CE (0..1)

Словарный домен: ContainerSeparator

Определение. Материал, добавляемый в контейнер для обеспечения и облегчения физического разделения компонентов образца различной плотности.

Обоснование. Состав или тип разделяющего вещества может оказывать воздействие на выполняемый анализ. Знание разделяющего материала помогает интерпретировать результаты.

Пример — В пробирки для сбора крови добавляется гель, который при последующем центрифугировании создает физический барьер между клетками крови и сывороткой или плазмой.

A.3.2.1.6 Container.barrierDeltaQuantity:: PQ (0..1)

Определение. Расстояние от контрольной точки до разделяющего материала (барьера) внутри контейнера.

Обоснование. Это расстояние может передаваться автоматизированной лабораторной системой измерительному инструменту и/или устройству обработки биоматериала, чтобы пробоотборник, вставляемый в образец, не касался разделителя. См. определение контрольной точки (Point of Reference) в стандарте NCCLS AUTO5 Laboratory Automation: Electromechanical Interfaces.

A.3.2.1.7 Container.bottomDeltaQuantity:: PQ (0..1)

Определение. Расстояние от контрольной точки до внешнего дна контейнера.

Обоснование. См. определение контрольной точки (Point of Reference) в стандарте NCCLS AUTO5 Laboratory Automation: Electromechanical Interfaces.

A.3.2.2 Класс Device (в предметной области Entities)

Свойства класса Device

Атрибуты класса Device:

- manufacturerModelName:: SC,
- softwareName:: SC,
- localRemoteControlStateCode:: CE,
- alertLevelCode:: CE,
- lastCalibrationTime:: TS.

Класс Device является специализацией класса ManufacturedMaterial.

Определение класса Device

Класс Device является специализацией класса ManufacturedMaterial, предназначенной для описания сущности, которая используется в деятельности, не претерпевая существенных изменений. Тип устройства указывается в атрибуте code, унаследованном от класса Entity.

Использование. Понятие устройства включает в себя как долговечное медицинское изделие (многократно используемое), так и одноразовое медицинское изделие.

Атрибуты класса Device

A.3.2.2.1 Device.manufacturerModelName:: SC (0..1)

Словарный домен: ManufacturerModelName

Определение. Человеко-читаемое имя модели устройства, присвоенное производителем.

Пример — Масс-спектрометр на основе индуктивно связанной плазмы Perkin Elmer 400.

A.3.2.2.2 Device.softwareName:: SC (0..1)

Словарный домен: SoftwareName

Определение. Наименование, версия и выпуск программного обеспечения, управляющего устройством, присвоенные производителем или разработчиком программного обеспечения.

Пример — Agilent Technologies Chemstation A.08.xx.

A.3.2.2.3 Device.localRemoteControlStateCode:: CE (0..1)

Словарный домен: LocalRemoteControlState

Определение. Значение, указывающее текущее состояние управления устройством.

Обоснование. Состояние управления устройством может быть передано другому устройству до дистанционной передачи команд управления. Если состояние управления устройством не имеет значение «R» (remote — дистанционный), то внешние команды будут игнорироваться.

Пример — Устройство может работать автономно (атрибут localRemoteControlStateCode имеет значение «L» (local — местный), или управляться другой системой (атрибут localRemoteControlStateCode имеет значение «R» (remote — дистанционный)).

A.3.2.2.4 Device.alertLevelCode:: CE (0..1)

Словарный домен: DeviceAlertLevel

Определение. Значение, указывающее текущее состояние функционирования автоматизированного устройства.

Примеры — Нормальное, предупреждение, критическое

Ограничения. Допустимые значения атрибута зависят от устройства.

A.3.2.2.5 Device.lastCalibrationTime:: TS (0..1)

Определение. Дата/время последней калибровки устройства.

Обоснование. Чтобы гарантировать, что устройства работают в соответствии со спецификациями, их надо с определенной периодичностью калибровать. Допустимый интервал между калибровками зависит от регламентов. Следовательно, для обеспечения правильности результатов точное время/дата последней калибровки надо рассматривать как критичный параметр.

A.3.2.3 Класс Entity (в предметной области Entities)

Свойства класса Entity

Атрибуты класса Entity:
- classCode:: CS,
- determinerCode:: CS,
- id:: SET<II>,
- code:: CD,
- quantity:: SET<PQ>,
- name:: BAG<EN>,
- desc:: ED,
- statusCode:: CS,
- existenceTime:: IVL<TS>,
- telecom:: BAG<TEL>,
- riskCode:: SET<CE>,
- handlingCode :: CE.

Ассоциации класса Entity:
- communicationFunction::(0..*) CommunicationFunction::entity::(1..*) (ассоциация с классом CommunicationFunction, роль entity — сущность);
- languageCommunication::(0..*) LanguageCommunication::entity::(1..1) (ассоциация с классом LanguageCommunication, роль entity — сущность);
- playedRole::(0..*) Role::player::(0..1) (ассоциация с классом LanguageCommunication, роль player — исполнитель);
- scopedRole::(0..*) Role::scoper::(0..1) (ассоциация с классом LanguageCommunication, роль scoper — контролер).

Класс Entity является генерализацией следующих классов:
- LivingSubject,
- Material,
- Organization,
- Place,
- EntityHeir.

Описание переходов состояния класса Entity приведено в подразделе A.3.2.3.13.

Определение класса Entity

Сущность (физический предмет, группа физических предметов или организация), способная участвовать в действиях, выполняя некоторую роль.

Обсуждение. Сущность — физический объект, который существует, существовал или будет существовать. Единственное исключение — организация, которая, не имея физического облика, обладает другими характеристиками класса Entity. Иерархия специализаций класса Entity включает в себя представления живых существ (включая людей), организации, материал, места и их специализации. Характеристики класса Entity не содержат указаний на выполняемые роли или действия, в которых эти сущности могут участвовать.

Ограничения. Понятие сущности не включает в себя события/дела/действия или выполняемые роли (например, пациент, поставщик).

Атрибуты класса Entity

A.3.2.3.1 Entity.classCode:: CS (1..1) Mandatory

Словарный домен: EntityClass

Определение. Код, определяемый стандартом HL7 и указывающий вид или категорию экземпляра класса Entity.

Обоснование. Вследствие чрезвычайно большого числа потенциальных значений, которые должна охватывать система кодирования, представляющая все физические объекты в мире, атрибут classCode указывает как подтип ветви в иерархии специализаций класса Entity, так и высокоуровневый классификатор экземпляра класса Entity. Его можно использовать для ограничения допустимых областей значений атрибута Entity.code.

Примеры — Человек, животное, химическая субстанция, группа, организация.

A.3.2.3.2 Entity.determinerCode:: CS (1..1) Mandatory

Словарный домен: EntityDeterminer

Определение. Код, определяемый стандартом HL7 и указывающий, представляет ли экземпляр класса Entity вид сущностей (kind-of) или конкретный экземпляр сущности.

Обоснование. Экземпляр класса Entity может представлять информацию о конкретном экземпляре сущности (наиболее общий случай), о перечисляемой группе сущностей, обладающих общими характеристиками, или об общем виде сущностей. Эти варианты представлений различаются с помощью значения атрибута determinerCode.

Примеры — 1 человек (экземпляр), 3 шприца (вид сущностей, количество которых определено) или население Индианаполиса (вид группы)

A.3.2.3.3 Entity.id:: SET<II> (0..*)

Определение. Уникальный идентификатор экземпляра класса Entity.

Обоснование. Для успешной передачи данных требуется лишь наличие у сущности уникального идентификатора. Но поскольку различные системы ведут различные базы данных, то одной и той же сущности в них могут быть присвоены разные идентификаторы. Учтите, что идентификатор экземпляра сущности представляет собой чистый идентификатор, а не классификатор. Для представления серийных номеров, присваиваемых производителями, или артикулов в каталогах поставщиков, или инвентарных номеров материалов и товаров, информация о которых передается в экземплярах класса Material, могут использоваться атрибуты Role.id, что позволяет более ясно отразить тот факт, что такие номера или артикулы присвоены определенной стороной, связанной с этим материалом или товаром.

A.3.2.3.4 Entity.code:: CD (0..1)

Словарный домен: EntityCode

Определение. Значение, указывающее определенный вид сущности, представляемой в форме экземпляра класса Entity.

Обоснование. Система кодирования значений этого атрибута зависит от значения атрибута Entity.classCode, например, своя система кодирования живых субъектов (таксономии животного и растительного мира), своя для химических субстанций (например, код IUPAC), системы кодирования организаций, страховых компаний, органов государственного управления, больниц, парков, озер, шприцев и т. д. В принципе система кодирования значений Entity.code может быть до такой степени детализирована, что будет содержать единственное значение. Примером может служить код CDC производителя вакцины, моделируемый как словарь понятий, в котором каждое понятие фактически относится к единственному экземпляру.

Примеры — Здание больницы, доберман-пинчер, пробирка для сбора крови, биоптат.

A.3.2.3.5 Entity.quantity:: SET<PQ> (0..*)

Определение. Число сущностей или количество сущности, с соответствующими единицами измерения, что определяется значением атрибута Entity.determinerCode.

Примеры

1 Для неопределенных видов значение этого атрибута представляет собой справочное значение для спецификации пропорций ингредиентов или компонентов (например, используя ассоциации RoleLink с экземплярами класса Role, у которых атрибуту RoleLink.typeCode присвоено значение «PART», означающее «имеет часть», «имеет компонент», или «имеет содержание»). К примеру, популяция, в составе которой 60% женщин, может быть описана в виде связи «имеет часть» экземпляра класса Person, у которого атрибут quantity имеет значение 100, с экземпляром класса Person, у которого атрибут quantity имеет значение 60, а атрибут sex — значение «F», т.е. женский пол. Амоксициллин в таблетках с дозировкой 500 мг может быть описан в виде связи «имеет ингредиент» экземпляра класса Material, у которого атрибут formCode имеет значение «TAB» (tablet — таблетка), а атрибут quantity имеет значение 1, с экземпляром класса Material, у которого атрибут code имеет значение Кода амоксициллина, а атрибут quantity имеет значение 500 мг. Раствор глюкозы 5 % (D5W) можно описать в виде связи «имеет ингредиент» экземпляра класса Material, у которого атрибут code имеет значение «D5W», а атрибут quantity — значение 1 кг, с экземпляром класса Material, у которого атрибут code имеет значение Кода глюкозы, а атрибут quantity имеет значение 50 г.

2 Количественные отношения, специфичные для материалов, выражаются, используя тот факт, что тип данных этого атрибута — множество физических величин (SET<PQ>). Если это множество имеет более одного значения, то каждый его элемент рассматривается как определяющий одно и то же количество вещества. Например, для одного литра воды можно было указать множество физических величин 1 л, 1 кг, 55,56 моль, задающих, соответственно, объем, массу и количество вещества для одного и того же самого количества воды, что эквивалентно плотности массы (объемной массе) 1 кг/л и молярной массе (18 г/моль). Для глюкозы можно указать множество значений 180 г, 1 моль в соответствии с ее молярной массой (180 г/моль).

Обсуждение. Если экземпляр класса Entity описывает часть субстанции, то атрибут quantity указывает количество этой части субстанции. Для неопределенной субстанции (вид) это количество удовлетворяет одновременно двум целям: а) предоставляет информацию об отношениях между определенными количествами этой субстанции, и б) является справочным количеством для спецификации ингредиентов или компонентов. Во всех случаях в атрибуте quantity передается экстенсивный вид «количества» (например, число, длина, объем, масса, площадь поверхности, энергия и т. д.). Учтите, что большинство относительных величин и фракций не представляют количества, в частности, фракция массы, концентрация, массовые отношения, проценты, и т. д. не представляют экстенсивные количества и являются запрещенными значениями для этого атрибута.

Ограничения. Экземпляры класса Entity, у которых атрибут determinerCode имеет значение «INSTANCE», должны иметь атрибут quantity со значением 1. Экземпляры класса Entity, у которых атрибут determinerCode имеет значение «QUANTIFIED_KIND», должны иметь атрибут quantity со значением, равным числу членов группы; значение атрибута quantity экземпляров класса Entity, у которых атрибут determinerCode имеет значение «KIND», не определено.

A.3.2.3.6 Entity.name:: BAG<EN> (0..*)

Определение. Неуникальный текстовый идентификатор или псевдоним экземпляра класса Entity.

Обоснование. Большинство сущностей имеет общеупотребительное название, которое может использоваться, чтобы отличить их от других сущностей, но не является уникальным идентификатором.

Примеры — Собственные имена, псевдонимы, юридические фамилии, имена и отчества людей, названия мест или предметов.

A.3.2.3.7 Entity.desc:: ED (0..1)

Определение. Текстовое или мультимедийное описание сущности.

Обсуждение. Содержание описания не рассматривается в качестве части функциональной информации, передаваемой другой системе. Оно предназначено для восприятия человеком. Вся информация, существенная для автоматизированных функций, должна передаваться в надлежащих атрибутах и связанных объектах.

Обоснование. Имена и описания сущностей обычно более информативны для человека, читающего информацию, нежели числовые, мнемонические или сокращенные кодированные значения. В атрибуте desc можно передать дополнительный контекст сущности, предназначенный человеку и не влияющий на функциональные компоненты сообщения.

A.3.2.3.8 Entity.statusCode:: CS (0..1)

Словарный домен: EntityStatus

Определение. Значение, указывающее, является ли информация, связанная с экземпляром класса Entity, активной или неактивной в отношении возможности участия в действиях.

Совет разработчикам: в исходной модели RIM этот атрибут был определен как повтояющийся, чтобы можно было отразить наличие вложенных подсостояний, указанных на диаграмме перехода состояний. На практике, однако, необходимость передачи нескольких значений состояния никогда не возникала. Поэтому комитетам рекомендуется ограничить кратность этого атрибута до 1 во всех конструкциях сообщений.

A.3.2.3.9 Entity.existenceTime:: IVL<TS> (0..1)

Определение. Интервал времени физического существования сущности.

Обоснование. Физические сущности имеют определенные периоды существования. Люди рождаются, живут и умирают. Оборудование производится, вводится в эксплуатацию, выводится из эксплуатации и списывается. Значение этого атрибута полезно учитывать при планировании и оценке доступности, а также при ретроспективном анализе.

Примеры — Дата рождения/дата смерти, дата производства/дата списания.

Ограничения. Этот интервал может указывать прошлые, настоящие и будущие периоды времени.

A.3.2.3.10 Entity.telecom:: BAG<TEL> (0..*)

Определение. Телекоммуникационный адрес сущности.

A.3.2.3.11 Entity.riskCode:: SET<CE> (0..*)

Словарный домен: EntityRisk

Определение. Значение, указывающее тип риска или опасности, связанной с сущностью.

Обоснование. Некоторые сущности обладают свойствами, представляющими потенциально повышенную опасность нанесения вреда или повреждения другой сущности. В атрибуте riskCode передается тип риска без указания степени риска.

Примеры — Нефтехимические или органические химикаты являются очень огнеопасными, что в определенных условиях увеличивает риск пожара. Материалы с естественными или привнесенными радиоактивными свойствами представляют риск для тех, кто с ними обращается. Предметы, содержащие биоматериал больных людей, представляют повышенный риск инфекции для тех, кто обращается с ними. Вспыльчивые люди или животные могут представлять опасность для медицинского персонала.

A.3.2.3.12 Entity.handlingCode:: SET<CE> (0..*)

Словарный домен: EntityHandling

Определение. Значение, указывающее специальные требования обработки сущности.

Обоснование. Этот атрибут используется для описания требования специальной обработки сущности во избежание ее повреждения или повреждения других сущностей.

Примеры — Хранить при комнатной температуре, хранить замороженным при температуре ниже 0 °С, хранить в сухом месте; хранить вертикально, не переворачивая вверх дном.

A.3.2.3.13 Переходы состояний экземпляра класса Entity

Диаграмма перехода состояний класса Entity показана на рисунке А.6.

Сущность может иметь следующие состояния:

- active (активное) — подсостояние состояния normal: сущность активна;
- inactive (не активное) — подсостояние состояния normal: сущность более не может быть активным участником событий;

- normal (нормальное): охватывает все ожидаемые состояния объекта услуги, за исключением nullified, которое представляет терминальные состояния экземпляра класса Entity, созданного по ошибке;
- nullified (аннулированное): терминальное состояние экземпляра класса Entity, созданного по ошибке.

Между состояниями действия возможны следующие переходы:
- revise (пересмотреть) — из состояния active в состояние active;
- inactivate (сделать неактивным) — из состояния active в состояние inactive;
- reactivate (активировать заново) — из состояния inactive в состояние active;
- revise (пересмотреть) — из состояния inactive в состояние inactive;
- nullify (аннулировать) — из состояния normal в состояние nullified;
- create (создать) — из начального (пустого) состояния в состояние active.

A.3.2.4 Класс LanguageCommunication (в предметной области Entities)

Атрибуты класса LanguageCommunication:
- languageCode:: CE,
- modeCode:: CE,
- proficiencyLevelCode:: CE,
- preferenceInd:: BL.

Ассоциации класса LanguageCommunication:
- entity::(1..1) Entity:: languageCommunication:: (0..*) (ассоциация с классом Entity, роль languageCommunication — язык общения).

Определение класса LanguageCommunication

Способности сущности к языковому общению.

Обоснование. Каждая сущность, способная к вербальному общению, обладает определенными степенями владения разными языками. В экземплярах класса LanguageCommunication можно указать языки, которыми владеет сущность, способ общения (речь, чтение, письмо), степень владения этим способом и предпочтительный язык общения.

Примеры — Пациент, ранее приехавший из Мексики, может иметь беглые языковые возможности чтения и письма на испанском языке, но при этом имеет только самые начальные способности общения на английском языке. Выходец из России может одинаково хорошо владеть разговорным русским, армянским или украинским языком, но предпочитает говорить на армянском языке.

Обсуждение. Хотя на первый взгляд использование этого класса должно быть ограничено только подтипами класса LivingSubject (живой организм), но устройства, представляемые классом Device, также могут использовать языковое общение, например, автоматизированные телефонные устройства, перенаправляющие пациента к живым операторам в зависимости от его нужд, или устройства, автоматически читающие врачам результаты лабораторных анализов.

Атрибуты класса LanguageCommunication

A.3.2.4.1 LanguageCommunication.languageCode:: CE (0..1)

Словарный домен: HumanLanguage

Определение. Значение, указывающее язык, которым сущность в определенной степени владеет для разговорного или письменного общения.

Обоснование. Многие люди и устройства способны общаться на разных языках, имея разные степени владения языками. Значение атрибута languageCode указывает способность к языковому общению, декларируемую сущностью.

Примеры — Испанский, итальянский, немецкий, английский, американский язык жестов и т.д.

Обсуждение. Разговорное или письменное языковое общение присуще не только живым существам. Для устройств, которые взаимодействуют с людьми, используя человеческий язык, также надо указать, на какие языки они рассчитаны. Автоматизированные системы речевого ответа передают сообщения на естественном языке и взаимодействуют с другими устройствами или людьми, используя естественный язык.

A.3.2.4.2 LanguageCommunication.modeCode:: CE (0..1)

Словарный домен: LanguageAbilityMode

Определение. Значение, указывающее метод применения языка.

Примеры — Речевое выражение, письменное выражение, выражение на языке жестов, восприятие речи, восприятие письменного языка, восприятие языка жестов.

A.3.2.4.3 LanguageCommunication.proficiencyLevelCode:: CE (0..1)

Словарный домен: LanguageAbilityProficiency

Определение. Значение, указывающее степень владения языком.

Примеры — Свободно, хорошо, удовлетворительно, плохо.

A.3.2.4.4 LanguageCommunication.preferenceInd:: BL (0..1)

Определение. Признак, указывающий, является ли язык предпочтительным для метода применения, указанного в атрибуте modeCode.

A.3.2.5 Класс LivingSubject (в предметной области Entities)

Свойства класса LivingSubject

Атрибуты класса LivingSubject:

- administrativeGenderCode:: CE,
- birthTime:: TS,
- deceasedInd:: BL,
- deceasedTime:: TS,
- multipleBirthInd:: BL,
- multipleBirthOrderNumber:: INT,
- organDonorInd:: BL.

Класс LivingSubject является генерализацией классов NonPersonLivingSubject, Person.

Класс LivingSubject является специализацией класса Entity.

Определение класса LivingSubject

Специализация класса Entity, описывающая организм или сложное животное, живое или мертвое.

Обоснование. Этот класс содержит «статические» или «административные» атрибуты, представляющие интерес для медицины и отличающие живые организмы от других сущностей.

Примеры — Человек, собака, микроорганизм или растение любой таксономической группы.

Ограничения. Экземпляры этого класса могут содержать информацию о млекопитающих, птицах, рыбах, бактериях, паразитах, грибах и вирусах. Класс Person (человек) является специализацией этого класса.

Атрибуты класса LivingSubject

A.3.2.5.1 LivingSubject.administrativeGenderCode:: CE (0..1)

Словарный домен: AdministrativeGender

Определение. Значение, указывающее пол живого организма.

Примеры — Женский, мужской

Обсуждение. Значения этого атрибута не включают в себя термины, относящиеся к клиническому полу. Пол — это сложное физиологическое, генетическое и социологическое понятие, для всестороннего описания которого необходим ряд наблюдений. Цель этого атрибута — обеспечить высокоуровневую классификацию, которая дополнительно может использоваться для соответствующего размещения пациентов на больничные койки.

Ограничение. Этот код используется в административных целях.

A.3.2.5.2 LivingSubject.birthTime:: TS (0..1)

Определение.

Дата и время рождения или выведения живого организма.

A.3.2.5.3 LivingSubject.deceasedInd:: BL (0..1)

Определение. Признак смерти организма.

A.3.2.5.4 LivingSubject.deceasedTime:: TS (0..1)

Определение. Дата и время смерти живого организма.

A.3.2.5.5 LivingSubject.multipleBirthInd:: BL (0..1)

Определение. Признак, появился ли живой организм в результате многоплодных родов.

A.3.2.5.6 LivingSubject.multipleBirthOrderNumber:: INT (0..1)

Определение. Порядок рождения живого организма при многоплодных родах.

A.3.2.5.7 LivingSubject.organDonorInd:: BL (0..1)

Определение. Признак того, что живой организм является кандидатом в доноры органа.

Обсуждение. Этот атрибут указывает, пожертвовал ли данный живой организм свой орган или готов его пожертвовать.

A.3.2.6 Класс ManufacturedMaterial (в предметной области Entities)

Свойства класса ManufacturedMaterial

Атрибуты класса ManufacturedMaterial:

- lotNumberText:: ST,
- expirationTime:: IVL<TS>,
- stabilityTime:: IVL<TS>.

Класс ManufacturedMaterial является генерализацией классов Container, Device.

Класс ManufacturedMaterial является специализацией класса Material.

Определение класса ManufacturedMaterial

Специализация класса Material, описывающая сущность или комбинацию сущностей, преобразованную в специальных целях с помощью искусственного или производственного процесса.

Обоснование. Этот класс используется для дальнейшего определения характеристик сущностей, которые созданы из других сущностей. Эти сущности идентифицируются и отслеживаются с помощью ассоциаций и механизмов, уникальных для данного класса, например, lotName, stabilityTime и expirationTime.

Примеры — Обработанные пищевые продукты, одноразовые шприцы, химический анализатор, физиологический раствор и т.д.

Обсуждение. К понятиям этого класса относятся контейнеры, медицинские изделия, программные модули и средства.

Атрибуты класса ManufacturedMaterial

A.3.2.6.1 ManufacturedMaterial.lotNumberText:: ST (0..1)

Определение. Алфавитно-цифровая строка, используемая для идентификации конкретной партии изготовленного материала.

Обсуждение. Номер партии обычно печатается на ярлыке, приложенном к контейнеру, содержащему субстанцию, и/или на упаковке, в которую вложен контейнер. Следует отметить, что номер партии не обязан быть уникальным идентификатором, он имеет значение только в сочетании с видом продукта и производителем.

A.3.2.6.2 ManufacturedMaterial.expirationTime:: IVL<TS> (0..1)

Определение. Дата и время, после которого производитель больше не гарантирует безопасность, качество, и/или надлежащее функционирование материала.

Обоснование. Во многих ситуациях требуется, чтобы используемые материалы имели конкретное качество или обладали определенной эффективностью, либо имели необходимое функциональное состояние. Конечная дата этой гарантии определяется производителем. Хотя после указанной даты материал может все еще обладать теми же самыми свойствами, производитель с этого момента снимает с себя ответственность за надлежащее функционирование материала и возможные последствия его использования.

A.3.2.6.3 ManufacturedMaterial.stabilityTime:: IVL<TS> (0..1)

Определение. Время, в течение которого материал считается пригодным к использованию после активирования.

Примеры — Время годности жидкости после открытия флакона; смесь двух химических реагентов для анализа, которая должна быть использована в течение двух часов после смешивания, иначе ее активность снижается.

Обсуждение. Если экземпляр класса ManufacturedMaterial описывает вид материала (его атрибут determinerCode имеет значение «KIND»), то может быть известна только длина интервала, например, время после открытия контейнера с реагентом, в течение которой вещество реагента считается пригодным для употребления в его стандартном аналитическом применении. Для определенного экземпляра реагента (например, конкретный флакон), значение stabilityTime.low типа TS указывает время, когда флакон был открыт (или реагент был активирован иным образом). Добавляя к нему типичное время стабильности, можно получить значение stabilityTime.high типа TS, после которого реагент уже не считается пригодным для употребления в его стандартном аналитическом применении.

A.3.2.7 Класс Material (в предметной области Entities)

Свойства класса Material

Атрибуты класса Material:

- formCode:: CE.

Класс Material является генерализацией класса ManufacturedMaterial.

Класс Material является специализацией класса Entity.

Определение класса Material

Специализация класса Entity, описывающая неодушевленную сущность, не зависящую от местонахождения.

Обоснование. Есть сущности, которые обладают дополнительными атрибутами по сравнению с классом Entity, но не могут быть классифицированы как живой организм или местонахождение.

Примеры — Фармацевтические субстанции (включая активные вакцины, содержащие ослабленные вирусные клетки), одноразовые предметы, оборудование длительного пользования, имплантанты, продукты питания (включая мясные или растительные продукты), отходы, продаваемые товары и т.д.

Обсуждение. Изготовленные или обработанные продукты считаются материалом, даже если они происходят от живой материи. Существует широкое разнообразие физических форм материалов, которые могут принимать различные состояния (например, газ, жидкость, твердое вещество), все еще сохраняя свой физический состав и материальные характеристики.

Атрибуты класса Material

A.3.2.7.1 Material.formCode:: CE (0..1)

Словарный домен: MaterialForm

Определение. Значение, указывающее состояние (твердое, жидкое, газообразное) и природу материала.

Примеры — Для терапевтических субстанций могут быть указаны такие формы, как таблетка, мазь, гель и т.д.

A.3.2.8 Класс NonPersonLivingSubject (в предметной области Entities)

Свойства класса NonPersonLivingSubject

Атрибуты класса NonPersonLivingSubject:

- strainText:: ED,

- genderStatusCode:: CE.

Класс NonPersonLivingSubject является специализацией класса LivingSubject.

Определение класса NonPersonLivingSubject

Специализация класса LivingSubject, описывающая все живые организмы, кроме человека.

Обоснование. Для описания живого организма, не являющегося человеком, может понадобиться дополнительная характерная информация, например, генетическая идентификация рода, которая не может быть передана в атрибуте code.

Примеры — Крупный рогатый скот, птицы, бактерии, формы растений и грибы и т. д.

Атрибуты класса NonPersonLivingSubject

A.3.2.8.1 NonPersonLivingSubject.strainText:: ED (0..1)

Определение. Текстовая строка, указывающая определенный генотипический или фенотипический вариант организма, информация о котором передается в экземпляре класса NonPersonLivingSubject.

Примеры — Миннесота 5 (порода поросят), DXL (порода домашней птицы), RB51 (штамм противобруцеллезной вакцины).

Обоснование. Универсальное руководство по именованию или каталогизированию породы, вида или штамма организмов отсуствует. Многие обозначения пород, видов и штаммов создавались, а затем исчезали, а другие по разным причинам приживались в различных областях производства (изготовление вакцин, племенное животноводство и т. д.). И поскольку каждый, кому это нужно, может обозначить организм как «новый» вид, вместо кодированного значения данный атрибут должен содержать описательный текст, в котором можно передать все такие обозначения.

A.3.2.8.2 NonPersonLivingSubject.genderStatusCode:: CE (0..1)

Словарный домен: GenderStatus

Определение. Значение, указывающее, имеются ли основные репродуктивные органы у организма, информация о котором передается в экземпляре класса NonPersonLivingSubject.

A.3.2.9 Класс Organization (в предметной области Entities)

Свойства класса Organization

Атрибуты класса Organization:

- addr:: BAG<AD>,
- standardIndustryClassCode:: CE.

Класс Organization является специализацией класса Entity.

Определение класса Organization

Специализация класса Entity, описывающая формализованную группу сущностей с общей целью (например, административной, юридической, политической) и инфраструктуру, необходимую для осуществления этой цели.

Примеры — Предприятия и организации, правительственная структура, корпоративный орган, ответственный за управление учреждением, страховая компания.

Атрибуты класса Organization

A.3.2.9.1 Organization.addr:: BAG<AD> (0..*)

Определение. Почтовый и/или юридический адрес организации.

A.3.2.9.2 Organization.standardIndustryClassCode:: CE (0..1)

Словарный домен: OrganizationIndustryClass

Определение. Значение, указывающее код организации согласно классификатору отраслей народного хозяйства.

Примеры — 11231 — производство куриных яиц, 6211 — амбулаторная помощь, 621511 — медицинские лаборатории, 524114 — непосредственное страхование здоровья и медицинской помощи.

A.3.2.10 Класс Person (в предметной области Entities)

Свойства класса Person

Атрибуты класса Person:

- addr:: BAG<AD>,
- maritalStatusCode:: CE,
- educationLevelCode:: CE,
- disabilityCode:: SET<CE>,
- livingArrangementCode:: CE,
- religiousAffiliationCode :: CE,
- raceCode:: SET<CE>,
- ethnicGroupCode:: SET<CE>.

Класс Person является специализацией класса LivingSubject.

Определение класса Person

Специализация класса LivingSubject, описывающая человека.

Ограничения. В зависимости от значений атрибутов determinerCode и quantity экземпляр этого класса может содержать информацию об одном лице или о группе лиц.

Атрибуты класса Person

A.3.2.10.1 Person.addr:: BAG<AD> (0..*)

Определение. Почтовый адрес или адрес места жительства лица.

A.3.2.10.2 Person.maritalStatusCode:: CE (0..1)

Словарный домен: MaritalStatus

Определение. Значение, указывающее статус семейного партнерства лица.

Обоснование. Эта информация передается в поле FL 16 универсального счета на оплату лечения (UB).

Примеры — Женат (замужем), раздельное проживание, в разводе, вдовец (вдова), гражданский брак.

A.3.2.10.3 Person.educationLevelCode:: CE (0..1)

Словарный домен: EducationLevel

Определение. Самый высокий уровень образования, достигнутый лицом (например, начальная школа, средняя школа или законенное среднее образование, колледж или законченное обучение в бакалавриате).

A.3.2.10.4 Person.disabilityCode:: SET<CE> (0..*)

Словарный домен: PersonDisabilityType

Определение. Значение, указывающее инвалидность человека.

Примеры — Ослабленное зрению, ослабленный слух.

A.3.2.10.5 Person.livingArrangementCode:: CE (0..1)

Словарный домен: LivingArrangement

Определение. Значение, указывающее условия проживания лица.

Примеры — Собственный дом, организация, интернат, реабилитационный центр, дом престарелых и т. д.

Обсуждение. Используется при планировании выписки, оценке необходимости социального обеспечения, психосоциальной оценке.

A.3.2.10.6 Person.religiousAffiliationCode:: CE (0..1)

Словарный домен: ReligiousAffiliation

Определение. Основное религиозное предпочтение лица (например, индуизм, ислам, римско-католическая церковь).

A.3.2.10.7 Person.raceCode:: SET<CE> (0..*)

Словарный домен: Race

Определение. Значение, указывающее расу лица.

A.3.2.10.8 Person.ethnicGroupCode:: SET<CE> (0..*)

Словарный домен: Ethnicity

Определение. Этническая группа лица.

A.3.2.11 Класс Place (в предметной области Entities)

Свойства класса Place

Атрибуты класса Place:

- mobileInd:: BL,
- addr:: BAG<AD>,
- directionsText:: ED,
- positionText:: ED,
- gpsText:: ST.

Класс Place является специализацией класса Entity.

Определение класса Place

Специализация класса Entity, описывающая ограниченное физическое место или участок с содержащимися в нем структурами, если таковые имеются.

Примеры — Поле, озеро, город, графство, штат, страна, участок (земли), здание, трубопровод, линия электропередачи, детская площадка, судно, грузовик.

Ограничения. Место может быть естественным или созданным человеком. Географическое положение места может быть непостоянным.

Обсуждение. Места могут быть производственными зданиями (где выполняются релевантные действия), домами (где живут люди) или конторами (где люди работают). Места могут содержать другие места (этаж, палата, кабина, койка). Экземпляры класса Place могут содержать информацию о местах, идентифицируемых в контексте оказания медицинской помощи, социальной помощи, санитарно-эпидемиологических мероприятий (например, здания, поляны для пикника, дневные стационары, тюрьмы, округа, штаты, и другие места эпидемиологических событий).

Атрибуты класса Place

A.3.2.11.1 Place.mobileInd:: BL (0..1)

Определение. Признак того, может ли место свободно перемещаться из одного местоположения в другое.

Примеры — Перемещаемыми местами оказания медицинской помощи могут быть суда, самолеты и санитарные машины.

A.3.2.11.2 Place.addr:: AD (0..1)

Определение. Физический адрес данного места.

Ограничения. Значением этого атрибута должен быть адрес, с помощью которого можно определить физическое положение места на карте.

A.3.2.11.3 Place.directionsText:: ED (0..1)

Определение. Свободный текст, содержащий информацию о данном месте, которая полезна для сущностей при поиске этого места.

Обсуждение. В этом атрибуте могут передаваться указания для поиска места, если информация об адресе не адекватна, его координаты в глобальной системе навигации (GPS) недоступны и/или сущность, которая ищет это место, не может непосредственно использовать координаты GPS. В нем можно передавать информацию, полезную для посетителей места. Например, «последний дом справа», «если владельца нет, узнайте в следующем доме, где его найти».

A.3.2.11.4 Place.positionText:: ED (0..1)

Определение. Совокупность кодов, задающих расположение места на карте.

Примеры — координаты на картах, издаваемых агентством US Geological Survey.

A.3.2.11.5 Place.gpsText:: ST (0..1)

Определение. Координаты места в глобальной системе навигации (GPS).

Обсуждение. Значения координат GPS должны соответствовать стандартам USGS Spatial Data Transmission (передача пространственных данных). К числу этих параметров относятся способы получения значений широты и долготы, ошибки сдвига, проекция.

Обоснование. В некоторых полевых условиях физический адрес интересующего места может отсутствовать. Поскольку все места поверхности земли имеют уникальные географические координаты, для точного определения и передачи информации о положении места можно использовать GPS-координаты.

A.3.3 Классы предметной области Roles

A.3.3.1 Класс Access (в предметной области Roles)

Свойства класса Access

Атрибуты класса Access:

- approachSiteCode:: CD,
- targetSiteCode:: CD,
- gaugeQuantity:: PQ.

Класс Access является специализацией класса Role.

Определение класса Access

Специализация класса Role, описывающая роль устройства, используемого для введения терапевтических агентов (лекарств и элементов, поддерживающих жизнь) в тело пациента, или изделия, используемого для вывода веществ (например, экссудатов, гноя, мочи, воздуха, крови) из его тела.

Обсуждение. В общем случае класс Access описывает роль произведенного материала или изделия, иногда специально изготовленного или созданного для определенной цели, например, роль катетера или канюли, помещенной в полый орган тела.

Изделия, выполняющие роль доступа к органам тела, обычно используются в наблюдениях поглощения/оттока, а также для введения лекарственных средств. Микробиологические исследования самого материала или жидкостей, выделяемых из дренажа, также являются обычными примерами.

Обоснование. Роль доступа, в основном, используется для описания материала, фактически применяемого для доступа, а не нового материала, полученного от изготовителя. Например, для представления заказа коробки катетеров у поставщика не обязательно использовать класс роли Access, так как атрибутов класса Material обычно вполне достаточно для описания и идентификации заказываемого изделия. А экземпляры класса Access используются для передачи данных об эксплуатации, поглощении/оттоке и своевременной замене трубок и дренажа.

Атрибуты класса Access

A.3.3.1.1 Access.approachSiteCode:: CD (0..1)

Словарный домен: ActSite

Определение. Кодированное представление анатомической локализации места, через которое изделие, участвующее в доступе, впервые проникает в тело и, если применимо, маршрут от места первого входа к целевому месту.

Обоснование. Поскольку изделия, обеспечивающие доступ, обычно помещаются в тело на значительный период времени и используются как ресурс для многих действий, то целевое место доступа становится важным идентифицирующим атрибутом самого доступа (в противоположность тому, когда оно является атрибутом процедуры введения).

Примеры — Катетер легочной артерии предназначен для легочной артерии, но при этом местом доступа обычно является внутренняя яремная вена в шее или подключичная вена в подключичной ямке.

Ограничения. Система кодирования значений атрибута Access.approachSiteCode та же, что у атрибута Procedure.approachSiteCode; действительно, атрибут approachSiteCode был скопирован из класса Procedure в класс Access. Значение атрибута Access.approachSiteCode должно быть идентично значению атрибута Procedure. approachSiteCode соответствующей процедуры введения изделия.

A.3.3.1.2 Access.targetSiteCode:: CD (0..1)

Словарный домен: ActSite

Определение. Кодированное указание анатомической локализации места или части тела, куда обеспечивается доступ, т. е. часть, в которую вводится материал, или часть, из которой он выводится.

Обоснование. Поскольку изделия, обеспечивающие доступ, обычно помещаются в тело на значительный период времени и используются как ресурс для многих действий, то целевое место доступа становится важным идентифицирующим атрибутом самого доступа (в противоположность тому, когда оно является атрибутом процедуры введения). Целевое место является важной информацией, по которой можно установить вид субстанций, которые могут или не могут быть применены в этом месте (например, должны быть предприняты специальные меры во избежание инъекции лекарства в изделие, находящееся в артерии).

Примеры — Катетер легочной артерии будет иметь целевым местом «легочную артерию».

Ограничения. Система кодирования значений атрибута Access.targetSiteCode та же, что у атрибута Procedure. targetSiteCode; действительно, атрибут targetSiteCode был скопирован из класса Procedure в класс Access. Значение атрибута Access.targetSiteCode должно быть идентично значению атрибута Procedure.targetSiteCode соответствующей процедуры введения устройства.

A.3.3.1.3 Access.gaugeQuantity:: PQ (0..1)

Определение. Мера внутреннего диаметра устройства доступа (например, просвет трубки).

A.3.3.2 Класс Employee (в предметной области Roles)

Свойства класса Employee

Атрибуты класса Employee:

- jobCode:: CE,
- jobTitleName:: SC,
- jobClassCode:: CE,
- occupationCode :: CE,
- salaryTypeCode:: CE,
- salaryQuantity:: MO,
- hazardExposureText:: ED,
- protectiveEquipmentText:: ED.

Класс Employee является специализацией класса Role.

Определение класса Employee

Специализация класса Role, описывающая роль лица, связанного с организацией (работодателем в роли контролера) для получения заработной платы рабочего или служащего.

Обсуждение. Класс Employee предназначен для описания трудовых отношений между работником и работодателем, а не для описания характера фактически выполненной работы (в противоположность классу AssignedEntity).

Атрибуты класса Employee

A.3.3.2.1 Employee.jobCode:: CE (0..1)

Словарный домен: EmployeeJob

Определение. Код, указывающий вид работы, классифицируемый работодателем и используемый в первую очередь для целей установления зарплаты/вознаграждения, а не для определения специфики выполняемой работы, ответственности и полномочий работника. Например, бухгалтер, аналитик-программист, сиделка, штатная медсестра и т. д.

A.3.3.2.2 Employee.jobTitleName:: SC (0..1)

Словарный домен: JobTitleName

Определение. Название занимаемой должности, например, вице-президент, старший технический аналитик. Это местное название должности работника, которое не обязательно соответствует какой-либо номенклатуре должностей. Торговые партнеры, которым требуется кодированное представление, должны использовать атрибут occupationCode.

A.3.3.2.3 Employee.jobClassCode:: CE (0..1)

Словарный домен: EmployeeJobClass

Определение. Код, указывающий частоту или периодичность занятости, например, полная ставка или совместитель.

A.3.3.2.4 Employee.occupationCode :: CE (0..1)

Словарный домен: EmployeeOccupationCode

Определение. Код, указывающий классификацию вида работы, основанную на территориальной или отраслевой номенклатуре.

A.3.3.2.5 Employee.salaryTypeCode:: CE (0..1)

Словарный домен: EmployeeSalaryType

Определение. Код, указывающий метод начисления заработной платы рабочему или служащему. Например, почасовая ставка, годовая ставка, комиссионные.

A.3.3.2.6 Employee.salaryQuantity:: MO (0..1)

Определение. Сумма заработной платы рабочего или служащего, начисляемая в соответствии с методом, указанным в значении атрибута salaryTypeCode. Например, если атрибут salaryTypeCode имеет значение «почасовая», то в атрибуте salaryQuantity передается сумма почасовой ставки.

A.3.3.2.7 Employee.hazardExposureText:: ED (0..1)

Определение. Тип вредности выполняемой работы. Например, асбест, инфекционные агенты.

A.3.3.2.8 Employee.protectiveEquipmentText:: ED (0..1)

Определение. Средства защиты, необходимые для выполнения работы. Например, защитные очки, каска.

A.3.3.3 Класс LicensedEntity (в предметной области Roles)

Свойства класса LicensedEntity

Атрибуты класса LicensedEntity:

- recertificationTime:: TS.

Класс LicensedEntity является специализацией класса Role.

Определение класса LicensedEntity

Специализация класса Role, описывающая роль сущности (исполнителя), обладающей лицензией или квалификацией (дипломом), подтверждающей право данной сущности выполнять специфичные функции.

Примеры

1 Диплом фельдшера скорой помощи.

2 Сертификат безопасности оборудования.

3 Сертификат специалиста или лицензия организации на право осуществления медицинской деятельности.

Ограничения. Контролером является организация, выпускающая диплом, сертификат или лицензию.

Атрибуты класса LicensedEntity

A.3.3.3.1 LicensedEntity.recertificationTime:: TS (0..1)

Определение. Требуемая дата повторной сертификации.

A.3.3.4 Класс Patient (в предметной области Roles)

Свойства класса Patient

Атрибуты класса Patient:

- veryImportantPersonCode:: CE.

Класс Patient является специализацией класса Role.

Определение класса Patient

Специализация класса Role, описывающая роль живого существа (исполнителя роли), получающего медицинскую помощь от ее поставщика (контролера роли).

Атрибуты класса Patient

A.3.3.4.1 Patient.veryImportantPersonCode:: CE (0..1)

Словарный домен: PatientImportance

Определение. Код, указывающий особый статус пациента, предоставленный ему контролирующей организацией, часто обеспечивающий приоритетное лечение и особые условия. Например, член правления, дипломат и т. д.

A.3.3.5 Класс Role (в предметной области Roles)

Свойства класса Role

Атрибуты класса Role:

- classCode:: CS,
- id:: SET<II>,
- code:: CE,
- negationInd:: BL,
- name:: BAG<EN>,
- addr:: BAG<AD>,
- telecom:: BAG<TEL>,
- statusCode:: CS,
- effectiveTime:: IVL<TS>,
- certificateText:: ED,
- confidentialityCode :: SET<CE>,
- quantity:: RTO,
- positionNumber:: LIST<INT>.

Ассоциации класса Role:

- participation::(0..*) Participation::role::(1..1) (ассоциация с классом Participation, роль role — роль);
- player::(0..1) Entity::playedRole::(0..*) (ассоциация с классом Entity, роль playedRole — выполняемая роль);
- scoper::(0..1) Entity::scopedRole::(0..*) (ассоциация с классом Entity, роль scopedRole — контролируемая роль);
- outboundLink::(0..*) RoleLink::source::(1..1) (ассоциация с классом RoleLink, роль source — источник);
- inboundLink::(0..*) RoleLink::target::(1..1) (ассоциация с классом RoleLink, роль target — цель).

Класс Role является обобщением следующих классов:

- Access,
- Employee,
- LicensedEntity,
- Patient.

Определение класса Role

Компетенция сущности, выполняющей роль в соответствии с указаниями, определениями, гарантиями или признанием другой сущности, контролирующей эту роль.

Обсуждение. Сущность участвует в действии, выполняя определенную роль. Следует отметить, что конкретная сущность в конкретной роли может участвовать в действии многими способами. Например, лицо в роли врача может участвовать в ведении пациента как дежурант или как лечащий врач. Роль определяет компетенцию сущности, не зависящую от любого действия, в противоположность участию, которое ограничено рамками действия.

Каждая роль «выполняется» одной сущностью, называемой «исполнителем», и «контролируется» другой сущностью, называемой «контролером». Например, роль «пациент» может выполняться некоторым лицом и контролироваться организацией, оказывающей пациенту медицинскую помощь. Аналогичным образом работодатель контролирует роль «работника».

Идентификатор роли идентифицирует сущность, выполняющую эту роль. Этот идентификатор присваивается исполнителю контролирующей сущностью. Контролирующая сущность не обязана создавать новые идентификаторы. Она может повторно использовать идентификатор, ранее присвоенный контролируемой сущности, чтобы идентифицировать эту сущность в контролируемой ею роли.

Большинство атрибутов класса Role представляют собой характеристики исполнителя роли, присущие ему в процессе выполнения конкретной роли.

Атрибуты класса Role

A.3.3.5.1 Role.classCode:: CS (1..1) Mandatory

Словарный домен: RoleClass

Определение. Код, указывающий основную категорию роли в соответствии со словарем, определенным в стандарте HL7.

A.3.3.5.2 Role.id:: SET<II> (0..*)

Определение. Уникальный идентификатор сущности, выполняющей данную роль.

A.3.3.5.3 Role.code:: CE (0..1)

Словарный домен: RoleCode

Определение. Код, уточняющий вид роли.

Обсуждение. Значения атрибута Role.code должны концептуально представлять собой специализации понятия, указанного в атрибуте Role.classCode. Атрибут Role.code не должен использоваться как модификатор значения атрибута Role.classCode. Каждый из этих атрибутов представляет завершенное понятие или ролевые отношения между двумя сущностями, но значение атрибута Role.code может быть более специфичным, нежели значение атрибута Role.classCode.

Значение Role.code может быть не кодированным, если данный тип роли обычно обозначается только не кодированным именем.

A.3.3.5.4 Role.negationInd:: BL (0..1)

Определение. Признак, указывающий, что данный экземпляр класса Role описывает компетенцию, которая отсутствует у сущности, выполняющей эту роль.

Примеры

1 Это лицо не является нашим работником.

2 Эта жидкость для полоскания рта не содержит алкоголь в качестве ингредиента.

Ограничение. Обычно все роли рассматриваются как утвердительные (по умолчанию этот атрибут имеет значение FALSE).

A.3.3.5.5 Role.name:: BAG<EN> (0..*)

Определение. Неуникальный текстовый идентификатор или псевдоним сущности, предназначенный прежде всего для использования во время выполнения роли.

A.3.3.5.6 Role.addr:: BAG<AD> (0..*)

Определение. Адрес сущности во время выполнения роли.

A.3.3.5.7 Role.telecom:: BAG<TEL> (0..*)

Определение. Телекоммуникационный адрес сущности во время выполнения роли.

A.3.3.5.8 Role.statusCode:: CS (0..1)

Словарный домен: RoleStatus

Определение. Код, указывающий состояние данной роли, определенное в модели перехода состояний.

Совет разработчикам: в исходной модели RIM этот атрибут был определен как повтояющийся, чтобы можно было отразить наличие вложенных подсостояний, указанных на диаграмме перехода состояний. На практике, однако, необходимость передачи нескольких значений состояния никогда не возникала. Поэтому комитетам рекомендуется ограничить кратность этого атрибута до 1 во всех конструкциях сообщений.

A.3.3.5.9 Role.effectiveTime:: IVL<TS> (0..1)

Определение. Интервал времени, указывающий время выполнения роли, если такое время применимо и известно.

A.3.3.5.10 Role.certificateText:: ED (0..*)

Определение. Текстовое или мультимедийное представление сертификата, изданного сущностью, контролирующей данную роль, и удостоверяющего, что данная роль действительно принадлежит исполнителю.

Пример — Сертификат может быть представлен многими разными способами либо по значению, либо по ссылке в соответствии с типом данных ED. Типичные случаи таковы:

1 бумажный сертификат: значение, имеющее тип данных ED, может содержать ссылку на некоторый документ или файл, который может быть найден с помощью электронного интерфейса к архиву бумажных документов.

2 электронный сертификат: в этом атрибуте можно передать практически любую электронную схему сертификата, например, электронный текстовый документ с электронной подписью (в том числе стандартной).

3 цифровой сертификат (сертификат открытого ключа): в частности, в этом атрибуте можно передавать цифровые сертификаты по значению или по ссылке. Блок данных сертификата может конструироваться в соответствии со стандартом цифровых сертификатов, например, X.509, SPKI, PGP и т. д.

Субъектом сертификата является сущность, выполняющая данную роль. Издателем сертификата является сущность, контролирующая данную роль.

A.3.3.5.11 Role.confidentialityCode:: SET<CE> (0..*)

Словарный домен: Confidentiality

Определение. Код, контролирующий раскрытие информации о данной роли сущности.

Обсуждение. Важно отметить, что необходимая конфиденциальность медицинской карты не может быть достигнута только с помощью кодов конфиденциальности, предназначенных для маскировки отдельных частей этой карты от определенных типов пользователей. При обеспечении конфиденциальности на основе кодов конфиденциальности, присваиваемых отдельным частям карты, возникают две проблемы: одна связана с возможностью логического вывода, а другая — с риском отсутствия доступа к информации, которая может быть критичной в определенных случаях оказания медицинской помощи. Возможность вывода означает, что фильтрованная чувствительная информация может быть выведена из другой, не отфильтрованной информации. Простейшей формой вывода может служить пример, когда из факта наличия направления на анализ иммуноблота или на анализ количества Т4- и Т8-клеток с большой вероятностью можно сделать вывод, что пациент ВИЧ-инфицирован, даже если результат анализа не известен. Очень часто диагнозы могут быть выведены из лекарственных назначений, например, назначение зидовудина свидетельствует о лечении ВИЧ. Проблема сокрытия отдельных частей медицинской карты становится особенно трудной при наличии текущих лекарственных назначений, поскольку должно быть обеспечено продолжение приема лекарств.

Чтобы ослабить некоторые риски вывода, следует считать, что агрегированные данные имеют уровень конфиденциальности самого конфиденциального действия в агрегате.

A.3.3.5.12 Role.quantity:: RTO (0..1)

Определение. Отношение (числитель:знаменатель), указывающее относительное количество исполнителя роли в сущности, контролирующей роль. Оно используется для ролей, описывающих отношения композиции между контролирующими сущностями и исполнителями.

Примеры

1 Ингредиентом этого сиропа (контролирующая сущность) являются 160 мг (числитель) ацетаминофена (исполнитель) на столовую ложку (знаменатель).

2 Стадо (контролирующая сущность) состоит из 500 (числитель) голов крупного рогатого скота (исполнитель).

3 У компьютера VAX 6630 (контролирующая сущность) имеются 3 (числитель) центральных процессора (исполнитель).

4 Эта упаковка (контролирующая сущность) содержит 100 (числитель) таблеток (исполнитель).

Обсуждение. В отношениях композиции (например, имеет части, имеет ингредиенты, имеет содержание) значение атрибута Role.quantity указывает, что количество числителя целевой сущности отношения содержит в себе количество сущности-источника, указанное в знаменателе. Например, если коробка (источник отношения) содержит 10 яиц (цель отношения), то атрибут quantity должен иметь значение 10:1; если 0,6 мл смеси содержат 75 мг FeSO4, то атрибут quantity должен иметь значение 75мг:0,6 мл. Как числитель, так и знаменатель должны быть количественными величинами (экстенсивными еличинами, например, число подсчетов, масса, объем, количество субстанции, количество энергии и т. д.).

A.3.3.5.13 Role.positionNumber:: LIST<INT> (0..*)

Определение. Целое число, указывающее положение исполнителя роли относительно сущности, контролирующей эту роль.

Обсуждение. Этот атрибут, в основном, используется в экземплярах класса Role, описывающих содержание в чем-либо. Например, у некоторых контейнеров есть дискретные положения, в которых может быть размещено содержимое. В зависимости от геометрии контейнера для указания конкретного положения может использоваться скалярное порядковое число или вектор порядковых чисел (координаты). Начальным значением координаты должно быть число 1.

Для некоторых контейнеров может использоваться собственный способ нумерации положений; способа нумерации может и не быть. В отсутствие какого-либо определенного способа позиционирования у конкретного типа контейнеров эмпирическое правило нумерации состоит в том, что координата, изменяющаяся раньше, должна позиционироваться первой. Для автоматизированного биохимического анализатора крови со штативом квадратной формы это означает, что первой координатой будет та, в каком направлении перемещается штатив на каждом шаге, а второй — та, в каком направлении штатив перемещается через каждые 10 (или иное число) шагов.

A.3.3.5.14 Переходы состояний экземпляра класса Role (атрибут состояния — statusCode)

Диаграмма перехода состояний класса Role показана на рисунке .8–.

Роль может иметь следующие состояния:

- active (активна) — подсостояние состояния normal: сущность в настоящий момент активна в данной роли;
- cancelled (отменена) — подсостояние состояния normal: роль была отменена до того, как стала активной;
- normal (нормальное): «типичное» состояние. Исключает состояние nullified, которое указывает, что экземпляр класса Role был создан по ошибке;
- nullified (аннулирована): это состояние является терминальным состоянием экземпляра класса Role, созданного по ошибке;
- pending (готовящаяся) — подсостояние состояния normal: это состояние отражает тот факт, что роль еще не стала активной;
- suspended (приостановлена) — подсостояние состояния normal: выполнение роли временно приостановлено. Переход в это состояние возможен из состояния active (активно);
- terminated (завершена) — подсостояние состояния normal: успешное завершение выполнения роли.

Между состояниями действия возможны следующие переходы:

- revise (пересмотреть) — из состояния active в состояние active;
- suspend (приостановить) — из состояния active в состояние suspended;
- terminate (завершить) — из состояния active в состояние terminated;
- nullify (аннулировать) — из состояния normal в состояние nullified;
- create (создать) — из начального (пустого) состояния в состояние active;
- create (создать) — из начального (пустого) состояния в состояние pending;
- revise (пересмотреть) — из состояния pending в состояние pending;
- resume (продолжить) — из состояния suspended в состояние active;
- revise (пересмотреть) — из состояния suspended в состояние suspended;
- terminate (завершить) — из состояния suspended в состояние terminated;
- reactivate (активизировать заново) — из состояния terminated в состояние active;
- revise (пересмотреть) — из состояния terminated в состояние terminated.

A.3.3.6 Класс RoleLink (в предметной области Roles)

Свойства класса RoleLink

Атрибуты класса RoleLink:

- typeCode:: CS,
- priorityNumber :: INT,
- effectiveTime:: IVL<TS>.

Ассоциации класса RoleLink:

- source::(1..1) Role::outboundLink::(0..*) (ассоциация с классом Role, роль outboundLink — исходящая связь),
- target::(1..1) Role::inboundLink::(0..*) (ассоциация с классом Role, роль inboundLink — входящая связь).

Определение класса RoleLink

Связь двух экземпляров класса Role, выражающая зависимость соответствующих ролей.

Примеры

1 Роль должности или агента зависит от другой роли, а именно, роли работника. Если роль работника завершается, то прекращается и роль должности. Тем самым определяется зависимость

роли должности от роли работника, или, другими словами, должность «является частью» роли работника.

2 Одна из ролей может иметь полномочия управления другой ролью (в иерархии подчинения). Например, работник категории «управляющий» может иметь полномочия управления служащими категории «аналитик», что может быть указано с помощью экземпляра класса RoleLink, у которого атрибут typeCode имеет значение «DIRAUTH» (has direct authority over — имеет прямые полномочия управления).

Обсуждение. Экземпляр класса RoleLink описывает отношения между ролями, а не между людьми (или другими сущностями). Люди (или другие сущности), в основном, связаны между собой прямыми отношениями исполнитель/контролер, возникающими у роли исполнителя, а в более общем плане — отношениями взаимодействия (т. е. их участием в действиях).

Атрибуты класса RoleLink

A.3.3.6.1 RoleLink.typeCode:: CS (1..1) Mandatory

Словарный домен: RoleLinkType

Определение. Код, указывающий вид связи, представленной данным экземпляром класса RoleLink, например, «PART» (has part — имеет частью), «DIRAUTH» (has direct authority over — имеет прямые полномочия управления).

A.3.3.6.2 RoleLink.priorityNumber:: INT (0..1)

Определение. Целое значение, указывающее относительный приоритет данной связи среди других связей похожих типов, у которых источником является один и тот же экземпляр класса Role. Связи с меньшими значениями атрибута priorityNumber рассматриваются раньше и выше тех, что имеют более высокие значения.

Примеры

1 Если имеет несколько резервных исполнителей, можно указать их очередность.

2 Предпочтительное место оказания медицинской помощи для врача, работающего в конкретной медицинской организации.

Обсуждение. Приоритеты могут быть полностью упорядоченными (все номера приоритета уникальны) и частично упорядоченными, когда одинаковый приоритет может быть присвоен нескольким связям между ролями.

A.3.3.6.3 RoleLink.effectiveTime:: IVL<TS> (0..1)

Определение. Временной интервал, указывающий период действия связи между ролями.

A.3.4 Классы предметной области CoreInfrastructure

A.3.4.1 Класс ActHeir (в предметной области CoreInfrastructure)

Класс ActHeir является специализацией класса Act.

Определение класса ActHeir

Этот класс определен исключительно для восполнения недостатка в текущих инструментальных средствах, не обеспечивающих возможность рефлексивного замыкания отношений генерализации (например, «действие является действием»).

Примеры — Рассмотрим уточненную информационную модель RMIM (refined message information model), в которую включен класс Act и его специализации Observation (исследование) и PatientEducationAct (санитарное просвещение). Последний является прямой специализацией («клоном») класса Act. В этом случае класс ActHeir используется как основа клона PatientEducationAct вместо самого класса Act из модели RIM. Класс Act используется здесь только как общая генерализация классов Observation и PatientEducationAct. Его использование диктуется исключительно (простительными) недостатками некоторых инструментов и структур данных, используемых в методологии HL7, и не несет никакой смысловой нагрузки.

Обсуждение. Примечательно, что класс ActHeir используется для представления действий, не описываемых другими подклассами в модели RIM, но при этом не отражает концептуальный принцип моделирования, согласно которому все генерализации должны быть абстрактными классами и только специализации-листья (наподобие класса ActHeir) могут иметь экземпляры. Однако класс ActHeir введен не по этой причине. Его роль состоит просто-напросто в решение практических проблем, вызванных применением определенных инструментов, и последующее развитие этих инструментов может позволить исключить такие классы в пользу эквивалентной абстракции, введенной в методологию.

Обратите внимание, что классы EntityHeir и RoleHeir используются аналогичным образом для классов Entity и Role соответственно.

A.3.4.2 Класс EntityHeir (в предметной области CoreInfrastructure)

Класс EntityHeir является специализацией класса Entity.

Определение класса EntityHeir

Этот класс определен исключительно для восполнения недостатка в текущих инструментальных средствах, не обеспечивающих возможность рефлексивного замыкания отношений генерализации (например, «сущность является сущностью»).

Обоснование. Использование этого класса диктуется исключительно (простительными) недостатками некоторых инструментов и структур данных, используемых в методологиии HL7, и не несет никакой смысловой нагрузки. Обратите внимание, что классы ActHeir и RoleHeir используются аналогичным образом для классов Act и Role соответственно.

Обоснование. Было обнаружено, что в иерархическом описании модели HMD нельзя создать структуру выбора choice для совокупности классов, если все они являются подтипами классов Act, Role или Entity, но при этом для них не определены отдельные физические классы. Другими словами, это классы, которые должны были бы быть определены в модели RIM как прямые потомки классов Act, Role или Entity, но при этом не имеют уникальных атрибутов или ассоциаций.

Добавление в каждую иерархию по одному такому пустому классу позволяет строить сообщения с подходящими требуемыми структурами выбора. Последующее развитие методологии и инструментальных средств может позволить исключить такие классы в пользу эквивалентной абстракции, введенной в методологию.

Примеры — Пусть уточненная информационная модель RMIM содержит класс Entity и его специализации EnvironmentalEntity (сущность окружающей среды) и LivingSubject (живой субъект). Класс EnvironmentalEntity является прямой специализацией («клоном») класса Entity. В этом случае класс EntityHeir используется как основа клона EnvironmentalEntity вместо самого класса Entity из модели RIM. Класс Entity используется здесь только как общая генерализация классов EnvironmentalEntity и LivingSubject.

Обсуждение. Примечательно, что класс EntityHeir используется для представления сущностей, не описываемых другими подклассами в модели RIM, но при этом не отражает концептуальный принцип моделирования, согласно которому все генерализации должны быть абстрактными классами и только специализации-листья (наподобие класса EntityHeir) могут иметь экземпляры. Однако класс EntityHeir введен не по этой причине. Его роль состоит просто-напросто в решение практических проблем, вызванных применением определенных инструментов, и последующее развитие этих инструментов может позволить исключить такие классы в пользу эквивалентной абстракции, введенной в методологию.

A.3.4.3 Класс InfrastructureRoot (в предметной области CoreInfrastructure)

Свойства класса InfrastructureRoot

Атрибуты класса InfrastructureRoot:

- nullFlavor :: CS,
- realmCode :: SET<CS>,
- typeId :: II,
- templateId :: LIST<II>.

Определение класса InfrastructureRoot

Этот абстрактный класс является генерализацией всех классов модели RIM, либо непосредственно, либо через наследование. В нем предусмотрен ряд атрибутов коммуникационной инфрастркутуры, которые могут использоваться в экземплярах коммуникаций, определенных комитетом HL7 на основе модели RIM. Будучи заданными в экземпляре коммуникации, эти атрибуты указывают, является ли структура информации ограниченной специально заданными шаблонами, областью применения или общими коммуникационными типами элементов.

В общем случае объявления ограничений наподобие тех, что могут наложены с помощью атрибутов данного класса, могут возникать при создании экземпляров класса модели RIM или одного из его производных клонов в целях осуществления коммуникации по протоколу HL7. Поэтому эти атрибуты должны присутствовать во всех классах модели RIM и их клонах.

Атрибуты класса InfrastructureRoot

A.3.4.3.1 InfrastructureRoot.nullFlavor :: CS (0..1)

Словарный домен: NullFlavor

Определение. Если в экземпляре класса этому атрибуту присвоено значение, то это служит признаком пустоты содержания класса. Остальная часть информации об этом классе и его свойствах не должна передаваться. Значение этого атрибута указывает конкретную причину отсутствия информации.

A.3.4.3.2 InfrastructureRoot.realmCode:: SET<CS> (0..*)

Словарный домен: Realm

Определение. Если в экземпляре класса этому атрибуту присвоено значение, то это служит признаком наличия ограничений, специфичных для области применения. Значение атрибута realmCode служит идентификатором этой области.

A.3.4.3.3 InfrastructureRoot.typeId :: II (0..1)

Определение. Если в экземпляре класса этому атрибуту присвоено значение, то это служит признаком наличия ограничений, определенных для специфичных сообщений стандарта HL7. Это могут быть ограничения общего типа (известные в коммуникационной среде обмена сообщениями как CMET (Common Message Elements Type — типы общих элементов сообщений) или содержание, включенное в определенную обертку. Значение атрибута typeId служит уникальным идентификатором этого типа.

A.3.4.3.4 InfrastructureRoot.templateId :: LIST<II> (0..*)

Определение. Если в экземпляре класса этому атрибуту присвоено значение, то это служит признаком наличия ограничений, определенных с помощью шаблона. Значение атрибута templateId служит уникальным идентификатором этого шаблона.

A.3.4.4 Класс RoleHeir (в предметной области CoreInfrastructure)

Класс RoleHeir является специализацией класса Role.

Определение класса RoleHeir

Этот класс определен исключительно для восполнения недостатка в текущих инструментальных средствах, не обеспечивающих возможность рефлексивного замыкания отношений генерализации (например, «роля является ролью»).

Примеры — Пусть уточненная информационная модель RMIM содержит класс Role и его специализации Employee (служащий) и Member (член). Класс Member является прямой специализацией («клоном») класса Role. В этом случае класс EntityHeir используется как основа клона Member вместо самого класса Entity из модели RIM. Класс Role используется здесь только как общая генерализация классов Employee и Member. Использование этого класса диктуется исключительно (простительными) недостатками некоторых инструментов и структур данных, используемых в методологии HL7, и не несет никакой смысловой нагрузки.

Обсуждение. Примечательно, что класс RoleHeir используется для представления ролей, не описываемых другими подклассами в модели RIM, но при этом не отражает концептуальный принцип моделирования, согласно которому все генерализации должны быть абстрактными классами и только специализации-листья (наподобие класса RoleHeir) могут иметь экземпляры. Однако класс RoleHeir введен не по этой причине. Его роль состоит просто-напросто в решение практических проблем, вызванных применением определенных инструментов, и последующее развитие этих инструментов может позволить исключить такие классы в пользу эквивалентной абстракции, введенной в методологию. Обратите внимание, что классы ActHeir и EntityHeir используются аналогичным образом для классов Act и Entity соответственно.

A.3.5 Классы предметной области MessageControl

A.3.5.1 Класс Acknowledgement (в предметной области MessageControl)

Свойства класса Acknowledgement

Атрибуты класса Acknowledgement:

- typeCode :: CS,
- expectedSequenceNumber :: INT,
- messageWaitingNumber :: INT,
- messageWaitingPriorityCode :: CE.

Ассоциации класса Acknowledgement:

- acknowledgementDetail::(0..*) AcknowledgementDetail::acknowledgement::(1..1) (ассоциация с классом AcknowledgementDetail, роль acknowledgement — подтверждение);
- acknowledges::(1..1) Transmission::acknowledgedBy::(0..*) (ассоциация с классом Transmission, роль acknowledgedBy — подтверждено приложением);
- conveyingTransmission::(1..1) Transmission::conveyedAcknowledgement::(0..*) (ассоциация с классом Transmission, роль conveyedAcknowledgement — переданное подтверждение).

Определение класса Acknowledgement

Класс Acknowledgement содержит информацию, передаваемую при подтверждении другого сообщения.

Атрибуты класса Acknowledgement

A.3.5.1.1 Acknowledgement.typeCode :: CS (0..1)

Словарный домен: AcknowledgementType

Определение. Этот атрибут содержит код подтверждения в соответствии с описанием, приведенным в правилах обработки сообщений HL7.

A.3.5.1.2 Acknowledgement.expectedSequenceNumber :: INT (0..1)

Определение. Этот атрибут используется в протоколе последовательной нумерации.

A.3.5.1.3 Acknowledgement.messageWaitingNumber :: INT (0..1)

Определение. Указывает число сообщений, которое подтверждающее приложение ожидает в очереди сообщений, предназначенных приложению-получателю.

Обсуждение. Эти сообщения должны быть вспоследствии получены с помощью запроса. Этот режим рассчитан на приложения-получатели, которые не могут получить прямое сообщение (то есть занимаются регулярным опросом).

Примеры — Если в очереди находятся 3 низкоприоритетных сообщений, 1 сообщение со средним приоритетом и 1 сообщений с высоким приоритетом, то число ожидаемых сообщений равно 5, то есть общему числу сообщений.

A .3.5.1.4 Acknowledgement.messageWaitingPriorityCode :: CE (0..1)

Словарный домен: MessageWaitingPriority

Определение. Указывает наивысший уровень важности в совокупности сообщений, которые подтверждающее приложение ожидает в очереди сообщений, предназначенных приложению-получателю.

Обсуждение. Эти сообщения должны быть вспоследствии получены с помощью запроса. Этот режим рассчитан на приложения-получатели, которые не могут получить прямое сообщение (то есть занимаются регулярным

опросом). Атрибут messageWaitingPriorityCode указывает, насколько важным является самое важное сообщение (и с его помощью можно управлять частотой опроса очереди приложением-получателем).

По местным соглашениям код важности может использоваться для определения интервала времени, в течение которого, как ожидается, приложение-получатель извлечет сообщения из очереди.

A.3.5.2 Класс AcknowledgementDetail (в предметной области MessageControl)

Свойства класса AcknowledgementDetail

Атрибуты класса AcknowledgementDetail:

- typeCode :: CS,
- code :: CE,
- text :: ED,
- location :: SET<ST>.

Ассоциации класса AcknowledgementDetail:

- acknowledgement::(1..1) Acknowledgement::acknowledgementDetail::(0..*) (ассоциация с классом Acknowledgement, роль acknowledgementDetail — детали подтверждения).

Определение класса AcknowledgementDetail

Сообщение, предоставляющее информацию о коммуникации, разборе или форматно-логическом контроле подтверждаемого сообщения.

Атрибуты класса AcknowledgementDetail

A.3.5.2.1 AcknowledgementDetail.typeCode :: CS (0..1)

Словарный домен: AcknowledgementDetailType

Определение. Идентифицирует вид информации, возвращаемой в подтверждающем сообщении. Возможны следующие варианты: «E» (ошибка), «W» (предупреждение) или «I» (информация).

A.3.5.2.2 AcknowledgementDetail.code :: CE (0..1)

Словарный домен: AcknowledgementDetailCode

Определение. Код, идентифицирующий возвращаемую информацию.

Обсуждение. Человеко-читаемое значение можно указать в компоненте текста, а при не кодированной информации — в компоненте исходного текста.

Примеры — «Обязательный атрибут xxx отсутствует», «Система будет недоступна 19 марта с 01:00 до 03:00».

A.3.5.2.3 AcknowledgementDetail.text :: ED (0..1)

Определение. Содержит дополнительную диагностическую информацию, релевантную возвращаемой информации.

Обсуждение. Диагностической информацией может быть свободный текст или структурированные данные (например, XML).

Примеры — Исключение в среде Java, дамп памяти, внутренний код ошибки, информация стека вызовов и т. д.

A.3.5.2.4 AcknowledgementDetail.location :: SET<ST> (0..*)

Определение. Указывает позицию в подтверждаемом сообщении, к которой относится возвращаемая информация.

Обсуждение. Не с любой возвращаемой информации можно связать соответствующую позицию. Некоторая возвращаемая информация мож относиться к нескольким позициям.

Пример — У отсутствующего элемента нет позиции, а в случае, когда два элемента нарушают условную обязательность, таких позиций может быть две.

A.3.5.3 Класс AttentionLine (в предметной области MessageControl)

Свойства класса AttentionLine

Атрибуты класса AttentionLine:

- keyWordText :: SC,
- value :: ANY.

Ассоциации класса AttentionLine:

- transmission::(1..1) Transmission::attentionLine::(0..*) (ассоциация с классом Transmission, роль attentionLine — идентификация адресата).

Определение класса AttentionLine

Экземпляры этого класса содержат параметры транспорта, специфичного для технологии реализации, предназначенные для передачи во внешней обертке сообщения стандарта V3.

Атрибуты класса AttentionLine

A.3.5.3.1 AttentionLine.keyWordText :: SC (0..1)

Словарный домен: AttentionKeyword

Определение. Слово, определяющее параметр.

A.3.5.3.2 AttentionLine.value :: ANY (0..1)

Документирование

Открытый вопрос: комитет Control-Query согласился предоставить новое определение, включающее в себя строгую совокупность ограничений типов данных, которые может принимать этот атрибут. Поскольку оно еще не устоялось, при гармонизации настоящего стандарта было решено не включать это определение.

A.3.5.4 Класс Batch (в предметной области MessageControl)

Свойства класса Batch

Атрибуты класса Batch:

- referenceControlId :: II,
- name :: SC,
- batchComment :: SET<ST>,
- transmissionQuantity :: INT,
- batchTotalNumber :: SET<INT>.

Ассоциации класса Batch:

- transmission::(0..*) Transmission::batch::(0..1) (ассоциация с классом Transmission, роль batch — пакет).

Класс Batch является специализацией класса Transmission.

Определение класса Batch

Класс Batch используется для задания сообщения, являющегося коллекцией сообщений стандарта HL7 V3.

Атрибуты класса Batch

A.3.5.4.1 Batch.referenceControlId :: II (0..1)

Определение. Этот атритбут содержит управляющий идентификатор пакета, присвоенный ему при первоначальной передаче.

A.3.5.4.2 Batch.name :: SC (0..1)

Словарный домен: BatchName

Определение. Этот атрибут используется приложением, обрабатывающим пакет.

A.3.5.4.3 Batch.batchComment :: SET<ST> (0..*)

Определение. В этом атрибуте могут передваться комментарии к содержанию пакета.

A.3.5.4.4 Batch.transmissionQuantity :: INT (0..1)

Определение. Этот атрибут содержит общее число отдельных сообщений, содержащихся в пакете.

A.3.5.4.5 Batch.batchTotalNumber :: SET<INT> (0..*)

Определение. Контрольная сумма пакета. Можно указать несколько контрольных сумм.

A.3.5.5 Класс CommunicationFunction (в предметной области MessageControl)

Свойства класса CommunicationFunction

Атрибуты класса CommunicationFunction:

- typeCode :: CS,
- telecom :: TEL.

Ассоциации класса CommunicationFunction:

- entity::(1..*) Entity::communicationFunction::(0..*) (ассоциация с классом Entity, роль communicationFunction — коммуникационная функция),
- transmission::(1..*) Transmission::communicationFunction::(0..*) (ассоциация с классом Transmission, роль communicationFunction — коммуникационная функция).

Определение класса CommunicationFunction

Этот класс связывает с передачей сообщений различные задействованные в ней сущности (отправитель, получатель, реагирующее приложение).

Атрибуты класса CommunicationFunction

A.3.5.5.1 CommunicationFunction.typeCode :: CS (0..1)

Словарный домен: CommunicationFunctionType

Определение. Тип коммуникационной функции, выполняемой сущностью по отношению к передаче, например, отправитель, получатель, реагирующее приложение и т. д.

A.3.5.5.2 CommunicationFunction.telecom :: TEL (0..1)

Определение. Телекоммуникационный адрес, который может использоваться для контакта с сущностью, выполняющую данную функцию.

A.3.5.6 Класс Message (в предметной области MessageControl)

Свойства класса Message

Атрибуты класса Message:

- versionCode :: CS,
- interactionId :: II,
- profileId :: LIST<II>,
- processingCode :: CS,
- processingModeCode :: CS,
- acceptAckCode :: CS,
- responseCode :: CS,

- sequenceNumber :: INT,
- attachmentText :: SET<ED>.

Ассоциации класса Message:

- controlAct::(0..1) ControlAct::payload::(0..*) (ассоциация с классом ControlAct, роль payload — полезная нагрузка).

Класс Message является специализацией класса Transmission.

Определение класса Message

Класс Message является родительским классом для всех сообщений стандарта HL7 Версии 3.

Атрибуты класса Message

A.3.5.6.1 Message.versionCode :: CS (0..1)

Словарный домен: HL7StandardVersionCode

Определение. Значение атрибута сравнивается системой получателем с собственной версией, чтобы убедиться в правильности интерпретации сообщения.

A.3.5.6.2 Message.interactionId :: II (0..1)

Определение. Содержит идентификатор взаимодействия, используемый для ссылки на конкретный обмен информацией. Определен с помощью методологии V3 MDF как часть сообщения.

A.3.5.6.3 Message.profileId :: LIST<II> (0..*)

Определение. Содержит идентификатор профиля, позволяющий явно обозначить отличие данной реализации от стандартного определения сообщения.

Если указано несколько профилей, то экземпляр сообщения должен соответствовать каждому из них. Однако получатель может выбрать проверку соответствия какому-то одному из этих профилей. Поэтому «предпочтительный» или более строгий профиль целесообразно указывать первым в списке.

A.3.5.6.4 Message.processingCode :: CS (0..1)

Словарный домен: ProcessingMode

Определение. Указывает, предназначено ли сообщение для производственной обработки, обучения или отладки.

A.3.5.6.5 Message.processingModeCode :: CS (0..1)

Словарный домен: ProcessingMode

Определение. Указывает, передается ли сообщение для текущей обработки, архивирования, начальной загрузки, восстановления из архива и т. д.

A.3.5.6.6 Message.acceptAckCode :: CS (0..1)

Словарный домен: AcknowledgementCondition

Определение. Этот атрибут идентифицирует условия, при которых в ответ на данное сообщение должно быть возвращено подтверждение приема.

A.3.5.6.7 Message.responseCode :: CS (0..1)

Словарный домен: ResponseLevel

Определение. Указывает, ожидается ли от адресата данного взаимодействия информация о прикладной обработке и насколько детальной она должна быть. Этот атрибут ограничивает варианты ответа по умолчанию, определяющие ответственность получателя за взаимодействие. Например, если в некотором взаимодействии получатель отвечает за посылку либо подтверждения приемлемости, либо отказа, а атрибут responseCode имеет значение «E» (exception — исключение), то получатель должен ответить на данное сообщение только в том случае, если он откажет в его обработке.

A.3.5.6.8 Message.sequenceNumber :: INT (0..1)

Определение. Этот атрибут предназначен для реализации протокола последовательной нумерации. Его значение увеличивается на 1 при каждом следующем присваивании.

A.3.5.6.9 Message.attachmentText :: SET<ED> (0..*)

Определение. Содержит произвольные вложения блоков данных, на которые могут иметься ссылки из внутреннего содержанмия сообщения. В любой спецификации технологии реализации рекомендуется выносить вложения за пределы основного тела сообщения. Для ссылок на вложения из тела сообщения используется ссылочная функциональность, предусмотренная в типе данных ED.

A.3.5.7 Класс Transmission (в предметной области MessageControl)

Свойства класса Transmission

Атрибуты класса Transmission:

- id :: II,
- creationTime :: TS,
- securityText :: ST.

Ассоциации класса Transmission:

- acknowledgedBy::(0..*) Acknowledgement::acknowledges::(1..1) (ассоциация с классом Acknowledgement, роль acknowledges — подтверждение),
- conveyedAcknowledgement::(0..*) Acknowledgement::conveyingTransmission::(1..1) (ассоциация с классом Acknowledgement, роль conveyingTransmission — передача данных),
- attentionLine::(0..*) AttentionLine::transmission::(1..1) (ассоциация с классом AttentionLine, роль transmission — передача),

- batch::(0..1) Batch::transmission::(0..*) (ассоциация с классом Batch, роль transmission — передача),
- communicationFunction::(0..*) CommunicationFunction::transmission::(1..*) (ассоциация с классом CommunicationFunction, роль transmission — передача).

Класс Transmission является генерализацией следующих классов:

- Batch,
- Message.

Определение класса Transmission

Представляет информацию о конкретном акте передачи информации от одного приложения к другому.

Атрибуты класса Transmission

A.3.5.7.1 Transmission.id :: II (0..1)

Определение. Уникальный идентификатор акта передачи информации.

A.3.5.7.2 Transmission.creationTime :: TS (0..1)

Определение. Дата и время акта передачи информации ее отправителем. Если указан часовой пояс, то по умолчанию он будет использоваться для всех передаваемых данных.

A.3.5.7.3 Transmission.securityText :: ST (0..1)

Определение. Этот атрибут используется приложениями для реализации функций безопасности акта передачи информации. В настоящее время его более детальное описание не предусмотрено.

A.3.6 Классы предметной области QueryControl

A.3.6.1 Класс LogicalExpression (в предметной области QueryControl)

Свойства класса LogicalExpression

Атрибуты класса LogicalExpression:

- relationalConjunctionCode :: CS.

Ассоциации класса LogicalExpression:

- userAsLeft::(0..*) SelectionExpression::leftSide::(0..1) (ассоциация с классом SelectionExpression, роль leftSide — левая часть),
- userAsRight::(0..*) SelectionExpression::rightSide::(0..1) (ассоциация с классом SelectionExpression, роль rightSide — правая часть).

Класс LogicalExpression является специализацией класса SelectionExpression.

Атрибуты класса LogicalExpression

A.3.6.1.1 LogicalExpression.relationalConjunctionCode :: CS (0..1)

Словарный домен: SQLConjunction

Определение. Если в условии поиска экземпляров задано более одного критерия, то с помощью данного атрибута можно указать логическую операцию связывания дополнительного критерия.

A.3.6.2 Класс Parameter (абстрактный) (в предметной области QueryControl)

Свойства класса Parameter

Атрибуты класса Parameter:

- id :: II.

Ассоциации класса Parameter:

- queryByParameter::(0..1) QueryByParameter::parameter::(0..*) (ассоциация с классом QueryByParameter, роль parameter — параметр),
- parameterList::(0..1) ParameterList::parameter::(0..*) (ассоциация с классом ParameterList, роль parameter — параметр).

Класс Parameter является генерализацией следующих классов:

- ParameterItem,
- ParameterList.

Определение класса Parameter

Класс Parameter предназначен для реализации стандарта и используетс ядля представления структуры параметров, которые могут быть указаны в запросе с параметрами, предусмотренном в спецификации запросов стандарта V3. Параметры могут передаваться как пары «имя/значение», как именованный список параметров, в котором указано множество пар «имя/значение», или как сочетание обоих этих вариантов.

Атрибуты класса Parameter

A.3.6.2.1 Parameter.id :: II (0..1)

Атрибут Parameter.id может использоваться для трассировки проблем при реализации запросов с параметрами.

A.3.6.3 Класс ParameterItem (в предметной области QueryControl)

Свойства класса ParameterItem

Атрибуты класса ParameterItem:

- value :: ANY,
- semanticsText :: ST.

Класс ParameterItem является специализацией класса Parameter.

Определение класса ParameterItem

Представляет структуру именованного значения (пара «имя-значение»), используемую для элементов, указанных в ответе на запрос.

Атрибуты класса ParameterItem

A.3.6.3.1 ParameterItem.value :: ANY (0..1)

Определение. Содержит значение, передаваемое в структуре именованного значения (пара «имя-значение»), используемой для элементов, указанных в ответе на запрос.

A.3.6.3.2 ParameterItem.semanticsText :: ST (0..1)

Определение. Содержит уникальный идентификатор (имя), передаваемый в структуре именованного значения (пара «имя-значение»), используемой для элементов, указанных в ответе на запрос.

A.3.6.4 Класс ParameterList (в предметной области QueryControl)

Свойства класса ParameterList

Ассоциации класса ParameterList:

- parameter::(0..*) Parameter::parameterList::(0..1) (ассоциация с классом Parameter, роль parameterList — список параметров).

Класс ParameterList является специализацией класса Parameter.

Определение класса ParameterList

Задает именованный список параметров (пар «имя/значение»), который приводится в объявлении соответствия запроса.

A.3.6.5 Класс QueryAck (в предметной области QueryControl)

Свойства класса QueryAck

Атрибуты класса QueryAck:

- queryResponseCode :: CS,
- resultTotalQuantity :: INT,
- resultCurrentQuantity :: INT,
- resultRemainingQuantity :: INT.

Класс QueryAck является специализацией класса QueryEvent.

Определение класса QueryAck

В экземпляре этого класса содержится информация, передаваемая в ответах на запрос.

Атрибуты класса QueryAck

A.3.6.5.1 QueryAck.queryResponseCode :: CS (0..1)

Словарный домен: QueryResponse

Определение. В этом атрибуте реагирующая система возвращает точную информацию о статусе ответа на запрос.

A.3.6.5.2 QueryAck.resultTotalQuantity :: INT (0..1)

Определение. Указывает общее число совпадений с условием запроса, на который дается ответ.

A.3.6.5.3 QueryAck.resultCurrentQuantity:: INT (0..1)

Определение. Указывает число совпадений с условием запроса, информация о которых передается в текущей порции ответа.

A.3.6.5.4 QueryAck.resultRemainingQuantity:: INT (0..1)

Определение. Указывает число оставшихся совпадений с условием запроса, информация о которых все еще должна быть передана получателю.

A.3.6.6 Класс QueryByParameter (в предметной области QueryControl)

Ассоциации класса QueryByParameter:

- parameter::(0..*) Parameter::queryByParameter::(0..1) (ассоциация с классом Parameter, роль queryByParameter — запрос с параметрами).

Класс QueryByParameter является специализацией класса QuerySpec.

Определение класса QueryByParameter

В экземпляре этого класса передается определение запроса с параметрами в формате, предложенном комитетом HL7 для замены транзакции QRD/QRF. Этот формат считается замкнутым, поскольку спецификация сервера данных содержит фиксированный список параметров, опубликованный в объявлении соответствия запроса.

A.3.6.7 Класс QueryBySelection (в предметной области QueryControl)

Ассоциации класса QueryBySelection:

- selectionExpression::(0..*) SelectionExpression::queryBySelection::(1..1) (ассоциация с классом Selection Expression, роль queryBySelection — селективный запрос).

Класс QueryBySelection является специализацией класса QuerySpec.

Определение класса QueryBySelection

В экземпляре этого класса передается определение селективного запроса. Это один из вариантов запросов, определенных в стандарте HL7. В условии этого запроса можно указывать любое число переменных из числа тех, что предусмотрены сервером данных, а также любые допустимые операторы и значения для каждой переменной, отвечающие опубликованному объявлению о соответствии запроса. Этот формат запроса является открытым, поскольку в нем допускается спецификация селекции, основанная на опубликованной схеме базы данных.

Класс QueryContinuation (в предметной области QueryControl)

Свойства класса QueryContinuation

Атрибуты класса QueryContinuation:

- startResultNumber :: INT,
- continuationQuantity :: INT.

A.3.6.8 Класс QueryContinuation является специализацией класса QueryEvent.

Определение класса QueryContinuation

В экземпляре этого класса передается информация о состоянии обмена, требуемая на прикладном уровне для обеспечения логического продолжения ответа на запрос.

Атрибуты класса QueryContinuation

A.3.6.8.1 QueryContinuation.startResultNumber :: INT (0..1)

Определение. Указывает номер экземпляра в серии результатов исходного запроса, с которого начнется следующее сообщение ответа на запрос.

A.3.6.8.2 QueryContinuation.continuationQuantity :: INT (0..1)

Определение. Указывает число совпадений с экземплярами, которые будут возвращены в следующем сообщении ответа на запрос.

A.3.6.9 Класс QueryEvent (в предметной области QueryControl)

Свойства класса QueryEvent

Атрибуты класса QueryEvent:

- queryId :: II,
- statusCode :: CS.

Ассоциации класса QueryEvent:

- controlAct::(1..1) ControlAct::queryEvent::(0..1) (ассоциация с классом ControlAct, роль queryEvent — событие запроса).

Определение класса QueryEvent:

Этот абстрактный класс используется для сборки частей обмена сообщениями, специфичных для сообщений запроса и ответа на запрос.

Обоснование. Техническим комитетом определен тип элементов сообщения, удовлетворяющий требованиям уровня сообщений к ответу на запрос (наподобие типа элементов сообщений, возвращаемого на запрос демографических данных). В пересмотренной модели запроса/ответа, предложенной в стандарте V3, экземпляр такого типа элементов сообщения может быть представлен как результат ответа на запрос. «Группа_возвращаемых_элементов» идентифицировала бы представление модели RIM, которое по форме аналогично представлению модели RIM, описывающему повествовательный или повелительный обмен сообщениями между приложениями.

Атрибуты класса QueryEvent

A.3.6.9.1 QueryEvent.queryId :: II (0..1)

Определение. Значение этого атрибута может быть присвоено инициирующим приложением в целях идентификации запроса. Оно предназначено для привязки ответных сообщений к исходному запросу. Идентификатор QueryEvent.queryId может оставаться одним и тем же в серии нескольких обменов сообщениями, осуществляемых при продолжении предшествующего им запроса.

A.3.6.9.2 QueryEvent.statusCode :: CS (0..1)

Словарный домен: QueryStatusCode

A.3.6.9.3 Переходы состояний класса QueryEvent (атрибутом состояния является statusCode)

Диаграмма перехода состояний класса QueryEvent показана на рисунке .14–.

Состояния класса QueryEvent:

- aborted (прекращен),
- deliveredResponse (ответ доставлен),
- executing (на выполнении),
- new (новый),

waitContinuedQueryResponse (ожидает продолжения ответа на запрос).

Переходы состояний класса QueryEvent:

- abort (прекратить) — (из состояния deliveredResponse в состояние aborted);
- activateQueryContinue (активировать продолжение запроса) — (из состояния deliveredResponse в состояние executing);
- abort (прекратить) — (из состояния executing в состояние aborted);
- completeInitialQueryResponse (завершить ответ на исходный запрос) — (из состояния executing в состояние deliveredResponse);
- completeQueryContinuation (завершить продолжение запроса) — (из состояния executing в состояние deliveredResponse);
- executeQuerySpec (выполнить спецификацию запроса) — (из состояния new в состояние executing)
- create (создать) — (из состояния null в состояние new).

A.3.6.10 Класс QuerySpec (в предметной области QueryControl)

Свойства класса QuerySpec

Атрибуты класса QuerySpec:

- modifyCode :: CS,
- responseElementGroupId :: SET<II>,

- responseModalityCode :: CS,
- responsePriorityCode :: CS,
- initialQuantity :: INT,
- initialQuantityCode :: CE,
- executionAndDeliveryTime :: TS.

Ассоциации класса QuerySpec:

- sortControl::(0..*) SortControl::querySpec::(1..1) (ассоциация с классом SortControl, роль querySpec — спецификация запроса).

Класс QuerySpec является детализацией класса QueryEvent.

Класс QuerySpec имеет следующие специализации:

- QueryByParameter,
- QueryBySelection.

Определение класса QuerySpec

Этот класс содержит спецификации всех типов запросов, предусмотренных в стандарте HL7 Версии 3. Его определение содержит атрибуты, общие для всех типов запросов.

Атрибуты класса QuerySpec

A.3.6.10.1 QuerySpec.modifyCode :: CS (0..1)

Словарный домен: ModifyIndicator

Определение. Указывает, является ли подписка на результаты запроса новой или модифицированной.

A.3.6.10.2 QuerySpec.responseElementGroupId :: SET<II> (0..*)

Определение. В атрибуте responseElementGroupId указан специфичный тип сообщения, возвращаемого в ответ на запрос. Он должен выбираться из числа типов, которые получатель обязан воспринимать в качестве ответов на запрос.

A.3.6.10.3 QuerySpec.responseModalityCode :: CS (0..1)

Словарный домен: ResponseModality

Определение. Задает оперативность и группировку экземпляров ответных сообщений.

A.3.6.10.4 QuerySpec.responsePriorityCode :: CS (0..1)

Словарный домен: QueryPriority

Определение. Указывает, является ли запрос синхронным или асинхронным.

A.3.6.10.5 QuerySpec.initialQuantity :: INT (0..1)

Определение. Указывает максимальный размер ответа, который может быть принят запрашивающим приложением.

A.3.6.10.6 QuerySpec.initialQuantityCode :: CE (0..1)

Словарный домен: QueryRequestLimit

Определение. Задает единицы измерения максимальной длины ответа на запрос, которая может быть принята запрашивающим приложением.

A.3.6.10.7 QuerySpec.executionAndDeliveryTime :: TS (0..1)

Определение. Задает время, в течение которого должен быть возвращен ответ.

A.3.6.11 Класс RelationalExpression (в предметной области QueryControl)

Свойства класса RelationalExpression

Атрибуты класса RelationalExpression:

- elementName :: SC,
- relationalOperatorCode :: CS,
- value :: ST.

Класс RelationalExpression является специализацией класса SelectionExpression.

Атрибуты класса RelationalExpression

A.3.6.11.1 RelationalExpression.elementName :: SC (0..1)

Словарный домен: RelationalName

Определение. Задает элемент модели RIM в качестве операнда критерия выборки.

A.3.6.11.2 RelationalExpression.relationalOperatorCode :: CS (0..1)

Словарный домен: RelationalOperator

Определение. Задает оператор отношения, используемого в критерии выборки.

A.3.6.11.3 RelationalExpression.value :: ST (0..1)

Определение. Значение, с которым сравнивается операнд критерия выборки.

A.3.6.12 Класс SelectionExpression (абстрактный) (в предметной области QueryControl)

Ассоциации класса SelectionExpression:

- leftSide::(0..1) LogicalExpression::userAsLeft::(0..*) (ассоциация с классом LogicalExpression, роль userAsLeft — используется как левая часть),
- queryBySelection::(1..1) QueryBySelection::selectionExpression::(0..*) (ассоциация с классом QueryBySelection, роль selectionExpression — выражение выборки),
- rightSide::(0..1) LogicalExpression::userAsRight::(0..*) (ассоциация с классом LogicalExpression, роль userAsRight — используется как правая часть).

Класс SelectionExpression имеет следующие специализации:
- LogicalExpression,
- RelationalExpression.

A.3.6.13 Класс SortControl (в предметной области QueryControl)

Свойства класса SortControl

Атрибуты класса SortControl:
- sequenceNumber :: INT,
- elementName :: SC,
- directionCode :: CS.

Ассоциации класса SortControl:
- querySpec::(1..1) QuerySpec::sortControl::(0..*) (ассоциация с классом QuerySpec, роль sortControl — управление сортировкой).

Определение класса SortControl

Спецификация порядка сортировки экземпляров, удовлетворяющих условию запроса.

Атрибуты класса SortControl

A.3.6.13.1 SortControl.sequenceNumber :: INT (0..1)

Определение. Задает порядок или очередность экземпляра класса SortControl в данном запросе.

A.3.6.13.2 SortControl.elementName :: SC (0..1)

Словарный домен: ElementName

Определение. Задает элемент модели RIM, по которому должны сортироваться ответы на запрос.

A.3.6.13.3 SortControl.directionCode :: CS (0..1)

Словарный домен: Sequencing

Определение. Задает направление сортировки (по возрастанию или по убыванию).

A.3.7 Классы предметной области StructuredDocuments

A.3.7.1 Класс ContextStructure (в предметной области StructuredDocuments)

Свойства класса ContextStructure

Атрибуты класса ContextStructure:
- setId :: II,
- versionNumber :: INT.

Класс ContextStructure является специализацией класса Act.

Класс ContextStructure имеет следующую специализацию:

Document.

Определение класса ContextStructure

Структура, представляющая собой контейнер внутри документа. Структуры имеют заголовки, которые могут кодироваться. Структуры могут быть вложенными и могут содержать подразделы.

Атрибуты класса ContextStructure

A.3.7.1.1 ContextStructure.setId :: II (0..1)

Определение. Идентификатор отчета, остающийся неизменных во всех версиях, произведенных от общего оригинала.

Исходный отчет является первой версией. Его атрибуту setId присваивается новое уникальное значение, а атрибуту versionNumber присваивается значение «1».

Дополнение является приложением к существующему отчету, содержащим лополнительную информацию. Приложение само по себе является исходным отчетом. Родительским отчет, к которому оно прилагается, связывается с им с помощью экземпляра класса ActRelationship, у которого атрибут ActRelationship.typeCode имеет значение «APND» (appends — дополняет). Дополняемый родительский отчет остается на своем месте, его содержание и состояние не изменяются.

Заменяющий отчет заменяет существующий отчет. Атрибут setId у заменяющего отчета имеет то же значение, что у родительского, а значение атрибута номера версии (versionNumber) увеличивается на 1. Состояние замененного родительского отчета изменяется на «superceded» (пересмотрен), но сам этот отчет должен остаться в системе для исторических ссылок.

A.3.7.1.2 ContextStructure.versionNumber :: INT (0..1)

Определение. Номер версии представляет собой целое число. Номера версий начинаются с 1 и увеличиваются на 1. Номер версии исходного отчета всегда должен равняться 1. Номер версии должен увеличиваться при замене отчета, но по местным требованиям его увеличение может происходить чаще.

A.3.7.2 Класс Document (в предметной области StructuredDocuments)

Свойства класса Document

Атрибуты класса Document:
- completionCode :: CE,
- storageCode :: CE,
- copyTime :: TS,
- bibliographicDesignationText :: SET<ED>.

Класс Document является специализацией класса ContextStructure.

Определение класса Document

Специализация класса Act, содержащая характеристики, уникальные для служб управления документами.

Атрибуты класса Document

A.3.7.2.1 Document.completionCode :: CE (0..1)

Словарный домен: DocumentCompletion

Определение. Код, указывающий состояние завершенности отчета (например, незавершен, заверен, юридически заверен).

A.3.7.2.2 Document.storageCode :: CE (0..1)

Словарный домен: DocumentStorage

Определение. Код, указывающий статус хранения отчета (например, активный, архивирован, удален).

A.3.7.2.3 Document.copyTime :: TS (0..1)

Определение. Дата и время предоставления документа (например, копирования или отправки на устройство визуализации) системой управления документами, осуществляющей контроль версий документа. Будучи присвоенным, это значение не может быть изменено. Этот атрибут был введен, чтобы читатель документа имел некоторое представление о том, как долго документ находился вне безопасного контекста его системы управления документами.

A.3.7.2.4 Document.bibliographicDesignationText :: SET<ED> (0..*)

Определение. Библиографическая ссылка на документ, позволяющая идентифицировать его, найти и/или извлечь из общих собраний.

A.4 Ассоциации

A.4.1 (1..1) Acknowledgement:: acknowledgementDetail :: (0..*) AcknowledgementDetail:: acknowledgement

Идентифицирует отношение между подтверждением и детальной информацией об ошибке, предупреждении или информации, передаваемой в этом подтверждении.

A.4.2 (0..*) ActRelationship:: source :: (1..1) Act:: outboundRelationship

A.4.3 (0..*) ActRelationship:: target :: (1..1) Act:: inboundRelationship

A.4.3 (0..*) Participation:: act :: (1..1) Act:: participation

A.4.5 (1..*) Entity:: communicationFunction :: (0..*) CommunicationFunction:: entity

Это отношение позволяет идентифицировать сущности, выполняющие различные коммуникационные функции.

A.4.6 (1..*) Transmission:: communicationFunction :: (0..*) CommunicationFunction:: transmission

Это отношение связывает акт передачи информации с ее отправителем, получателем, стороной обратного вызова и т. д.

A.4.7 (0..1) QueryEvent:: controlAct :: (1..1) ControlAct:: queryEvent

A.4.8 (0..*) Role:: player :: (0..1) Entity:: playedRole

На эту ассоциацию накладывается следующее ограничение:

Invariant (Role x) { not(x.player.equals(null)) or not(x.scoper.equals(null)) }

A.4.9 (0..*) Role:: scoper :: (0..1) Entity:: scopedRole

На эту ассоциацию накладывается следующее ограничение:

Invariant (Role x) { not(x.player.equals(null)) or not(x.scoper.equals(null)) }

A.4.10 (1..1) Entity:: languageCommunication :: (0..*) LanguageCommunication:: entity

A.4.11 (0..*) SelectionExpression:: leftSide :: (0..1) LogicalExpression:: userAsLeft

A.4.12 (0..*) SelectionExpression:: rightSide :: (0..1) LogicalExpression:: userAsRight

A.4.13 (0..1) ControlAct:: payload :: (0..*) Message:: controlAct

A.4.14 (0..1) ParameterList:: parameter :: (0..*) Parameter:: parameterList

Задает отношение между списком параметров и входящими в него параметрами.

A.4.15 (0..*) Parameter:: queryByParameter :: (0..1) QueryByParameter:: parameter

A.4.16 (0..*) SelectionExpression:: queryBySelection :: (1..1) QueryBySelection:: selectionExpression

A.4.17 (0..*) SortControl:: querySpec :: (1..1) QuerySpec:: sortControl

A.4.18 (0..*) Participation:: role :: (1..1) Role:: participation

A.4.19 (0..*) RoleLink:: source :: (1..1) Role:: outboundLink

A.4.20 (0..*) RoleLink:: target :: (1..1) Role:: inboundLink

A.4.21 (0..*) Acknowledgement:: acknowledges :: (1..1) Transmission:: acknowledgedBy

Идентифицирует отношение между актом передачи информации и подтверждениями этого акта.

A.4.22 (0..*) Acknowledgement:: conveyingTransmission :: (1..1) Transmission:: conveyedAcknowledgement

Идентифицирует отношение между подтверждением и актом его передачи.

A.4.23 (0..*) AttentionLine:: transmission :: (1..1) Transmission:: attentionLine

Это отношение позволяет представить параметры транспорта, специфичного для технологии реализации, во внешней обертке сообщения стандарта V3.

A.4.24 (0..1) Batch:: transmission :: (0..*) Transmission:: batch

А.5 Обзор модели RIM

А.5.1 Цель

Этот документ представляет собой краткий обзор базовых понятий и логических обоснований, положенных в основу разработки и развития модели HL7 RIM. Он не является руководством по применению модели RIM, хотя в нем и представлено несколько частных примеров ее использования, полезных для объяснения фундаментальных понятий разработки и развития модели RIM.

А.5.2 Общие сведения

Модель RIM строится в целях согласованного совместного использования данных во многих «местных» контекстах. Вообще говоря, чем шире сфера интересов, тем важнее сделать явными все предположения, полагаемые в основу темы или области интереса. Модель RIM HL7 Версии 3 спроектирована как единая база и развитый источник всей информации, используемой в спецификации HL7. RIM конкретно и однозначно формулирует точные определения понятий здравоохранения («предметы интереса»), используемые в сфере информационных систем здравоохранения, а также описывает отношения (они же «ассоциации») между этими понятиями.

Спецификации стандарта HL7 Версии 3 (например, сообщения HL7 Версии 3, структурированные документы и т. д.) позволяют взаимодействовать (например, обмениваться данными) слабо связанным информационным системам, используемым при разных условиях оказания медицинской помощи, предоставляемой разными поставщиками и на разных территориях. Поэтому область применения модели RIM HL7 включает в себя всю информацию, которой должны обмениваться участвующие информационные системы здравоохранения в целях ее обработки. Кроме того, следует отметить, что она не моделирует (и не должна моделировать) информацию, хранящуюся в конкретной информационной системе здравоохранения, но никогда не передаваемую другим системам.

Модель RIM представлена с помощью синтаксиса визуального моделирования, основанного на Унифицированном языке моделирования UML (Unified Modeling Language). (Определенные различия между «стандартным UML» и «UML HL7», использованным в модели RIM, например, размещение названий ассоциаций, в настоящее время рассматриваются техническим комитетом по моделированию и методологии HL7 (Modeling and Methodology Technical Committee) в целях движения к максимальному согласованию обоих синтаксисов.) Комитет HL7 также ведет базу данных («хранилище модели RIM»), содержащую детальную информацию о каждом понятии модели, каждом атрибуте и каждой ассоциации, включая логическое обоснование элемента, определение, ограничения и историю редактирования/изменений. В перспективе эта информация может быть формально опубликована (частично или целиком) в виде профиля UML, специфичного для стандарта HL7.

Важно отметить, что модель RIM создается не для того, чтобы стать логической или физической моделью базы данных или иной структуры информационной системы конкретного производителя либо обслуживать интересы конкретной организации здравоохранения или сети организаций. В действительности модель RIM даже не представляет конкретные группы сообщений HL7, она является обобщенной совокупностью данных и отношений между данными, из которых может быть сконструировано любое релевантное сообщение HL7. Ожидается, что по мере необходимости конкретные пользователи модели RIM используют ее релевантные части, принимая во внимание их содержание при разработке и описании своих информационных моделей.

А.5.3 Обоснование конструкции модели RIM

Разветвленная структура RIM основана на шести «базовых» классах: Act (действие), Entity (сущность), Role (роль), Participation (участие), ActRelationship (связь действий) и RoleLink (связь ролей). Определения этих классов приведены в разделе А.1.4. Далее приводится обсуждение фундаментальных принципов, положенных в основу этих шести классов и их связей, а также базовые рекомендации по их использованию.

В модели HL7 RIM выделены два основных «высокоуровневых» понятия, являющихся фундаментальными для понимания мира информации здравоохранения: намеренные «действия» или «услуги» (экземпляры класса Act), и «люди, места и предметы», представляющие интерес в мире здравоохранения (экземпляры класса Entity).

Класс Act (и его подклассы) представляет все намеренные действия, документируемые медицинским работником в клиническом или административном контексте. Появление класса Act в качестве одного из центральных классов модели RIM отражает следующую позицию комитета HL7: с точки зрения информационного взаимодействия и обмена сообщениями оказание медицинской помощи состоит из последовательности атрибутированных намеренных действий. Таким образом, экземпляры класса Act описывают и клинические исследования (например, температуру пациента), и вмешательства (например, применение лекарств), и административные действия (например, госпитализацию пациента). Учтите, что с такой «действиецентричной» точки зрения на оказание медицинской помощи действие исследования имеет два очевидно противоположных значения: «действие установления и документирования конкретного факта» и «описание исследуемого предмета». Другими словами, экземпляр класса Observation, являющегося специализацией класса Act, описывает как атрибутированное действие исследования, так и результаты исследования. Оба аспекта этого расширенного определения класса Observation отражены в специфических атрибутах класса Act или его подкласса Observation.

Класс Entity (и его подклассы) представляет все живые субъекты (например, людей, животных), организации (как формальные, так и неформальные), материалы (например, прочные и непрочные предметы, продукты, биоматериал, контейнеры) и места, которые могут быть представлять интерес в передаваемом контексте оказания медицинской помощи. Необходимо отметить, что понятие «коллекции информации» (например, медицинская карта), представляется не в виде экземпляра класса Entity или его подклассов, а моделируется как коллекция атрибутированных действий, описываемых экземплярами класса Act.

Между классами Act и Entity модель RIM помещает два дополнительных класса — Role (роль) и Participation (участие). Класс Role моделирует несколько важных понятий, превалирующих в предметной области оказания медицинской помощи. Во-первых, класс Role отражает тот факт, что в конкретном контексте медицинской помощи различные «статические» объекты могут «временно» выполнять одну или несколько «ролей» (например, пациент, врач общей практики, ответственная сторона, медицинская сестра и т. д.). Во-вторых, понятия «способности» (например, возможность оказания медицинской помощи в соответствии с рекомендациями Advanced Cardiac Life Support) и «сертификации» (например, диплом медицинской сестры) также могут моделироваться с помощью экземпляров класса Role. Наконец, тщательное изучение кратности (0..1) и имен (контролер, исполнитель) двух ассоциаций между классами Entity и Role показывает, что класс Role может использоваться для «группировки» экземпляров класса Entity.

Важно различать понятие сущности, имеющей определенную роль, от понятия «функциональной роли, выполняемой этой сущностью в контексте определенного действия», специфичной для конкретного действия. Функциональные роли моделируются с помощью экземпляров класса Participation. Например, анестезиолог-ординатор (сущность, имеющая роль) применяет анестезию (что моделируется с помощью экземпляра класса Participation, описывающего функциональную роль «поставщика» в действии «применить анестезию») к пациенту (что моделируется с помощью экземпляра класса Participation, описывающего функциональную роль «получателя» в действии «применить анестезию»). Учтите, что отсутствие прямой ассоциации между классами Participation и Entity вытекает из принципиального положения модели HL7 RIM, согласно которому все экземпляры класса Entity, вовлеченные в действие, указанное в экземпляре класса Act, описывают участие сущностей в этом действии в конкретной роли, указанной в экземплярах класса Role.

В целом классы Participation и Role необходимы для полного моделирования сложной семантики взаимоотношений экземпляров классов Entity и Act. Точное определение точки зрения на оказание медицинской помощи, положенной в основу модели RIM, гласит: на самом верхнем уровне абстракции оказание медицинской помощи представляет собой ряд намеренных атрибутивных действий, в которых разными способами участвуют (выполняют, действуют по поручению, пользуются результатами и т. д.) несколько сущностей, имеющих определенные роли (например, «Джон Смит в роли пациента») и выполняющих определенные функции, описанные экземплярами класса Participaton (например, «участковый врач» и т. д.).

Два остальных базовых класса модели RIM — ActRelationship и RoleLink — используются, чтобы «объединить» или «связать» экземпляры классов, с которыми они ассоциированы. Класс RoleLink используется, чтобы описать «связь на основе зависимости» (например, подотчетность, цепочку доверительных отношений и т. д.) между двумя экземплярами сущностей, выполняющих определенные роли. Семантика класса ActRelationship объясняется далее.

А.5.4 Связывание действий. Семантика класса ActRelationship

Понимание семантики и применения класса ActRelationship начинается с понимания «фрактальной» или «пошаговой» природы совокупности действий. В свою очередь, для этого лучше всего отправляться от разветвленной структуры классификации трех типов «связывания отношений», представленных экземплярами класса ActRelationship: «целое/часть» (например, панели лабораторных анализов (см. далее обсуждение «пошаговых действий»)); «связь на основе правил» (например, план лечения, протоколы и т. д.); «когнитивные действия» (например, суждение, переименование, замена, категоризация, обоснование, причина и т. д.).

В части уже упомянутой «фрактальной» или «пошаговой» природы экземпляры классов ActRelationship могут использоваться для моделирования понятий «фрактальных» или «пошаговых» отношений внутри иерархии «целое/часть». Рассмотрим хирургическую процедуру, например, лапароскопическую холецистэктомию. Она может быть представлена как единственный экземпляр класса Act или же в качестве альтернативы как «совокупность» (частично упорядоченных) экземпляров класса Act, каждый из которых описывает более тонкие детали, например, получить согласие на проведение операции, применить перед операцией определенные лекарства, управлять анестезией (в течение всей хирургической процедуры), сделать разрез и т. д. В свою очередь, каждое из этих более тонких действий может быть расчленено на еще более тонкие действия. Уровень степени детализации четко зависит от контекста действия или действий и степени интереса или намерений стороны, выполняющей (или не выполняющей) декомпозицию действий. На рисунке 15 показана «точка зрения хирурга» на некоторые экземпляры классов Act и ActRelationship при моделировании действия «холецисэктомия».

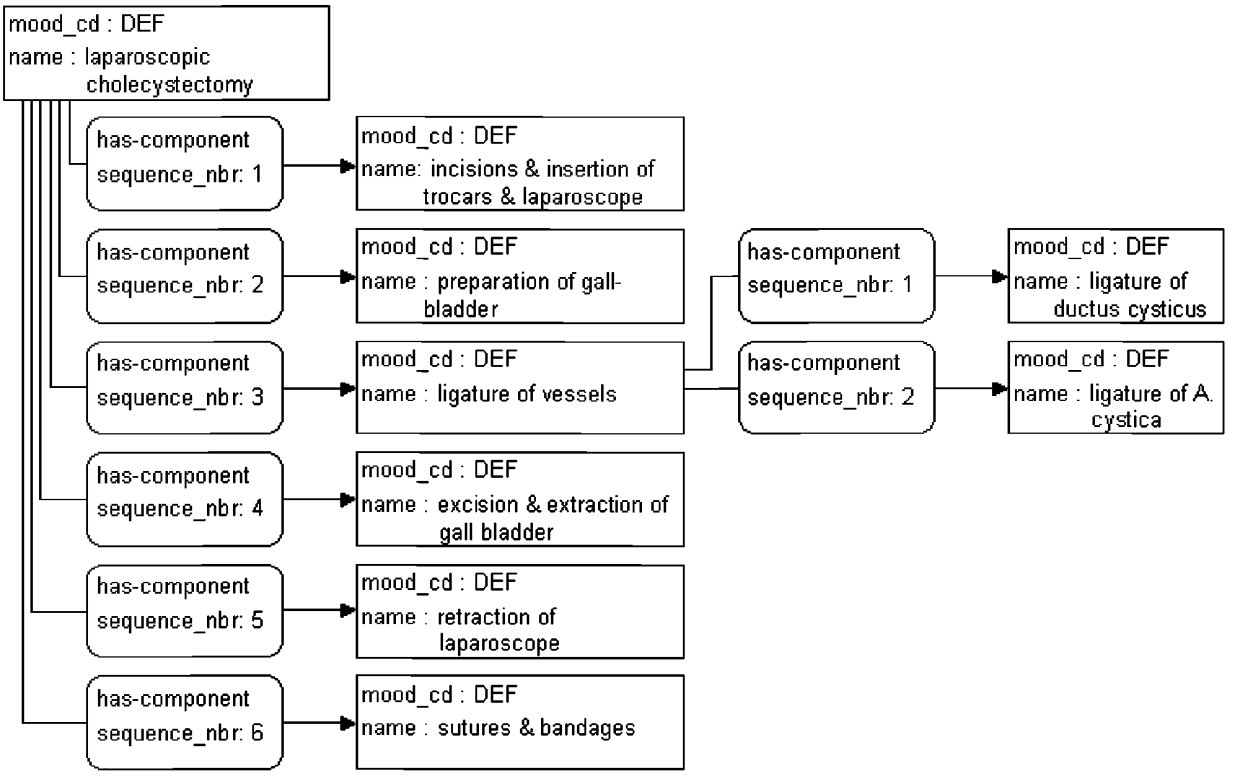


Рисунок А.15 — Пример конструкции последовательного плана лапароскопической холецистэктомии

Прямоугольники с острыми углами представляют экземпляры класса Act (у каждого из них атрибут moodCode имеет значение «DEF», что означает описание действия). Скругленные прямоугольники представляют экземпляры класса ActRelationship, у которых атрибут typeCode имеет значение «COMP» (has component — имеет компонент). Значения атрибута sequenceNumber определяют порядок связи. Каждое действие, в свою очередь, может быть расщеплено на компоненты плана.

В целом классы Act и ActRealshionship необходимы для полного моделирования семантики каждого из трех упомянутых ранее типов «связывания отношений» с той степенью грубости или тонкости, которая необходима для конкретного контекста передачи данных. Кроме того, различные типы связывания и разные уровни детализации, представленные экземплярами класса ActRealshionship, могут (и будут) использоваться для коллективного формирования сложной семантики клинического мышления. Например, экземпляр класса ActRealshionship (описывающий связь «имеет обоснование») может использоваться для формирования связи экземпляра класса Observation, описывающего конкретный результат лабораторного анализа (например, скорость оседания эритроцитов равна 48), с экземпляром класса Observation, описывающим конкретный диагноз (например, системная красная волчанка).

А.5.5 Определения шести базовых классов RIM

А.5.5.1 Класс Act (действие)

Класс Act описывает интересующее действие, которое произошло, может произойти, происходит, запланировано или затребовано. Оно должно быть намеренным действием в предметной области стандарта HL7. Оказание медицинской помощи (и любая профессиональная или деловая деятельность) состоит из намеренных действий. Экземпляр класса Act представляет собой запись о таком намеренном действии.

А.5.5.2 Класс Entity (сущность)

Класс Entity описывает класс физических сущностей, или конкретный экземпляр такой сущности, или организацию либо группу физических сущностей, которые могут участвовать в действиях, артефакт. К ним относятся живые существа (включая людей), организации, материал и места. Иерархия специализаций класса Entity охватывает информацию о людях, организациях, живых организмах, устройствах, фармацевтических субстанциях и т. д. Она не включает в себя информацию о событиях, действиях, деятельности, определения предметов или о ролях, выполняемых сущностями (например, пациент, поставщик медицинской помощи).

А.5.5.3 Класс Role (роль)

Класс Role описывает компетентность сущности, выполняющей роль в соответствии с определениями, данными другой сущностью, контролирующей эту роль.

Сущность, имеющая конкретную роль, может участвовать в действии. Следует отметить, что конкретная сущность в конкретной роли может участвовать в действии многими способами. Например, лицо в роли врача может участвовать в ведении пациента как дежурант или как лечащий врач. Роль определяет компетентность сущности, не зависящую от любого действия, в противоположность участию, которое ограничено рамками действия.

Каждая роль выполняется одной сущностью (той, что имеет эту роль) и обычно «контролируется» другой сущностью. Например, роль «пациент» может выполняться (обычно) некоторым лицом и контролироваться организацией, оказывающей пациенту медицинскую помощь. Аналогичным образом работодатель контролирует роль «работника».

А.5.5.4 Класс Participation (участие)

Класс Participation моделирует ассоциацию между классом роли Role и классом действия Act. Экземпляр класса Participation описывает участие сущности, имеющей определенную роль, в ассоциированном действии. Один и тот же экземпляр класса Role может быть связан с помощью экземпляров класса Participation с несколькими экземплярами класса Act, а один и тот же экземпляр класса Act может быть связан с помощью экземпляров класса Participation с несколькими экземплярами класса Role. Один экземпляр класса Participation всегда является ассоциацией между конкретным экземпляром класса Role и конкретным экземпляром класса Act. Участие ограничено рамками действия, роль же, напротив, описывает компетенцию сущности, не зависящую от какого-либо действия.

А.5.5.5 Класс ActRelationship (связь действий)

Класс ActRelationship моделирует ассоциацию между двумя экземплярами класса Act. Примерами могут служить такие отношения между действиями, как целое/часть, предшественник/последователь, причина/следствие.

Класс ActRelationship имеет две ассоциации с классом Act, одна из них имеет имя source (источник), вторая — target (цель). Связи, ассоциированные с экземпляром класса Act, должны рассматриваться как свойства экземпляра-источника. Это означает, что источник информации, передаваемой в экземпляре класса Act, несет ответственность не только за значения атрибутов этого экземпляра, но и за все связи с этим экземпляром, имеющие его в качестве источника.

Правило приписывания состоит в том, что все экземпляры класса ActRelationship приписаны ответственному действующему лицу того экземпляра класса Act, который является источником ассоциации с экземплярами класса ActRelationship (действием-источником).

А.5.5.6 Класс RoleLink (связь ролей)

Класс RoleLink описывает связь между двумя ролями, выражающую зависимость между этими ролями.

А.5.5.7 Подклассы классов Act, Entity и Role

Классы Act, Entity и Role представляют собой классы «высокого уровня», хотя и не являются «абстрактными» в формальном смысле (т. е. значащие экземпляры этих классов вполне обычны). Поэтому потребовалось определить некоторое число более специализированных подклассов этих трех классов, чтобы указать дополнительные данные (атрибуты классов), требуемые в более конкретном контексте (например, класс Observation, являющийся подклассом класса Act, и классы LivingSubject и Material, являющиеся подклассами класса Entity). Атрибуты подкласса должны быть и полезны, и уникальны для этого подкласса. Подклассы наследуют все атрибуты родительского класса.

В каждой из этих иерархий есть значимые подклассы, для которых не нужны дополнительные атрибуты, поэтому они не представляются как отдельные классы в RIM. Атрибут «classCode» в каждой из этих иерархий указывает, какой именно подкласс представлен данным классом. Система кодирования значений атрибута «classCode» строго контролируется комитетом HL7. Второй атрибут в каждой иерархии — атрибут «code» — используется для дальнейшей классификации подтипов каждого подкласса.

А.5.5.8 Понятие наклонения

Класс Act описывает намеренные действия. Эти действия могут существовать в разных «наклонениях» («moods»). Наклонения описывают цикл деятельности от определения деятельности к ее планированию, от запланированной или потребованной деятельности к завершенной. Наклонение действия, описанного в экземпляре класса Act, определяется значением атрибута Act.moodCode.

Любой экземпляр класса Act имеет одно и только одно наклонение и не меняет его на протяжении своего жизненного цикла. Наклонения — определение, намерение, требование, событие — описывают разные моменты жизненного цикла деятельности. Однако участники каждого такого момента деятельности различны, как и данные, которыми они оперируют. Поэтому наклонение, описанное в экземпляре класса Act, является статическим. Актуализацию процесса деятельности (т. е. движение от ее определения к планированию, требованию и выполнению) называют «циклом деятельности», чтобы отличить ее от «жизненного цикла» единственного экземпляра действия. Экземпляры класса Act, формирующие такой «цикл деятельности», связаны между собой с помощью экземпляров класса ActRelationship.

А.5.6 Типы данных и спецификации словарей

Определения классов модели RIM, атрибутов и ассоциаций дают информацию о логическом значении этих элементов, но для полного определения необходимы типы данных и спецификации словарных доменов. Типы данных определяют допустимые значения атрибутов и «смысл» этих значений. Поэтому типы данных являются фундаментальными строительными блоками, образующими (и ограничивающими) всю семантику, которую в конечном счете можно выразить в модели RIM. В стандарте HL7 Версии 3 спецификации типов данных описаны в документах Data Types Abstract Specification (абстрактная спецификация типов данных) и V3 Data Types Implementable

Technology Specification for XML (технологическая спецификация реализации типов данных Версии 3 на языке XML). Сводный перечень типов данных приведен также в разделе А.6.

Спецификации словарей значений атрибутов указывают словарный домен по умолчанию и «квалификатор расширяемости» («coding strength»). Словарные домены явно определены в разделе 9. Спецификации словарей документируют перекрестные ссылки и альтернативные представления систем кодирования, а также описывают логические понятия, выражаемые каждым кодом. Каждый кодированный атрибут может принимать значения только из соответствующего словарного домена.

Квалификатор расширяемости (Coding strength) является кодированным свойством атрибута, указывающим, может ли пользователь стандарта HL7 Версии 3 передавать кодированное значение, не входящее в словарный домен. Такое значение называется «исключением» для словарного домена. Квалификатор расширяемости может иметь одно из двух значений. Значение квалификатора расширяемости «**CWE**» (Coded with Exceptions — кодируемый с исключениями) указывает, что конечный пользователь может передавать кодированный термин, взятый из местной системы кодирования, а не из формально определенного словарного домена. Значение квалификатора расширяемости «**CNE**» (Coded, No Exceptions — кодируемый без исключений) указывает, что экземпляры сообщений могут содержать только коды из словарного домена, ограничивающего значения данного атрибута.

А.5.7 Методология стандарта HL7, версия 3 и модель RIM

В целом основным мотивом разработки модели RIM было желание точного определения различных элементов данных и отношений, образующих информационное пространство здравоохранения. Прямым результатом такого способа применения знаний явилась способность повторного использования того же самого понятия во многих сообщениях передачи медицинских данных. Полный процесс определения сообщения определен и обсужден в документе V3 Guide и в дополняющем его руководстве HL7 Message Definition Framework (MDF). Основными шагами этого процесса являются определение и документирование потребностей в обмене данными (например, комплекса сообщений для поддержки конкретного клинического или административного процесса) в форме сценариев (storyboard), выбор из модели RIM тех классов и атрибутов, которые необходимы для составления данного сообщения, и последующее применение дополнительных ограничений на число повторений и допустимые значения каждого атрибута.

Дополнительную информацию см. в упомянутых ранее документах V3 Guide и MDF.

А.6 Сводный перечень типов данных версии 3

А.6.1 Обзор типов данных

В таблице А.1 приведены определения типов данных, которые были приняты как часть стандарта HL7, версия 3 по результатам голосования, проведенного 2 декабря 2002 г. Следующие версии модели RIM будут содержать полную спецификацию типов данных, а тело модели RIM будет содержать гиперссылки на определения типов данных.

Таблица А.1 — Типы данных в модели RIM

Имя	Символ	Описание
DataValue	ANY	Определяет основные свойства каждого типа данных. Это абстрактный тип, означающий, что никакое значение не может быть только значением данных, не принадлежащим никакому конкретному типу. Каждый конкретный тип является специализацией этого общего абстрактного типа DataValue
Boolean	BL	Булевский тип представляет значения двузначной логики. Булевское значение может быть или TRUE (истина), или FALSE (ложь), или, как и любое другое значение, может быть пустым (NULL)
Encapsulated Data	ED	Данные, в основном рассчитанные на чтение человеком или на дальнейшую машинную обработку данных за пределами области применения стандарта HL7. К ним относятся неформатированный или форматированный текст, мультимедийные данные, или структурированная информация, определенная другим стандартом (например, XML-подпись). Вместо самих данных значение типа ED может содержать только ссылку на данные (см. описание типа данных TEL). Заметьте, что тип данных ST — это специализация типа данных ED, у которой компонент типа среды ED имеет значение text/plain (только текст)
Character String	ST	Строковый тип данных, предназначенный для текстовых данных, используемых главным образом для машинной обработки (например, сортировка, запрос, индексирование и т. д.). Используется для имен, символов и формальных выражений

Продолжение таблицы А.1

Имя	Символ	Описание
Concept Descriptor	CD	Описание понятия, предназначенное для представления любого вида понятий, обычно с помощью кода, определенного в некоторой системе кодирования. Описание понятия может содержать исходный текст или фразу, послужившую основой для кодирования, и один или несколько эквивалентов этого кода в других системах кодирования. Описание понятия может также содержать квалификатор, используемый, к примеру, для описания понятия «левая нога» как посткоординированного термина, образованного от исходного кода «нога» и квалификатора «левый». В исключительных случаях описание понятия может не содержать код, а только исходный текст, описывающий это понятие
Coded Simple Value	CS	Кодированные данные в их простейшей форме, в которой указан только код без его значения, предназначенного для вывода на экран. Система кодирования и ее версия определяются контекстом, в котором передается значение типа CS. Тип данных CS используется для значений кодированных атрибутов, имеющих единственный словарный домен, определенный в стандарте HL7
Coded With Equivalents	CE	Кодированные данные, состоящие из кодированного значения (типа CV) и, необязательно, одного или нескольких кодированных значений, принадлежащих другим системам кодирования и описывающих то же самое понятие. Используется при наличии альтернативных кодов
Character String with Code	SC	Строка символов, к которой при необходимости может быть добавлен код. Если код присутствует, то должен присутствовать и текст. Код часто берется из местной системы кодирования
Instance Identifier	II	Идентификатор, который уникально идентифицирует предмет или объект. Примерами могут служить объектные идентификаторы (ОИД) объектов RIM HL7, номер медицинской карты, идентификатор направления, идентификатор позиции прейскуранта, идентификационный номер транспортного средства и т. д. Идентификаторы экземпляров определяются на основе объектных идентификаторов ветви ИСО
Telecommunication Address	TEL	Номер телефона (голосового или факса), адрес электронной почты или другой указатель ресурса, управляемого телекоммуникационным оборудованием. Адрес указывается в форме Единого указателя ресурсов URL (Uniform Resource Locator), к которому добавлена спецификация времени и коды использования, помогающие установить, какой адрес используется в конкретное время и для каких целей
Postal Address	AD	Почтовый, домашний или юридический адрес. Последовательность компонентов адреса, например, улица или почтовый ящик, город, почтовый индекс, страна и т. д.
Entity Name	EN	Фамилия, имя, отчество лица, название организации, места или предмета. Последовательность компонентов, например, имя или фамилия, приставка, суффикс и т. д. Примерами могут служить «Джим Боб Уолтон, мл.», «HL7, Inc.», «Озеро Tahoe» и т. д. Именование объекта может быть простой строкой символов или состоять из нескольких компонентов, например, «Джим», «Боб», «Уолтон» и «мл.»; «HL7» и «Inc.»; «Озеро» и «Tahoe»
Trivial Name	TN	Ограничение именования сущности в форме одной строки, используемой для простого названия предметов и мест.
Person Name	PN	Фамилия, имя, отчество лица. Последовательность компонентов, например, имя, фамилия, приставка, суффикс и т. д.
Organization Name	ON	Наименование организации. Последовательность компонентов наименования
Integer Number	INT	Целые числа (–1,0,1,2,100,3398129 и т. д.) — точные числа, являющиеся результатами подсчета и нумерации. Целые числа дискретны, набор целых чисел бесконечен, но счетен. Диапазон значений целых чисел не ограничен никаким произвольным пределом. Для типа данных INT определены две причины пустоты (NULL flavor), описывающие положительную и отрицательную бесконечность

Окончание таблицы А.1

Имя	Символ	Описание
Real Number	REAL	Вещественные числа. Обычно используются в тех случаях, когда количества измеряются, оцениваются или вычисляются из других вещественных чисел. Типичным представлением вещественного числа служит десятичная запись, в которой число значащих десятичных цифр известно как точность
Ratio	RTO	Количество, представленное как отношение числителя к знаменателю. Общие множители в числителе и знаменателе автоматически не сокращаются. Тип данных RTO используется для титров (например, «1:128») и других величин в результатах лабораторных анализов, которые действительно представляют собой отношения. Отношения не являются просто «структурированными числовыми данными», поэтому измерения артериального давления (например, «120/60») не являются отношениями. Во многих случаях вместо типа данных RTO должен использоваться тип данных REAL
Physical Quantity	PQ	Размерная величина, выражающая результат измерения.
Monetary Amount	MO	Денежная сумма, выражающая количество денег в некоторой валюте. Валюты — единицы, в которых указаны денежные суммы в разных экономических регионах. В то время как денежная сумма является единственным видом количества (денег), обменные курсы валют являются переменными. Это принципиальное отличие между физическим количеством и денежными количествами, а также причина того, почему единицы валюты не являются физическими единицами
Point in Time	TS	Величина, указывающая момент времени на оси естественного времени. Момент времени чаще всего представляется как календарное выражение
Set	SET	Значение, содержащее другие различные значения ни в каком конкретном порядке
Sequence	LIST	Значение, содержащее другие дискретные значения в определенной последовательности
Bag	BAG	Неупорядоченная коллекция значений, в которой каждое значение может содержаться более одного раза
Interval	IVL	Множество последовательных значений упорядоченного базового типа данных
History	HIST	Множество значений данных, которые соответствуют типу исторических данных (HXIT), т. е. имеют свойство действительного времени. Историческая информация не ограничена прошлым; в ней могут присутствовать ожидаемые будущие значения
Uncertain Value — Probabilistic	UVP	Общее расширение типа данных, используемое, чтобы указать вероятность, отражающую степень доверия поставщика данных к их значениям
Parametric Probability Distribution	PPD	Общее расширение типа данных, выражающее неопределенность количественных данных с помощью функции распределения и ее параметров. Помимо специфических параметров распределения всегда указывается среднее значение (математическое ожидание) и стандартное отклонение, чтобы обеспечить минимальный уровень интероперабельности на тот случай, если приложение-получатель не в состоянии работать с определенным распределением вероятности
General Timing Specification	GTS	Множество моментов времени, указывающее времена событий и действий, а также шаблоны циклов, которые могут применяться к некоторым видам информации, например, к телефонным номерам (утро, вечер), к адресам (для так называемых «перелетных птиц», которые живут зимой на юге, а летом на севере), а также к часам работы

Приложение В
(справочное)

Типы данных. Абстрактная спецификация

Основной/технический редактор: Gunther Schadow, gunther@aurora.rg.iupui.edu, Regenstrief Institute for Health Care.

Технический редактор: Paul Biron, paul.v.biron@kp.org Kaiser Permanente, Southern California.

Технический редактор: Lloyd McKenzie, lmckenzi@ca.ibm.com, IBM Global Services.

Технический редактор: Grahame Grieve, grahame@kestral.com.au, Kestral Computing Pty. Ltd.

Технический редактор: Doug Pratt, Douglas.Pratt@siemens.com, Siemens.

Дата и время последней публикации: 20080909 14:49

Предисловие

Типы данных, используемые в стандарте HL7 Версии 3, определены в настоящем документе на абстрактном уровне, не зависящем от представления. Под «независимостью от представления» понимается независимость как абстрактного синтаксиса, так и его применения от конкретной технологии реализации.

Этот документ сопровождается спецификациями технологии реализации ITS (Implementation Technology Specification). Документы ITS могут служить краткими справочниками по типам данных, которые лучше ориентированы на практическое представление этих типов с помощью данной технологии реализации.

Для ускоренного информирования в таблицах словарных значений, приведенных в настоящей спецификации, перечислено текущее содержание словарных доменов. Однако в любой заданный момент времени нормативным источником этих доменов являются таблицы словарных значений, хранящихся в базе данных модели RIM. Для некоторых больших доменов приведены только примеры возможных значений. Для получения полного содержания домена можно осуществить поиск по имени домена, ассоциированному с таблицей значений, хранящейся в словаре модели RIM.

Выражения признательности

Настоящая спецификация воплощает результат многолетней интенсивной работы с использованием электронной почты, телеконференций и обсуждений на различных встречах. Ее содержание согласовано с помощью голосования. Выражаем признательность многих лицам, которые в разное время участвовали в проектировании, дискуссиях и подаче предложений на голосование. Председатель рабочей группы по разработке этой спецификации Gunther Schadow (Regenstrief Institute for Health Care) возглавлял эту работу и стал основным автором настоящего документа. В разное время соредакторами этого документы были Paul V. Biron (Kaiser Permanente), Doug Pratt (Siemens), Lloyd McKenzie (IBM) и Grahame Grieve (Kestral Computing Pty. Ltd.). Основной вклад в форме размышлений и поддержки внесли (Regenstrief Institute), George Beeler, Stan Huff (Intermountain Health Care), а также Mike Henderson (Kaiser Permanente), Anthony Julian (Mayo), Joann Larson (Kaiser Permanente), Mark Shafarman (Oasis Healthcare Systems), Wes Rishel (Gartner Group), и Robin Zimmerman (Kaiser Permanente). Выражаем признательность за критические замечания и предложения новых идей Bob Dolin (Kaiser Permanente), Clem McDonald (Regenstrief Institute), Kai Heitmann (HL7 Germany), Rob Seliger (Sentillion), и Harold Solbrig (Mayo Clinic). Неоценимую поддержку оказали члены рабочей группы Laticia Fitzpatrick (Kaiser Permanente), Matt Huges, Randy Marbach (Kaiser Permanente), Larry Reis (Wizdom Systems), Carlos Sanroman (Kaiser Permanente), Greg Thomas (Kaiser Permanente). Благодарим James Case (University of California, Davis), Norman Daoust (Partners HealthCare Systems), Irma Jongeneel (HL7 The Netherlands), Michio Kimura (HL7 Japan), John Molina (SMS), Richard Ohlmann (McKessonHBOC), David Rowed (HL7 Australia) и Klaus Veil (Macquarie Health Corp., HL7 Australia) за предоставление своих знаний при решении критичных вопросов. Разработка настоящей спецификации была обеспечена организацией Regenstrief Institute for Health Care.

Открытые вопросы

Спецификация соответствия с помощью ограничения типов данных.

В.1 Введение

В.1.1 Что такое тип данных?

Каждый элемент данных имеет определенный тип данных, определяющий смысл (семантику) значений, которые могут быть присвоены элементу данных. Для содержательного обмена данных необходимо знать определения передаваемых значений. Это верно как для комплексных «значений», например, деловых сообщений, так и для более простых значений, например, строк символов или целых чисел.

В соответствии со стандартом ИСО 11404, тип данных представляет собой «совокупность различных значений, характеризуемую свойствами этих значений и операциями над этими значениями». С типом данных связаны специфика и широта. Спецификой типа данных являются свойства, присущие каждому значению этого типа. Широта типа данных оценивается множеством значений, имеющих этот тип («набором значений» этого типа).

Семантическими свойствами типов данных является то, что стандарт 11404 называет «свойствами этих значений и операциями над этими значениями». Семантическому свойству типа данных присваивается некоторое

имя. Семантическое свойство имеет определенное значение для каждого значения этого типа данных. Значение свойства значения данных само должно иметь определенный тип данных — не существует никаких значений данных, для которых нельзя было бы определить тип данных.

Поэтому типы данных являются основными строительными блоками, используемыми для конструирования сущностей более высокого порядка — сообщений, компьютеризованных медицинских документов, объектов деловой сферы и транзакций с этими объектами. В чем же тогда состоит различие между типом данных и сообщением, документом или деловым объектом? Значения типа данных существуют сами по себе, рассматривается только значение, для него не определяются ни идентичность, ни состояние, ни изменение состояния. Напротив, для деловых объектов можно отслеживать идентичность и состояние, свойства идентифицированного объекта могут изменяться с течением времени. Этого нет у значений данных: значение и его свойства постоянны. Например, число 5 всегда является числом 5, между этим числом 5 и тем числом 5 нет никакой разницы (они не обладают идентичностью), число 5 никогда не станет числом 6 (состояние не изменяется). Значения данных можно рассматривать как неизменные объекты, идентичность которых не играет никакой роли (идентичность и равенство суть одно и то же)[6)].

B.1.2 Представление значений данных

Значения данных могут быть представлены с помощью разных символов, но смысл значения не связан ни с каким конкретным представлением.

Например, порядковые номера (неотрицательные целые числа) намеренно определены как тип данных, в котором для каждого значения есть следующее значение, а нуль не следует ни за каким порядковым значением. На основе этого определения можно определить операции сложения, умножения и другие математические операции. Какое бы представление не отражало правила, описанные в намеренном определении порядкового типа данных, оно будет действительным представлением порядковых номеров. Примерами действительных представлений порядковых номеров могут служить строки десятичных цифр, пакеты стеклянных шариков или царапины на стене. Число «пять» может быть представлено словом «пять», арабской цифрой «5» или римской цифрой «V». Представление не играет роли, пока оно соответствует семантическому определению типа данных.

Другим примером может служить булевский тип данных, который определяется совокупностью двух разных значений, соответствующих истине (true) и лжи (false), и правилами отрицания значения и сочетания этих значений в операциях конъюнкции и дизъюнкции. Булевские значения могут быть представлены словами «true» и «false», «да» и «нет», числами 0 и 1 или любыми двумя знаками, отличающимися друг от друга. Представление значений типа данных не играет роли, пока оно соответствует семантическому определению типа данных.

Настоящая спецификация определяет семантику, смысл типов данных, используемых в стандартах HL7. **Она касается только семантики и не зависит от особенностей представления или обработки либо от специфической технологии реализации**. Для различных технологических подходов представления значений определенных в ней типов данных разработаны дополнительные стандарты. Они называются «Спецификацией реализуемой технологии» (Implementable Technology Specification — ITS). Эти стандарты указывают, как должны представляться значения, соответствующие семантическим определениям, приведенным в настоящей спецификации, они могут задавать синтаксис символьного или двоичного представления, а также описывать компьютерные процедуры выполнения действий над этими представлениями значений данных. Смысл этих представлений, используемых при передаче данных, при создании данных и их обработке компьютерными программами определяется на основе настоящее стандарта, являющегося семантической спецификацией типов данных.

B.1.3 Свойства значений данных

Значения данных имеют свойства, определяемые их типом данных. Наиболее общим примером таких свойств являются «поля комплексных типов данных». Однако в более общем смысле под свойством значения данных надо подразумевать логические предикаты или математические функции; в более простых, но тем не менее корректных терминах свойства являются вопросами, которые надо задать о значении данных, чтобы получить в ответ другое значение данных.

Свойство обозначается именем. Например, тип данных integer (целое число) может иметь свойство с именем «sign» (знак). Свойство имеет домен, образованный множеством возможных «ответов». Это множество определяется типом данных свойства, но домен свойства может быть подмножеством набора значений этого типа данных.

Свойство может иметь аргументы, то есть дополнительную информацию, которую надо представить в вопросе для получения ответа. Например, важным свойством целого числа является то, что результат сложения одного целого числа с другим также представляет собой целое число. Поэтому для свойства целого числа «сложение» требуется аргумент, а именно, другое целое число.

Имеют ли семантические свойства аргументы или нет, не является сколько-нибудь принципиальным отличием. Семантическое свойство типа данных, не имеющее аргументов, не обязательно представляет собой «поле комплексного типа данных». Например, для целочисленных значений можно определить свойство is-zero, прини-

---

[6)] В методологии HL7 Message Development Framework определены «режимы изменения» полей сообщения. Поскольку значения данных не обладают ни идентичностью, ни состоянием, ни изменением состояния, то эти режимы не применяются к свойствам значений данных. Значения данных и их свойства никогда не изменяются. Поле объекта (например, сообщения) может быть изменено, и в этом случае значение этого поля заменяется другим значением. Но само значение никогда не изменяется.

мающее булевское значение «true», если число равно нулю, и «false», если оно отлично от нуля. Это отнюдь не означает, что свойство is-zero должно быть явным компонентом любого представления целого числа.

Семантическое свойство типа данных, имеющее аргументы, не имеет специфических операционных обозначений наподобие «вызова процедуры», «передачи аргументов», «возвращения значений», «инициирования исключений» и т. д. Все перечисленное относится к реализации типов данных в компьютерных системах, но не имеет отношения к семантике типов данных.

Настоящая спецификация затрагивает только семантику типов данных. В ней не обсуждается ни синтаксис представления значений (даже абстрактный синтаксис), ни интерфейс операций над значениями данных.

B.1.4 Необходимость абстрагирования

Почему в настоящей спецификации делается такой акцент на абстрагирование от синтаксиса представления и реализации операций?

Такой вид абстрактной семантической спецификации типа данных необходим в стандартах HL7 по весьма практическим соображениям. Одной из важных особенностей конструирования стандартов HL7 Версии 3 является открытость по отношению к технологиям представления и реализации. Предполагается, что все спецификации стандартов HL7 Версии 3 будут представляться в форме, независимой от технологий представления и реализации. Комитет HL7 отдает себе отчет о том, что какое-то время некоторые технологии представления и реализации являются более популярными, но технологии подвержены изменениям, и при изменение технологии представление данных также изменится. Основной областью применения стандартов HL7 является обработка информации в сфере здравоохранения, не зависящая от технологии, обеспечивающей эту обработку. Комитет HL7 рассчитывает, что спецификации, не зависящие от современной технологии, будут оставаться полезными даже после очередной «смены технологической парадигмы».

Спецификация типов данных ближе к технологии реализации, нежели большинство других информационных стандартов HL7. Поэтому существует определенная опасность, что описания типов данных окажутся слишком зависимыми от современных технологий реализации.

Большинство стандартов HL7 посвящено сложным деловым объектам. Такие объекты, обладающие большим числом информационных атрибутов, могут быть определены с помощью абстрактного синтаксиса, в котором компоненты объектов описываются в терминах типов данных. Напротив, определение типов данных в терминах абстрактного синтаксиса приносит мало пользы, поскольку компоненты соответствующих синтаксический конструкций сами должны иметь типы данных[7)].

Почему настоящая спецификация столько циклична? Почему тип данных «ANY» определен в терминах специализации самого себя?

Настоящая спецификация должна быть независимой от конкретной реализации, следовательно, абстрактной, не предназначенной для реализации. В этом отношении цикличность не является проблемой, поскольку не вносит никакой неопределенности в содержание спецификации.

Почему в настоящей спецификации не определен набор примитивных типов данных, с помощью которых комплексные типы данных могут быть определены в терминах абстрактного синтаксиса?

Технология любой конкретной реализации стандартов HL7 должна опираться исключительно на встроенные типы данных. Поэтому необходимо иметь возможность очень гибкого отображения абстрактных типов данных HL7 на типы данных, встроенные в конкретную технологию реализации. Соответствие спецификации реализуемой технологии (Implementable Technology Specification, ITS) семантической спецификации может быть обеспечено просто с помощью отображения конструкций, используемых этой технологией, на семантику типов данных HL7 Версии 3. С точки зрения семантики не имеет никакого значения, является ли тип данных примитивным или комплексным, и ответ может различаться для разных технологий реализации.

Например, настоящий стандарт описывает строку символов как тип данных со многими свойствами (например, кодировка, язык и т. д.). Однако во многих технологиях реализации, строки символов являются примитивными типами данных первого класса. Рекомендуется использовать такие нативные типы данных вместо структур, раболепно представляющих все семантические свойства как «компоненты». Настоящая спецификация требует лишь то, что свойства, определенные для значений данных, должны каким-то образом выводиться из выбранного представления, а само представление особого значения не имеет. Не столь важно, используются ли «примитивные» или «комплексные» типы данных, с малым или большим числом «компонентов», определяемых как «поля» или «методы».

Другим примером могут служить такие представления вещественного числа, как десятичное представление, представление с плавающей точкой и представление в виде масштабированного целого числа. Все они являются нативными для разных технологий реализации. Некоторые из них имеют свойства, которыми другие не обладают. Например, представление в виде масштабированного целого числа обеспечивает фиксированную точность и относительно небольшой диапазон значений. Представление в виде числа с плавающей точкой обеспечивает переменную точность и большой диапазон значений, однако информация о точности при этом потеряна. Представление в виде десятичного числа обеспечивает переменную точность и сохраняет информацию о точности (но

[7)] По этой причине Абстрактная синтаксическая нотация версии один (ACH.1), стандартизованная ISO, не может рассматриваться как формализм, пригодный для семантических спецификаций типов данных.

при этом медленнее обрабатывается). Семантика типов данных должна не зависеть от таких случайных свойств различных представлений и должна определять существенные свойства, которые могут быть представлены с помощью любой технологии.

B.1.5 Необходимость в стандарте типов данных HL7

Зачем комитету HL7 понадобился собственный стандарт типов данных? Почему нельзя было просто воспользоваться стандартом другого разработчика?

Как упоминалось в предыдущем разделе, во всех технологиях реализации, предложенных комитетом HL7, используется некоторая система типов данных, но в разных технологиях системы типов данных обладают определенными отличиями. Кроме того, во многих технологиях реализации используемая система типов данных является недостаточно мощной для представления понятий, необходимых в стандартах HL7 прикладного уровня.

Например, понятия физических величин, точности, диапазонов, отсутствующих значений и неопределенности, необходимые в научных вычислениях и при обработке информации в сфере здравоохранения, предусмотрены мало в каких технологиях реализации.

С другой стороны, в технологиях реализации проводятся различия, которые не релевантны абстрактной семантике, например, вещественные числа с фиксированной и плавающей точкой, 8, 16, 32 или 64-битовые целые числа, дата и время и штамп даты и времени.

Для данной спецификации в качестве входной информации использовался целый ряд систем типов данных. К ним относятся системы, используемые в основных языках программирования, включая BASIC, Pascal, MODULA-2, C, C++, JAVA, ADA, LISP и SCHEME. Использовались и системы типов данных, не зависящие от языков программирования, например, Абстрактная синтаксическая нотация версии один (ACH.1), язык описания интерфейсов IDL (Interface Definition Language) и язык объектных ограничений OCL (Object Constraint Language), разработанные организацией OMG (Object Management Group), языки манипулирования данными SQL 92 и SQL 99, не зависящие от языка типы данных, описанные в стандарте ИСО 11404, а также типы данных, описанные в спецификации XML Schema Part 2. Рассматривались также типы данных, используемые в стандартах информатизации здоровья, в том числе HL7 версии 2.x, стандарты сообщений обмена медицинскими данными, принятые техническим комитетом CEN TC 251, архитектура электронной медицинской карты EHCRA (Electronic Health Record Architecture) и стандарт DICOM.

B.1.6 Требования

При разработке типов данных, описанных в настоящей спецификации, учитывался ряд требований, включая следующие:

- требования, связанные с процессом моделирования,
- требования реализации,
- совместимость с другими стандартами типов данных,
- функциональные требования, описанные в других стандартах HL7, где используются типы данных.

Среди них наибольшее внимание уделялось последней группе. Эти типы данных обеспечивают функциональность, требуемую стандартами HL7. Все эти требования не всегда совместимы и в настоящей спецификации можно найти ряд мест, где конкретное решение не слишком оптимально для какой-либо из четырех указанных выше групп требований. В некоторых из этих случаев в подразделе требований перечислены конкретные требования, в соответствии с которыми предложено данное решение. Эти подразделы требований является не нормативными, а справочными.

Требование

В Эталонной информационной модели RIM определен ряд базовых классов, на которых основаны все информационные модели предметных областей. Каждый из этих базовых классов имеет ряд атрибутов, которым присвоены определенные типы данных. Когда эти классы используются (в качестве клонов) в моделях предметных областей, то для уточнения и ограничения использования атрибутов в клонированном классе типы данных атрибутов могут быть заменены другими типами.

В настоящей спецификации должны быть определены правила подобной замены типов данных. В качестве основы для правил замены в ней используется метафора специализации, поскольку она понятна и широко используется как в теории, так и на практике, и такие правила легче воспринимаются и управляются по сравнению с другими альтернативами. Однако использование специализации может привести к результатам, которые некоторым могут показаться непривычными.

B.1.7 Формы определений типов данных

В настоящей спецификации используется несколько форм определений типов данных, а именно, текстовое описание, диаграммы классов на языке UML, таблицы и формальное определение.

B.1.7.1 Язык формального определения типов данных

Формальное определение типов данных используется, чтобы наиболее однозначным образом описать семантику предлагаемых типов данных. Этот язык определения типов данных детально описан в подразделе 0 «

Введение в формальный язык определения типов данных DTDL (Formal Data Type Definition Language)». Формальные языки обеспечивают возможность составления весьма четких утверждений, которые могут использоваться для представления некоторого формального аргумента в пользу подтверждения или опровержения. Однако лаконичность таких формальных утверждений может также затруднить их понимание человеком. Поэтому все важные выводы из формальных утверждений представлены также в виде предложений на естественном языке.

B.1.7.2 Таблицы свойств

Для удобства ознакомления в начало описаний многих типов данных, приведенных в настоящей спецификации, включены таблицы «основных» свойств. К числу «основных» относится несколько расплывчатая группа тех свойств, которые более похожи на «поля», когда такие типы данных реализуются как записи, или которые наиболее часто используются. Эти таблица предназначены для облегчения обзора содержания и назначения типов данных. Не требуется, чтобы свойства, перечисленные в этих таблицах, были представлены как поля, и эти таблицы не являются определениями абстрактного синтаксиса.

Каждая строка таблицы свойств описывает одно свойство и содержит следующие элементы:

Имя — имя свойства, указанное в формальном определении. Для некоторых типов данных поле имени первого свойства в таблице может быть пустым. Это может иметь место для тех типов данных, которые определены как расширения других типов данных и в перечне свойств дочернего типа нет смысла показывать какие-либо свойства родительского типа.

Тип — тип данных этого свойства.

Определение. Краткий текст, описывающий смысл свойства.

B.1.7.3 Диаграммы на Унифицированном языке моделирования UML (Unified Modeling Language)

Для графического представления связей между типами данных используются диаграммы классов на Унифицированном языке моделирования UML (Unified Modeling Language). Типы данных показаны как классы UML, которым присвоены краткие имена. Порождающие типы показаны как параметризованные классы UML с отношениями реализации, связывающими их экземпляры.

Многие детали объявлений типов данных не могут быть представлены на диаграммах UML. Поэтому для детальной спецификации таких типов данных должно использоваться формальное определение на языке DTDL (Data Type Definition Language — язык описания типов данных).

Некоторые из ограничений в определениях на языке DTDL представлены как ограничения на операции. Если ограничения присутствуют, то они представляют собой утверждения, взятые из спецификации на языке DTDL, которые должны быть истинными.

В диаграммах UML используется стереотип «mixin». Он применяется к параметризованному классу и означает, что этот класс является специализацией типа данных, имя которого является значением параметра T, и в дополнение к собственным свойствам содержит все свойства этого типа данных.

B.1.8 Обзор типов данных

Представление типов данных в форме диаграммы классов на языке UML показано на рисунке B.1. Перечень этих типов данных приведен в таблице B.1.

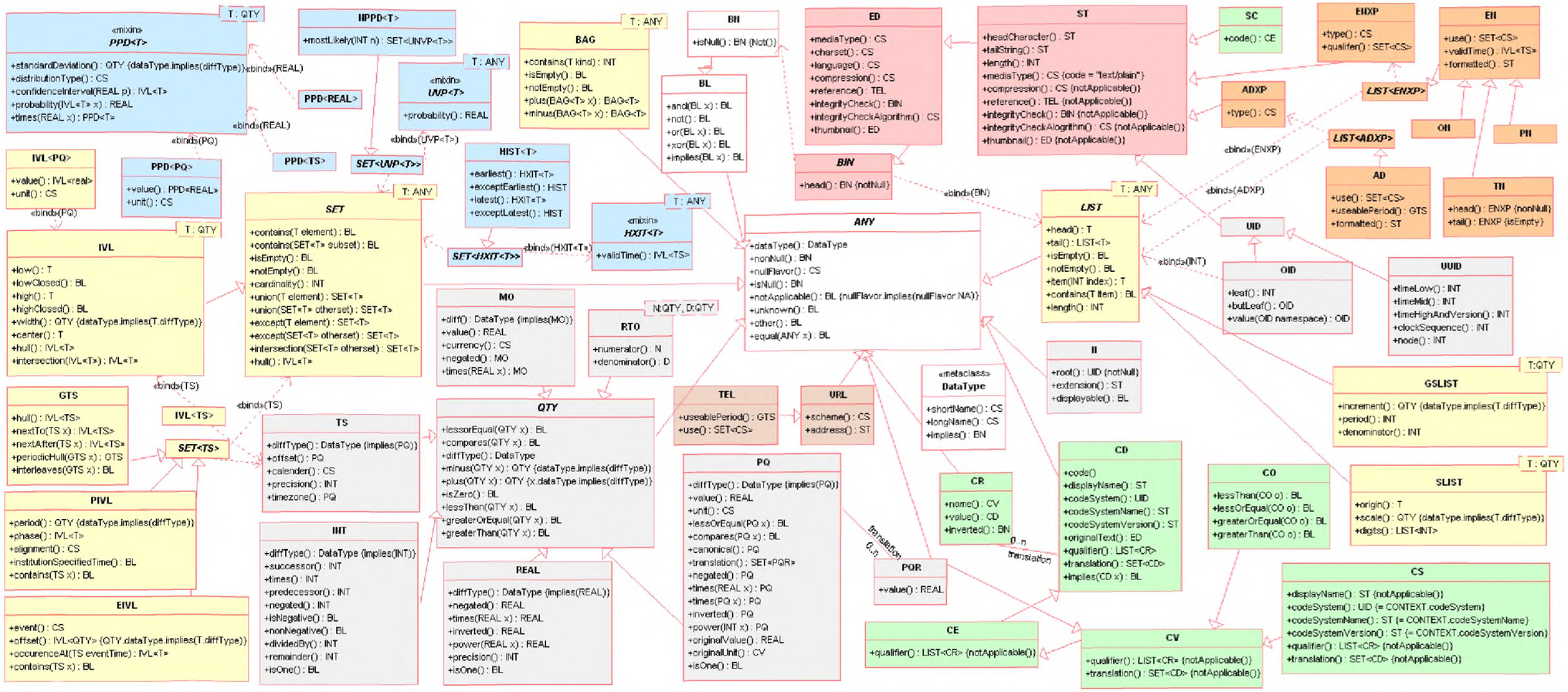


Рисунок В.1 — Представление типов данных в форме диаграммы классов на языке UML

Таблица В.1 — Перечень типов данных в модели RIM

Имя	Символ	Описание
DataValue	ANY	Определяет основные свойства каждого типа данных. Это абстрактный тип, означающий, что никакое значение не может быть только значением данных, не принадлежащим никакому конкретному типу. Каждый конкретный тип является специализацией этого общего абстрактного типа DataValue
Boolean	BL	Булевский тип представляет значения двузначной логики. Булевское значение может быть или TRUE (истина), или FALSE (ложь), или, как и любое другое значение, может быть пустым (NULL)
BooleanNonNull	BN	Тип данных BN представляет собой ограничение типа данных BL, в котором запрещено пустое значение. Этот тип предназначен для использования в тех случаях, когда пустое значение недопустимо
Encapsulated Data	ED	Данные, в основном рассчитанные на чтение человеком или на дальнейшую машинную обработку данных за пределами области применения стандарта HL7. К ним относятся неформатированный или форматированный текст, мультимедийные данные, или структурированная информация, определенная другим стандартом (например, XML-подпись). Вместо самих данных значение типа ED может содержать только ссылку на данные (см. описание типа данных TEL). Заметьте, что тип данных ST — это специализация типа данных ED, у которой компонент типа среды ED имеет значение text/plain (только текст)
Character String	ST	Строковый тип данных, предназначенный для текстовых данных, используемых главным образом для машинной обработки (например, сортировка, запрос, индексирование и т. д.). Используется для имен, символов и формальных выражений
Concept Descriptor	CD	Описание понятия, предназначенное для представления любого вида понятий, обычно с помощью кода, определенного в некоторой системе кодирования. Описание понятия может содержать исходный текст или фразу, послужившую основой для кодирования, и один или несколько эквивалентов этого кода в других системах кодирования. Описание понятия может также содержать квалификатор, используемый, к примеру, для описания понятия «левая нога» как посткоординированного термина, образованного от исходного кода «нога» и квалификатора «левый». В исключительных случаях описание понятия может не содержать код, а только исходный текст, описывающий это понятие
Coded Simple Value	CS	Кодированные данные в их простейшей форме, в которой указан только код без его значения, предназначенного для вывода на экран. Система кодирования и ее версия определяются контекстом, в котором передается значение типа CS. Тип данных CS используется для значений кодированных атрибутов, имеющих единственный словарный домен, определенный в стандарте HL7
Coded Ordinal	CO	Кодированные данные, образующие систему кодирования, в которой коды упорядочены. Этот тип данных добавляет семантику упорядоченности, так что в моделях, использующих такие словарные домены, могут использоваться элементы, для которых существен порядок терминов в домене
Coded With Equivalents	CE	Кодированные данные, состоящие из кодированного значения (типа CV) и, необязательно, одного или нескольких кодированных значений, принадлежащих другим системам кодирования и описывающих то же самое понятие. Используется при наличии альтернативных кодов
Character String with Code	SC	Строка символов, к которой при необходимости может быть добавлен код. Если код присутствует, то должен присутствовать и текст. Код часто берется из местной системы кодирования

Продолжение таблицы В.1

Имя	Символ	Описание
Instance Identifier	II	Идентификатор, который уникально идентифицирует предмет или объект. Примерами могут служить объектные идентификаторы (ОИД) объектов RIM HL7, номер медицинской карты, идентификатор направления, идентификатор позиции прейскуранта, идентификационный номер транспортного средства и т. д. Идентификаторы экземпляров определяются на основе объектных идентификаторов ветви ИСО
Telecommunication Address	TEL	Номер телефона (голосового или факса), адрес электронной почты или другой указатель ресурса, управляемого телекоммуникационным оборудованием. Адрес указывается в форме Единого указателя ресурсов URL (Uniform Resource Locator), к которому добавлена спецификация времени и коды использования, помогающие установить, какой адрес используется в конкретное время и для каких целей
Postal Address	AD	Почтовый, домашний или юридический адрес. Последовательность компонентов адреса, например, улица или почтовый ящик, город, почтовый индекс, страна и т. д.
Entity Name	EN	Фамилия, имя, отчество лица, название организации, места или предмета. Последовательность компонентов, например, имя или фамилия, приставка, суффикс и т. д. Примерами могут служить «Джим Боб Уолтон, мл.», «Health Level Seven, Inc.», «Озеро Tahoe» и т. д. Именование объекта может быть простой строкой символов или состоять из нескольких компонентов, например, «Джим», «Боб», «Уолтон» и «мл.»; «Health Level Seven» и «Inc.»; «Озеро» и «Tahoe»
Trivial Name	TN	Ограничение именования сущности в форме одной строки, используемой для простого названия предметов и мест
Person Name	PN	Тип данных EN, используемый, когда именованной сущностью является физическое лицо. Последовательность компонен-тов, являющихся фамилией, именем, отчеством, префиксом, суффиксом и т. д. Эти компоненты являются ограничениями компонентов имени сущности, позволяющими указывать только те квалификаторы части, которые применимы к фамилиям, именам, отчествам физических лиц. Поскольку структура типа данных EN навеяна в основном фамилиями, именами и отчествами, эти ограничения весьма минимальны
Organization Name	ON	Тип данных EN, используемый для названия организации. Последовательность компонентов названия
Integer Number	INT	Целые числа (–1,0,1,2,100,3398129 и т. д.) — точные числа, являющиеся результатами подсчета и нумерации. Целые числа дискретны, набор целых чисел бесконечен, но счетен. Диапазон значений целых чисел не ограничен никаким произвольным пределом. Для типа данных INT определены две причины пустоты (NULL flavor), описывающие положительную и отрицательную бесконечность
Real Number	REAL	Вещественные числа. Обычно используются в тех случаях, когда количества измеряются, оцениваются или вычисляются из других вещественных чисел. Типичным представлением вещественного числа служит десятичная запись, в которой число значащих десятичных цифр известно как точность
Ratio	RTO	Количество, представленное как отношение числителя к знаменателю. Общие множители в числителе и знаменателе автоматически не сокращаются. Тип данных RTO используется для титров (например, «1:128») и других величин в результатах лабораторных анализов, которые действительно представляют собой отношения. Отношения не являются просто «структурированными числовыми данными», поэтому измерения артериального давления (например, «120/60») не являются отношениями. Во многих случаях вместо типа данных RTO должен использоваться тип данных REAL
Physical Quantity	PQ	Размерная величина, выражающая результат измерения

Окончание таблицы В.1

Имя	Символ	Описание
Monetary Amount	MO	Денежная сумма, выражающая количество денег в некоторой валюте. Валюты — единицы, в которых указаны денежные суммы в разных экономических регионах. В то время как денежная сумма является единственным видом количества (денег), обменные курсы валют являются переменными. Это принципиальное отличие между физическим количеством и денежными количествами, а также причина того, почему единицы валюты не являются единицами физических величин
Point in Time	TS	Величина, указывающая момент времени на оси естественного времени. Момент времени чаще всего представляется как календарное выражение
Set	SET	Значение, содержащее другие различные значения ни в каком конкретном порядке
Sequence	LIST	Значение, содержащее другие дискретные значения в определенной последовательности
Bag	BAG	Неупорядоченная коллекция значений, в которой каждое значение может содержаться более одного раза
Interval	IVL	Множество последовательных значений упорядоченного базового типа данных
History	HIST	Множество значений данных, которые соответствуют типу исторических данных (HXIT), т. е. имеют свойство действительного времени. Историческая информация не ограничена прошлым; в ней могут присутствовать ожидаемые будущие значения
Uncertain Value — Probabilistic	UVP	Общее расширение типа данных, используемое, чтобы указать вероятность, отражающую степень доверия поставщика данных к их значениям
Periodic Interval of Time	PIVL	Периодически повторяющийся интервал времени. Имеет два свойства: фазу и период. Фаза задает «прототип интервала», который повторяется каждый период
Event-Related Periodic Interval of Time	EIVL	Периодически повторяющийся интервал времени, в котором повторение зависит от активности человека в течение дня или от других важных событий, которые связаны со временем, но не имеют точно заданного времени
General Timing Specification	GTS	Множество моментов времени, указывающее времена событий и действий, а также шаблоны циклов, которые могут применяться к некоторым видам информации, например, к телефонным номерам (утро, вечер), к адресам (для так называемых «перелетных птиц», которые живут зимой на юге, а летом на севере), а также к часам работы
Parametric Probability Distribution	PPD	Общее расширение типа данных, выражающее неопределенность количественных данных с помощью функции распределения и ее параметров. Помимо специфических параметров распределения всегда указывается среднее значение (математическое ожидание) и стандартное отклонение, чтобы обеспечить минимальный уровень интероперабельности на тот случай, если приложение-получатель не в состоянии работать с определенным распределением вероятности

В.1.9 Введение в формальный язык определения типов данных DTDL (Formal Data Type Definition Language)

Примечание — Это не спецификация прикладного программного интерфейса (API).

В то время как данный формальный язык может напоминать некоторые языки программирования или определения интерфейсов, он не предназначен для определения деталей программ и других средств реализации. Формальные определения являются нормативной частью настоящей спецификации, но данный конкретный язык не требует реализации или использования в системах, соответствующих стандарту. От них не требуется также реализации или использования всех семантических свойств. Внутренняя деятельность систем, способ реализации типов данных, их функциональность и предоставляемые сервисы полностью выходят за область применения настоящего стандарта. Формальное определение только указывает смысл значений данных с помощью утверждений, как с точки зрения теории эти значения связаны с собой и какими свойствами обладают.

Этот формальный язык определения типов данных[8] задает:

- полное и краткое имя данных;
- именованные значения для полностью перечисляемого расширения;
- семантические свойства, унарные, бинарные свойства, а также свойства более высокого порядка;
- инварианты, то есть ограничения свойств;
- допустимые преобразования типа;
- синтаксис строкового представления значения (если таковое имеется).

Определение типа данных осуществляется в два этапа. Сначала делается объявление типа данных. В объявлении указываются имя нового типа данных и список имен, типов и сигнатур его семантических свойств. Это лишь объявление, а не определение типа. Определение содержит логические утверждения о том, что всегда истинно для значений этого типа и их свойств (инвариантные утверждения).

B.1.9.1 Объявление

Объявление каждого типа данных начинается с ключевого слова **type**. Например, ниже показан заголовок объявления типа данных Boolean, имеющего краткое имя BL и являющегося специализацией типа данных ANY[9]:

```
type Boolean alias BL specializes ANY
  values(true, false)
{
  BL not;
  BL and(BL x);
};
```

Объявление типа данных Boolean содержит конструкцию **values**, специфицирующую полный набор значений типа Boolean (его расширение) в виде именованных сущностей. Эти именованные значения являются также допустимыми строковыми литералами. Поскольку никакой другой тип данных из числа определенных в настоящей спецификации не имеет конечное множество значений, то конструкция **values** уникальна для типа данных Boolean. В размеченном формальном языке имена значений выделяются курсивом.

Блок текста в фигурных скобках, следующий за заголовком, содержит объявления семантических свойств, которыми обладает каждое значение типа данных. Объявление свойства завершается точкой с запятой, а другая точка с запятой, стоящая после закрывающей фигурной скобки, завершает объявление типа данных.

Объявления свойства включает в себя следующие компоненты (слева направо): (1) тип данных домена значений свойства, (2) имя свойства и (3) необязательный список аргументов. Этот список окружен круглыми скобками, внутри которых указана последовательность объявлений аргументов. Объявление аргумента содержит имя типа данных и имя аргумента. Для семантических свойств, не имеющих аргументов, пустой список не указывается[10].

---

[8] Используемый здесь язык определения данных возник в результате экспериментов и опыта применения различных альтернатив, включая таблицы определения типов данных и язык определения интерфейса IDL (Interface Definition Language), предложенный организацией OMG (Object Management Group). Недостаток таблиц определения типов данных в том, что они дают неверное представление, что настоящий документ является спецификаций абстрактного синтаксиса, а не семантики. Аналогично, недостатком IDL является то, что его применение дает неверное впечатление, что данный документ является определением прикладного программного интерфейса API (application programming interface).

Получившийся язык определения типов данных многое позаимствовал от свойств и стиля языка IDL, языка объектных ограничений OCL (Object Constraint Language), языков программирования JAVA, C++, а также от средств генерации разборщиков синтаксиса LEX и YACC. Заимствованные свойства были объединены и дополнены в целях получения ровно того, что требовалось для данной спецификации типов данных и создания минимального и самодостаточного языка. Кроме того, коль скоро основным назначением данного языка является определение типа данных, была сделана попытка обойтись без встроенных типов данных.

[9] Нетрудно видеть, что ключевое слово type указано вместо ключевого слова keyword, используемого при описании интерфейсов на языках IDL и Java и классов на языках C++ и Java. Конструкция alias уникальна для настоящей спецификации и введена для того, чтобы можно было добавить весьма краткую мнемонику к более описательным именам. Конструкция specializes предпочтительнее двоеточия, используемого в языках C++ и IDL, поскольку ее значение более очевидно.

[10] Обратите внимание, что понятия входных и выходных аргументов, используемые в языке IDL, а также понятия возвращаемых значений и исключений, используемые в языках IDL, JAVA и C++, не уместны для настоящей спецификации. Семантика типов данных не имеет отношения к вызовам процедур, передаче параметров или нормального и аварийного возвращения управления из тела процедуры. Вместо этого каждое семантическое свойство можно концептуально рассматривать как функцию, отображающую значение и необязательны аргумента на другое значение. Это отображение не «вычисляется» и не «генерируется», оно существует на логическом уровне и для актуализации отображения не требуется «вызывать» такую функцию.

Конструкция **specializes** означает (а) наследование свойств от рода к виду, (б) возможность подстановки значений видового типа в переменные родового типа. Эта конструкция может включать в себя определение дополнительных свойств и спецификацию ограничений свойств, наследуемых специализированным типом.

Приведем следующий пример наследования: коль скоро тип данных CS является специализацией типа данных CD, а тип данных CD имеет свойство code, то тип данных CS также имеет это свойство code, даже если свойство isNull не указано явно в объявлении свойства BL. Справедлива также следующая возможность подстановки: если свойство имеет тип данных CD, а тип данных CS является специализацией типа данных CD, то значение этого свойства может иметь тип данных CS. Другими словами, возможность подстановки означает, что все значения типа данных CS являются также значениями типа данных CD[11)].

Объявление **type** может иметь в качестве квалификаторов ключевые слова **abstract** (абстрактный), **protected** (защищенный) или **private** (приватный). Абстрактный тип представляет собой тип данных, у которого ни одно значение не может иметь этот тип данных и при этом не принадлежать к его конкретной специализации. Защищенный тип представляет собой тип данных, который используется внутри данной спецификации, но при этом не может быть присвоен никакому свойству вне этой спецификации. Приватный тип представляет собой внутреннюю абстракцию «подсказки». Он определяется только для целей определения некоторого аспекта семантики типа данных, но не используется как тип другого защищенного или публичного свойства[12)]. (Квалификатор private используется здесь с единственной целью. Приватные типы определены только для формального определения других типов данных и ни в каком виде не используются вне этой спецификации).

B.1.9.2 Инвариантное утверждение

Объявление семантических свойств, их имен, типов данных и аргументов представляет собой только ключ к пониманию назначения типа данных. Истинное определение заключается в инвариантных утверждениях, являющихся логическими утверждениями, истинными в любое время.

В настоящей спецификации инвариантные утверждения представлены с использованием формального синтаксиса, а также на естественном языке. Преимуществами формального синтаксиса являются однозначность его интерпретации и строгое типизирование. А утверждения на естественном языке более доступны для понимания, особенно тем лицам, кто не натренирован в чтении утверждений на формальных языках.

Формальный синтаксис помогает придать убедительность настоящей спецификации. Однако в некоторых случаях полная семантика типа не может быть полностью выражена с помощью таких инвариантных утверждений. Сочетание естественного и формального языка помогает сделать настоящую спецификацию более точной.

Инвариантные утверждения образуются с помощью ключевого слова **invariant**, которое объявляет одну или несколько переменных в том же формате, что у списка аргументов свойства. Инвариантное утверждение может содержать конструкцию **where**, ограничивающую аргументы для всего тела утверждения. Тело инвариантного утверждения заключено в фигурные скобки. Оно содержит список высказываний, которые все должны быть истинными.

```
invariant(BL x)
    where x.nonNull {
  x.and(true).equal(x);
};
```

Инвариантное утверждение имеет семантику логического предиката с квантором общности («для всех»).

Приведенное выше инвариантное утверждение может быть прочитано на естественном языке следующим образом: «Для всех значений x типа Boolean, где x не является пустым, справедливо высказывание «x AND *true*

---

11) Ограничивающий аспект специализации заслуживает разъяснений. Обычно утверждается, что наследование на должно отказываться от каких-либо свойств, определенных для рода. Это действительно так для ограничения, поскольку при этом от свойства не отказываются, но ограничивают его значения меньшим множеством. Но если свойство родительского класса может быть пустым (NULL), то ограничение может состоять в том, что оно всегда пустое. В любом случае логически ограничение является специализацией с наследованием и возможностью подстановки.

12) Обратите внимание, что значение ключевого слова protected несколько отличается от квалификаторов доступности (public, package, protected, private), используемы в языках JAVA и C++. Оно используется здесь не как признак скрытия информации об этом типе или исключения свойств защищенного типа из доступа вне настоящего «пакета» спецификаций. Его надо рассматривать как настоятельную рекомендацию не объявлять атрибуты или другие свойства таких защищенных типов. Защищенные типы должны использоваться как «встроенные» в другие типы данных. Защищенный тип непосредственно доступен внутри «обертки», в которую они встроены. Понятие «делегированных свойств» отсутствует.

равно x». Все свойства должны быть именованными, чтобы высказывания можно было читать как предложения на естественном языке[13].

Если для инвариантного утверждения аргументы не требуется, то список аргументов указывать в нем не надо.

```
invariant {
  true.not.equal(false);
  false.not.equal(true);
};
```

B.1.9.2.1 Выражение высказывания

Высказывания, включенные в инвариантные утверждения, представляют собой выражения, аргументами которых являются семантические свойства определяемых типов данных. Выражение высказывания должно иметь значение типа Boolean («true» или «false»)[14]. Никакие примитивные типы данных или операции не существуют до определения типов данных. Единственными предопределенными свойствами языка выражений высказываний являются[15]:

- строки символов, представляющие словесные выражения языка определения типов данных;
- понятие справедливости высказывания (*true*) или его ложности (*false*);
- инвариантное утверждение: invariant(...) where ... {...};
- выражение квантора общности в форме forall (...) where ... {...}; является синонимом инвариантного утверждения;
- выражения квантора существования в форме exists (...) where ... {...};
- явная конъюнкция (логический оператор AND) между утверждениями: утверждение_1; утверждение_2; ... утверждение_n; разделенными точками с запятой;
- переменные и объявления в списке аргументов инвариантного утверждения;
- ссылка на свойство с помощью точки: x.property;
- явное и неявное преобразование типов: (T)x;
- круглые скобки для переопределения приоритетов операторов преобразования и разрешения свойств: (T) x.property по сравнению с ((T)x).property.

B.1.9.2.2 Выражения с вложенными кванторами

По аналогии с инвариантными выражениями внутри выражения высказываний могут быть указаны вложенные кванторы. В действительности квантор общности, записанный с помощью ключевого слова forall, является ничем иным, как инвариантным выражением. Как показано в следующем примере, квантор общности может использоваться во вложенном выражении, если это требует сложность проблемы:

```
invariant(SET<T> x, y)
    where x.nonNull {
  x.subset(y).equal(
    forall(T element) where x.contains(element) {
```

---

[13] Синтаксис и семантика инвариантного учреждения похожи на конструкцию «inv» в языке OCL. Однако этот язык не используется в настоящей спецификации по следующим причинам: (1) стиль синтаксиса языка OCL напоминает Smalltalk, отличающихся от стиля определения типов данных в языках C++/Java; (2) в языке OCL предусмотрено много примитивных конструкцию и типов данных, чего настоящая спецификация старается по возможности избежать; (3) богатство примитивных конструкций в языке OCL является одной из причин его сложности, не требуемой в настоящей спецификации.

[14] Эта конструкция в определенном отношении является циклической; она исходит из существования типа данных Boolean, хотя этот тип данных сам определен таким же образом, как и другие типы данных. Кроме того, поскольку данный язык определения типов данных представлен в виде строк символов, понятие строки символов должно существовать до определения строкового типа данных character string. Таким образом, эти два типа, character string и Boolean, являются особыми, но на первый взгляд они определены аналогично другим типам данных. Поскольку не предполагается, что данная спецификация типов данных будет реализована, такая цикличность не представляет собой особую проблему. Даже если бы этот язык предполагалось реализовать, можно было бы использовать технологию «самозагрузки», которая вполне обычна, к примеру, в компиляторах, которые компилируют сами себя.

[15] Большинство этих синтаксических свойств навеяны языком JAVA: использование списков аргументов, фигурные скобки для окаймления блоков, точка с запятой для завершения утверждения и точка для ссылки на свойства значения. Удвоенное двоеточие «::», используемое в языках C++ и IDL для различения ссылки на члена и ссылки на значение, здесь не используется (как и в языке Java). В отличие от языка Java и ближе к языкам C++ и IDL каждое утверждение завершается точкой с запятой, в том числе декларации типа. Неявное преобразование типов также позаимствовано у языка C++.

```
      y.contains(element);
      });
};
```

Квантор существования имеет то же значение, что и в общей логике высказываний. Например, следующее инвариантное выражение означает: «Значения x и y типа SET (множество) пересекаются в том и только том случае, когда существует элемент e, принадлежащий обоим множествам x и y.

```
invariant(SET x, y)
    where x.nonNull {
  x.intersects(y).equal(
    exists(T e) {
      x.contains(e);
      y.contains(e);
      });
};
```

В кванторе существования может использоваться конструкция where, однако нет разницы в том, сделано ли высказывание в конструкции where или в теле квантора существования. Напротив, в кванторах общности конструкция where ослабляет высказывание, поскольку при наличии этой конструкции тело применяется только к тем значениям, которые удовлетворяют критерию, указанному в конструкции where.

B.1.9.3 Преобразование типов

В настоящей спецификации определены некоторые допустимые преобразования типов данных. Например, для типов данных ST (строка символов) и ED (инкапсулированные данные) существует пара преобразований. Это означает, что если ожидается значение типа ED, но получено значение типа ST, то можно преобразовать значение типа ST в значение типа ED[16].

Определены три вида преобразования типов данных: повышающее приведение (promotion), понижающее приведение (demotion) и литералы строк символов. Преобразования типов могут быть явными и неявными. Неявное преобразование типов имеет место, если ожидается значение определенного типа (например, в аргументе выражения), но в действительности получено значение другого типа. Если тип полученного значения допускает преобразование в ожидаемый тип, то преобразование должно быть сделано неявным образом.

Примечание — В спецификации реализуемой технологии должно быть указано, каким образом поддерживается неявное преобразование типов данных. В некоторых технологиях неявное преобразование поддерживается, в других нет; в любом случае должны быть описаны правила обработки, указывающие, как такие преобразования реализуются.

Явное преобразование может быть задано в выражении утверждения, указав имя целевого типа данных в скобках перед преобразуемым значением. Например, ниже показано явное преобразование типа данных в конструкции where инвариантного выражения:

```
invariant(ED x)
  where ((ST)x).nonNull { ... };
```

Преобразование типа имеет более низкий приоритет по сравнению с точкой, используемой для указания свойства. Так, «(T)a.b « задает преобразование значения свойства b переменной a в тип данных T, в то время как «((T)a).b « задает сначала преобразование значения переменой a в тип данных T, а затем ссылку на свойство b результата преобразования.

Неявное преобразование типа данных в выражениях высказываний осуществляется при возможности. Пусть объявлено, что формальный аргумент свойства имеет тип данных T. Если в выражении он используется как фактический аргумент типа U; и при этом тип данных U не является расширением типа T, то в случае, если для типа данных U определено преобразование в тип данных T, выполняется преобразование из типа данных T в тип данных U.

B.1.9.3.1 Понижающее приведение

Понижающее приведение (demotion) представляет собой преобразование с потерей информации. В общем случае это означает, что более сложный тип преобразуется в более простой.

---

[16] Это означает, что если некто ожидает значение типа ED, но вместо него получил значение типа ST, то он может преобразовать значение типа ST в значение типа ED.

Примером понижающего приведения может служить преобразование значения, имеющего тип данных интервала (IVL), в простое количество (тип данных QTY), например, в центр интервала. В языке определения типов данных понижающее приведение объявляется с помощью ключевого слова **demotion** и имени результирующего типа данных:

```
type Interval alias IVL {
  ...
  demotion  QTY;
  ...
};
```

В спецификации понижающего приведения должно быть указано, какая информация теряется и каковы главные последствия потери информации.

B.1.9.3.2 Повышающее приведение

Повышающее приведение (promotion) представляет собой преобразование, при котором генерируется новая информация. В общем случае это означает, что более простой тип преобразуется в более сложный.

Например, можно преобразовать значение количества, имеющее тип данных (QTY), в значение интервала (тип данных IVL). Однако у типа данных IVL имеются дополнительные семантические свойства по отношению к типу данных QTY, а именно, нижняя и верхняя граница. Таким образом, преобразование из типа данных QTY в тип данных IVL является повышающим приведением. Дополнительным свойствам типа данных QTY, отсутствующим у типа данных IVL, должны быть присвоены новые значения (значения по умолчанию или вычисляемые значения). В спецификации повышающего приведения должно быть указано, каковы будут новые значения и как они будут генерироваться.

Повышающее приведение типа данных QTY к типу данных IVL объявляется с помощью ключевого слова promotion и имени результирующего типа данных:

```
type Quantity alias QTY {
  ...
  promotion  IVL;
  ...
};
```

Обычно повышающее приведение определяется для преобразования более простого типа данных в более сложный тип. Обычно также простой тип объявляется в настоящем документе раньше более сложного. Объявление всех повышающих приведений в простом типе потребовало бы ссылок вперед по тексту, что неудобно читателю. Поэтому альтернативный синтаксис разрешает определять повышающее приведение в объявлении более сложного типа. Оно обозначается с помощью указания имени типа, к которому применяется повышающее приведение, в списке аргументов после имени результирующего типа:

```
type Interval alias IVL {
  ...
  promotion  IVL (QTY x);
  ...
};
```

B.1.9.4 Литеральная форма

Литерал представляет собой представление значения данных в виде строки символов. Литералы определены для многих типов данных. Литерал представляет собой преобразование типа из строкового типа данных ST и обратно с помощью специального определенного синтаксиса.

Не каждое преобразование в тип данных ST и обратно является литеральным преобразованием. Литерал типа данных должен обеспечивать представление всего множества значений этого типа данных, в то время как другие преобразования в тип данных ST и обратно могут отображать только меньшее подмножество значений преобразуемого типа данных.

Целью использования литералов является представление значений в краткой человеко-читаемой форме. Например, литеральные представления целых чисел (тип данных INT) и вещественных чисел (тип данных REAL) являются строками, состоящими из знака числа, цифр, необязательной десятичной точки и т. д. Более важные типы интервалов (IVL<REAL>, IVL<PQ>, IVL<TS>) имеют литеральные представления наподобие «<5», означающего «меньше 5», что гораздо удобнее для чтения, нежели полностью структурированная форма интервала. Для некоторых более сложных типов данных, например, интервалов, общей спецификации периодичности и пара-

метрического распределения вероятности литеральная форма представления значений может оставаться единственной до тех пор, пока пользователь не привыкнет к используемой концептуализации.

Каждое литеральное преобразование имеет свой собственный синтаксис (грамматику), который может быть не очень простым для компьютерной обработки[17].

Примечание — Спецификация реализуемой технологии, использующая строковое представление значений этих абстрактных типов данных, может использовать, а может и не использовать литеральные представления, приведенные в настоящем документе. Ожидается, что спецификация реализуемой технологии на языке XML будет использовать не все, но некоторую часть определенных здесь представлений.

B.1.9.4.1 Объявление

На языке определения типов данных литерал объявляется как свойство типа данных, используя ключевое слово literal, за которым следует имя типа данных ST, поскольку литерал представляет собой преобразование в тип данных ST и обратно:

```
type IntegerNumber alias INT {
  ...
  literal  ST;
  ...
};
```

B.1.9.4.2 Определение

Фактическое определение литеральной формы осуществляется вне тела объявления типа данных, используя атрибутивную грамматику. Такая грамматика определяет как семантику, так и синтаксис структур языка. Определенный в ней синтаксис по существу является формой Бэкуса-Наура BNF (Backus-Naur-Form)[18].

К примеру, рассмотрим следующее простое определение типа данных порядковых чисел (положительных целых чисел). Определение этого типа данных зависит только от типа данных Boolean (BL) и содержит объявление литерала строкового типа:

```
type CardinalNumber alias CARD {
  BL      isZero;
  BL      equal(ANY x);
  CARD    successor;
  CARD    plus(CARD x);
  CARD    timesTen;
  literal ST;
};
```

Ниже полностью представлены синтаксис и семантика литерала, а затем его детальное описание.

```
CARD.literal ST {
  CARD : CARD digit   { $.equal($1.timesTen.plus($2); }
       | digit        { $.equal($1); };

  CARD digit  : "0"   { $.isZero; }
              | "1"   { $.equal(0.successor); }
```

[17] Наличие различных грамматик литералов не означает, что они будут объединены в одну общую грамматику для представления значений в стандартах HL7. Хотя и можно предпринять попытки разрешения неоднозначности для литералов разных типов, они могут оказаться опасными, поскольку некоторые из неоднозначностей останутся. Например, значение «1.2» является допустимым литералом как для объектного идентификатора (тип данных OID), так и для вещественного числа.

[18] Используемый здесь вариант формы BNF близок к тем, что используются в языках синтаксического разборщика YACC и генератора лексического анализатора LEX, но он упрощен и сделан совместимым с синтаксисом и стилем объявлений, принятыми в данном языке определения типов данных. Отличия состоят в том, что все символы имеют ровно один атрибут, их значение строго типизировано с использованием одного из определенных типов данных. Тип каждого символа объявлен перед определением символа (например, INT digit : «0» | «1» | ... | «9»;). Начальный символ представляет собой не имя, а тип (например, INT : digit | INT digit;). Имя типа данных может служить именем символа, означающего литерал этого типа данных.

```
                | "2"  { $.equal(1.successor); }
  ...
                | "8"  { $.equal(7.successor); }
                | «9» { $.equal(8.successor); }
};
```

Каждое синтаксическое правило состоит из имени символа, двоеточия и определения символа (называемого продукцией). Продукция представляет собой последовательность символов. Эти другие символы либо также определены в грамматике, либо являются терминальными символами, представляющими собой строки символов, заключенные в двойные кавычки, либо шаблоны строк (называемые регулярными выражениями). Таким образом, запись

```
CARD : CARD digit
       | digit;
```

означает, что любой символ порядкового номера является либо порядковым номером, за которым следует цифра, либо просто цифрой. Вертикальная черта означает логическое «ИЛИ» (OR). Синтаксическое правило завершается точкой с запятой.

Каждый символ имеет ровно одно значение определенного типа данных. Тип данных значения символа объявляется в его определении. Запись

```
CARD digit : «0»
             | «1»
             | «2»
             | ...
             | «8»
             | «9»;
```

означает, что символ digit имеет значение типа CARD. Начальный символ является именем типа данных, отдельного имени не требуется.

Семантика литерального выражения описана семантическими правилами, заключенными в фигурные скобки. Эти правила задаются для каждой определенной продукции символа:

```
символ : продукция1 { правило1 } | продукция2 { правило 2 } | ... | продукцияn { правило n };
```

Семантическое правило представляет собой простой список булевских выражений высказывания, разделяемых точкой с запятой. Эти выражения имеют тот же вид, что и выражения высказывания, используемые в инвариантных выражениях. Однако в семантическом правиле могут быть определены специальные переменные, имена которых начинаются со знака доллара (например, $, $1, $2, $3, ...). Одинокий знак $ означает значение текущего определенного символа, а $1, $2, $3 и т. д. означают значения частей продукции, ассоциированной с семантическим правилом. Например, в записи

```
CARD : CARD digit  { $.equal($1.timesTen.plus($2); }
       | digit          { $.equal($1); };
```

с первой продукцией «CARD digit» связано семантическое правило, которое гласит: «значение определенного символа ($) равно значению $1 первого символа CARD, умноженному на 10 (timesTen), сложенному (plus) со вторым символом digit»[19].

В качестве терминального символа может быть указан шаблон строки, так называемое регулярное выражение. Здесь использован классический синтаксис регулярных выражений, придуманный Aho и используемый в AWK, LEX, GREP и PERL. Регулярные выражения указывают между двумя косыми чертами /.../. В регулярном выражении каждый символ, кроме [ ] ^ $. / : ( ) \ | ? * + { }, совпадает сам с собой. Другие символы, фактически используемые в настоящей спецификации, приведены в таблице В.2.

---

[19] Учтите, что свойство equal (равенство), определенное для всех типов данных (см. equal) является отношением, а именно, проверкой на равенство, а не оператором присваивания. Значению нельзя присвоить значение. В отличие от анализаторов YACC и LEX, данный язык определения типов данных является чисто декларативным и понятие присваивания в нем отсутствует. Поэтому грамматические правила определяют литеральные выражения как для разбора, так и для конструирования.

Таблица В.2 — Специальные символы для регулярных выражений

Шаблон	Определение
[...]	Описывает класс символов. Например, /[A-Za-z]/ совпадает с прописными и строчными буквами английского алфавита
[^ ...]	Задает исключение из класса символов. Например, /[^BCD]/ совпадает с любым символом, кроме B, C и D
...?	Предшествующий шаблон не обязателен. Например, /ab?c/ совпадает и с «ac», и с «abc»
...*	Предшествующий шаблон может повторяться от нуля до любого числа раз. Например, /ab*c/ совпадает с «ac», «abc», «abbc», «abbbc» и т. д.
...+	Предшествующий шаблон может повторяться от одного до любого числа раз. Например, /ab+c/ совпадает с «abc», «abbc», «abbbc», но не с «ac»
... {n,m}	Предшествующий шаблон может повторяться от n до m раз, где n и m порядковые числа, удовлетворяющие условию 0 < n < m. Например, /ab{2,4}c/ совпадает с «abbc», «abbbc» и «abbbbc»
... | ...	Совпадение может быть с любым из шаблонов, разделенных вертикальной чертой. Например, /ab|cd/ совпадает с «abd» и»acd», но не с «abcd»
(...)	Шаблон в скобках используется в описанных выше операторах как единое целое. Например, /a(bc)*/ совпадает с «a», «abc», «abcbc», «abcbcbc» и т. д.
... : ...	Совпадение с левым шаблоном будет в том случае, если за ним следует правый шаблон, но при этом правый шаблон не участвует в проверке совпадения. Например, /ab:c/ совпадает с «abc», но не с «ab», однако значением совпадающего символа будет «ab», а «c» остается для следующего символа. Двоеточие представляет собой небольшую модификацию традиционной косой черты, но косая черта также традиционно используется для завершения всего шаблона и может также встретиться как совпадающий символ — три разных значения одного символа были бы уже слишком
... \ ...	Буквальное совпадение со следующим символом, то есть устраняет любое специальное значение этого символа. Например, /a\+b/ совпадает с «a+b»
... V ...	Совпадение с косой чертой как с символом. Например, /aVbc/ совпадает с «a/bc»

В.1.9.5 Параметризованные типы данных

Параметризованные типы данных имеют неполные определения. Эта неполнота обозначается наличием одного или нескольких параметров в определении типа. Обычно в качестве параметров выступают другие типы данных. С помощью параметров параметризованный тип может объявить семантические свойства других не полностью определенных типов данных. Например, параметризованный тип данных Interval объявлен с параметром *T*, значение которого может быть любым типом данных физической величины QTY. Его компоненты *low* и *high* объявлены как имеющие тип данных *T*:

```
template<QTY T>
type Interval<T> alias IVL<T> {
  T low;
  T high;
};
```

Создание экземпляра параметризованного типа данных завершает его определение. Например, для создания экземпляра интервального типа данных (Interval), необходимо указать базовый тип данных интервала. Это делается с помощью связывания параметра *T*. Чтобы создать экземпляр интервала целых чисел, надо связать параметр *T* с типом данных Integer. После этого неполный тип данных Interval становится полным типом данных *Interval of Integer* (интервал целых чисел).

Например, в следующем определении типа данных MyType объявляется свойство с именем «multiplicity» (кратность), имеющее тип данных интервала порядкового типа данных, использованного в предыдущих примерах:

```
type MyType alias MT {
  IVL<CARD>  multiplicity;
};
```

B.1.9.5.1 Параметризованные коллекции

В настоящей спецификации активно используются параметризованный типы данных для коллекций. Среди них наиболее важными являются:

Set (SET<T>) — множество, содержащее не упорядоченные и не повторяющиеся элементы.

Sequence (LIST<T>) — последовательность, представляющая собой коллекцию значений, имеющих произвольный, но конкретный порядок. У последовательности есть голова и хвост, где головой является элемент, а хвостом — последовательность без своей головы.

Interval (IVL<T>) — интервал, представляющий собой непрерывное подмножество упорядоченного типа данных.

Эти и другие параметризованные типы данных полностью описаны в подразделе 0 «

Параметризованные типы данных». Эти параметризованные типы данных и их свойства используются в данной спецификации, начиная с ранних разделов. Для лучшего понимания настоящей спецификации необходимо иметь определенные знания о множестве, последовательности и интервале; когда речь пойдет об использовании параметризованного типа для определения другого типа данных, рекомендуется обратиться к подразделу 0 «

Параметризованные типы данных».

B.1.9.5.2 Расширения параметризованного типа данных

Расширения параметризованного типа данных являются параметризованными типами данных с одним типом параметра, специализируемым параметризованным типом. На формальном языке определения типов данных для специализации параметризованного типа используется следующий шаблон:

```
template<ANY T> type GenericTypeExtensionName specializes T {
  ...
};
```

Такие расширения параметризованного типа наследуют свойства от своего базового типа и добавляют к ним некоторые специфические особенности. Поскольку расширение параметризованного типа является специализацией базового типа, то значение, имеющее расширенный тип данных, может быть использовано вместо значения, имеющего базовый тип данных[20].

Примечание — Значения, имеющие расширенный тип, могут быть заменены значениями, имеющими его базовый тип. Однако спецификация реализуемой технологии может наложить некоторые ограничения на то, какие расширения она допускает. В частности, расширения не должны определяться для тех компонентов, в которых содержатся значения свойств значения данных. Таким образом, для любого типа данных может быть указана аннотация вне спецификации типа данных, но спецификация реализуемой технологии может не обеспечивать возможность указания аннотации значения свойства значения данных.

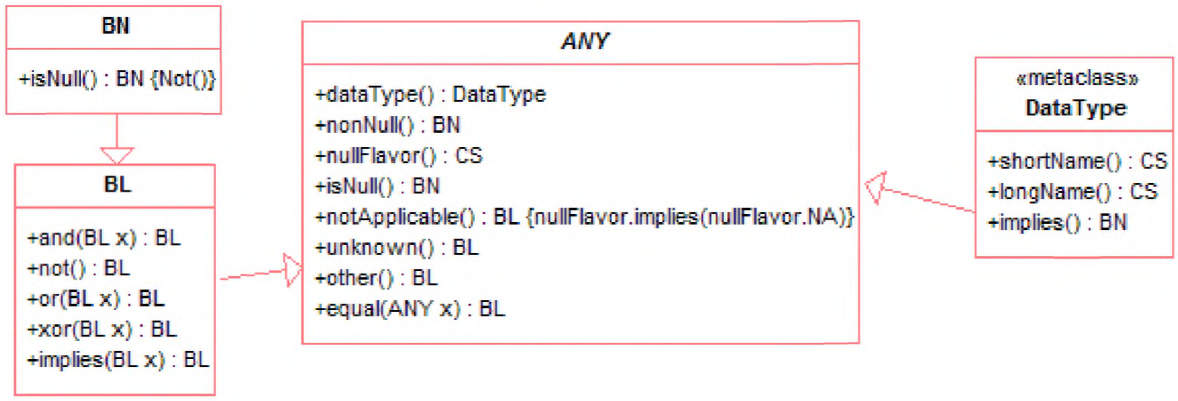


Рисунок B.2 — Фундаментальные типы данных

B.1.10 Соответствие

Если приложение получает или разбирает экземпляр, который не является допустимым по отношению к настоящей спецификации, то получатель может отклонить этот экземпляр таким способом, который он сочтет наиболее приемлемым, однако это не является обязательным. Обратите внимание, что некоторые другие стандарты

[20] Расширения параметризованного типа данных иногда называют «смесями» (mixins), поскольку их эффект состоит в примешивании некоторых свойств к уже определенному типу данных.

HL7 или такие артефакты, как объявление соответствия, могут накладывать дополнительные ограничения на поведение получателя в таких случаях.

B.1.11 Тип данных DataValue (ANY)

Этот тип данных определяет базовые свойства любого типа данных. Он является абстрактным, то есть значение не может иметь этот тип данных и не принадлежать какому-либо конкретному типу. Каждый конкретный тип данных является специализацией общего абстрактного типа данных DataValue.

```
abstract type DataValue alias ANY {
  TYPE  dataType;
  BN    nonNull;
  CS    nullFlavor;
  BN    isNull;
  BL    notApplicable;
  BL    unknown;
  BL    other;
  BL    equal(ANY x);
};
```

B.1.11.1 Свойство dataType TYPE

Это свойство идентифицирует тип данных и отражает тот факт, что каждое значение данных неявно несет информацию о своем типе данных. Таким образом, получив значение данных, можно запросить его тип данных.

```
invariant(ANY x) {
  x.dataType.nonNull;
};
```

B.1.11.2 Свойство nonNull: BN

Указывает, что данное значение является непустым, то есть допустимым для данного типа данных.

```
invariant(ANY x) {
  x.isNull.equal(x.nonNull.not);
};
```

Если свойство, атрибут модели RIM или поле сообщения объявлены обязательными и непустыми (mandatory), то любое непустое значение типа данных, к которому принадлежит свойство, должно иметь непустое значение этого свойства. Другими словами, поле не может быть пустым, коль скоро его контейнер (объект, сегмент и т. д.) должно иметь непустое значение.

B.1.11.3 Свойство isNull: BN

Указывает, что значение является исключительным, то есть пустым (NULL). Пустота означает, что информация не существует, или недоступна, или не может быть представлена в нормальном наборе значений этого типа данных.

Каждый элемент данных либо имеет допустимое значение, либо считается пустым. Если (и только если) оно пусто, то свойство nullFlavor предоставляет более детальную информацию о том, каким образом и почему у этого свойства нет правильного значения.

```
invariant(ANY x) {
  x.isNull.equal(x.nullFlavor.implies(NI));
};
```

B.1.11.4 Свойство nullFlavor: CS

Если значение является исключительным (NULL), то это свойство указывает, каким образом и почему подходящая информация отсутствует.

```
invariant(ANY x) {
  x.nonNull.equal(x.nullFlavor.isNull);
};
```

Таблица В.3 — Словарный домен свойства nullFlavor (причина пустоты)

Код	Имя	Определение
NI	NoInformation	Отсутствует какая бы то ни было информация, которую можно вывести из данного исключительного значения. Это наиболее общее исключительное значение, оно также используется по умолчанию
OTH	other	Фактическое значение не является элементом домена значений переменной (например, в используемой системе кодирования данное понятие отсутствует)
NINF	negative infinity	Отрицательная бесконечность чисел
PINF	positive infinity	Положительная бесконечность чисел
UNK	unknown	Правильное значение имеется, но оно не известно
ASKU	asked but unknown	Информация запрошена, но ответ не получен (например, пациенту задали вопрос, а ответ он не знает)
NAV	temporarily unavailable	Информация в данное время недоступна, но может стать доступной позже
NASK	not asked	Эта информация не запрашивалась (например, пациенту вопрос не задавался)
TRC	trace	Содержание отлично от нуля, но слишком мало, чтобы можно было узнать его количество
MSK	masked	Информация об этом элементе имеется, но не предоставлена отправителем по причине безопасности, конфиденциальности и т. д. Для получения доступа к этой информации могут существовать альтернативные механизмы. Хотя детали информации и не раскрываются, что предоставление такой причины пустоты может привести к нарушению конфиденциальности информа-ции. Такая причина пустоты может быть указана в ситуации, когда получателя необходимо информировать о том, что информация существует, не представляя самое информацию
NA	not applicable	В данном контексте правильное значение не существует (например, последний менструальный период у мужчины)
NP	not present	Значение не представлено в сообщении. Такая причина пустоты определена только для сообщений, а не для прикладных данных! Все значения, не представленные в сообщении, должны быть заменены на правильное значение по умолчанию или иметь признак (NI) (нет информации), используемый как умолчание для всех умолчаний

Причины пустоты представляют собой общее доменное расширение всех нормальных типов данных. Обратите внимание на отличие домена значений от словарного домена кодированных типов данных. Словарный домен является доменом кодированных значений, но не все домены значений являются словарными доменами.

Причина пустоты «other» (другое) используется, когда фактическое значение не принадлежит требуемому домену значений. Такое может быть, скажем, когда значение не удовлетворяет некоторым слишком строгим ограничениям (например, возраст должен быть меньше 100 лет).

Примечание — Причины пустоты применимы к любому свойству значения данных или атрибуту объекта более высокого уровня. Если различия в причинах пустоты не существенны, то причины пустоты можно не отражать в спецификации реализуемой технологии. Если иное не указано в настоящем документе, то спецификации реализуемой технологии может не представлять общую причину пустоты свойств значений данных.

Некоторые из причин пустоты ассоциируются с именованными свойствами, которые могут использоваться как простые предикаты для всех значений данных. Это сделано для упрощения формулирования инвариантных выражений в остальной части настоящего документа.

Запомните разницу между семантическими свойствами и «компонентами» представления значений данных. В спецификации реализуемой технологии должны быть представлены только те компоненты, которые необходимы для вывода семантических свойств. Все предикаты пустоты значения nonNull, isNull, notApplicable, unknown и other могут быть выведены из свойства nullFlavor.

В.1.11.5 Предикат notApplicable: BL

Предикат, указывающий, что причина пустоты nullFlavor данного исключительного значения равна «NA» (не применимо), то есть правильное значение не имеет смысла в данном контексте.

```
invariant(ANY x) {
  x.notApplicable.equal(x.nullFlavor.implies(NA));
};
```

B.1.11.6 Предикат unknown : BL

Предикат, указывающий, что причина пустоты nullFlavor данного исключительного значения равна «UNK» (не известно).

```
invariant(ANY x) {
  x.unknown.equal(x.nullFlavor.implies(UNK));
};
```

B.1.11.7 Предикат other: BL

Предикат, указывающий, что причина пустоты nullFlavor данного исключительного значения равна «OTH» (другое), то есть правильное значение не принадлежит требуемому домену значений.

```
invariant(ANY x) {
  x.other.equal(x.nullFlavor.implies(OTH));
};
```

B.1.11.8 Предикат equal: BL (равенство)

Равенство является рефлексивным, симметричным и транзитивным отношением между двумя значениями данных. Равенство возможно только между допустимыми значениями, пустые значения никогда не равны (даже если имеют одинаковую причину пустоты).

```
invariant(ANY x, y, z)
    where x.nonNull.and(y.nonNull).and(z.nonNull)  {
  x.equal(x);                                      /* рефлексивность */
  x.equal(y).equal(y.equal(x));                    /* симметричность */
  x.equal(y).and(y.equal(z)).implies(x.equal(z))   /* транзитивность */
  x.equal(y).implies(x.dataType.equal(y.dataType);
};
```

Способ установления равенства должен быть определен для каждого типа данных. Если иное не указано, то два значения данных равны, если они не различимы, то есть если у них нет различающихся семантических свойств. Это общее определение равенства может быть переопределено в типе данных с помощью указания собственного отношения равенства. Такое переопределение отношения равенства может быть использовано для исключения семантических свойств из проверки на равенство. Если в типе данных какие-то семантические свойства исключены из его определения равенства, это означает, что определенные свойства (или аспекты свойств), не ставшие частью проверки на равенство, не существенны для смысла значения.

Например, физическая величина имеет два семантических свойства: (1) вещественное число, (2) кодированная единица измерения. Однако при проверке на равенство необходимо учитывать тот факт, что, к примеру, 1 метр равен 100 сантиметрам. Таким образом, независимые равенства двух семантических свойств являются слишком строгим критерием равенства двух величин. Поэтому в определении типа данных физической величины необходимо переопределить отношение равенства.

B.1.12 Тип данных DataType (TYPE) (специализация типа данных ANY)

Это метатип, объявленный, чтобы можно было давать формальное определение типа данных значения. Любой тип данных, определенный в настоящей спецификации, является значением типа DataType.

```
private type DataType alias TYPE specializes DataValue {
  CS  shortName;
  CS  longName;
  BN  implies(TYPE that);
};
```

B.1.12.1 Свойство shortName: CS

Значение типа CS, содержащее краткое имя типа данных.

```
invariant(DataType x)
    where x.nonNull {
  x.shortName.nonNull;
};
```

B.1.12.2 Свойство longName: CS

Значение типа CS, содержащее полное имя типа данных.

B.1.12.3 Свойство implies : BN

Тип данных вытекает (implies) из другого типа данных, если он имеет тот же тип или является его специализацией.

B.2 Базовые типы

B.2.1 Тип данных Boolean (BL) (специализация типа данных ANY)

Тип данных *BL* описывает значения двузначной логики. Значением типа *BL* может быть или «true» (истина), или «false» (ложь), или, как и у других значений, NULL (пустое).

```
type Boolean alias BL specializes ANY
  values(true, false) {
        BL  and(BL x);
        BL  not;
  literal  ST;
        BL  or(BL x);
        BL  xor(BL x);
        BL  implies(BL x);
};
```

При наличии пустого значения NULL двузначная логика расширяется до трехзначной в соответствии с таблицей .4–.

Таблица B.4 — Таблицы истинности для булевской логики с пустыми значениями

NOT	
true	false
false	true
NULL	NULL

AND	true	false	NULL
true	true	false	NULL
false	false	false	false
NULL	NULL	false	NULL

OR	true	false	NULL
true	true	true	true
false	true	false	NULL
NULL	true	NULL	NULL

Когда булевская операция выполняется над двумя экземплярами типа данных, имеющими разные причины пустоты nullFlavor, то причиной пустоты результата является первый общий предшественник этих двух разных значений nullFlavor. Однако приложения, соответствующие настоящему стандарту, могут выбрать любого общего предшественника.

B.2.1.1 Свойство not: BL

Отрицание (not) значения типа *BL* превращает «true» в «false», «false» в «true» и NULL в NULL.

```
invariant(BL x) {
  true.not.equal(false);
  false.not.equal(true);
  x.isNull.equal(x.not.isNull);
};
```

B.2.1.2 Свойство and: BL

Конъюнкция (and) является ассоциативной и коммутативной, при этом в качестве нейтрального элемента выступает значение «true». Конъюнкция значения «false» с любым булевским значение дает результат «false». Эти правила справедливы, даже если один или оба операнда пусты (NULL). Конъюнкция двух операндов со значением NULL дает результат NULL.

```
invariant(BL x) {
  x.and(true).equal(x);
  x.and(false).equal(false);
  x.isNull.implies(x.and(y).isNull);
};
```

B.2.1.3 Свойство or: BL

Дизъюнкция x or y дает результат «false» в том и только том случае, когда x имеет значение «false» и y имеет значение «false».

```
invariant(BL x, y) {
  x.or(y).equal(x.not.and(y.not).not);
};
```

B.2.1.4 Свойство xor: BL

Исключающая дизъюнкция xor ограничивает дизъюнкцию or таким образом, что два операнда не могут одновременно иметь значение «true».

```
invariant(BL x, y) {
  x.xor(y).equal(x.or(y).and(x.and(y).not));
};
```

B.2.1.5 Свойство implies: BL

Правило импликации в форме «Если условие ТО следствие». Логические импликация определяется как дизъюнкция следствия (conclusion) и отрицания условия. Это означает, что если условие (condition) имеет значение «true», то для того, чтобы результат всего выражения имел значение «true», необходимо, чтобы следствие (conclusion) имело значение «true». Логическая импликация важна для конструирования инвариантных выражений.

```
invariant(BL condition, conclusion) {
  condition.implies(conclusion).equal(
    condition.not.or(conclusion));
};
```

Импликация не обратима и не указывает, что имеет значение «true», если условие имеет значение «false». (ex falso quodlibet — лат. «из лжи следует что угодно»).

B.2.1.6 Литеральная форма

Литеральная форма типа данных Boolean определяется именованными значениями, указанными в конструкции values, а именно, «true» и «false».

B.2.2 Тип данных BooleanNonNull (BN) (специализация типа данных BL)

Тип данных *BN* ограничивает тип данных BL таким образом, то значение этого типа данных не может быть пустым. Этот тип данных предназначен для использования в тех случаях, когда пустое значение не допустимо.

```
private type BooleanNonNull alias BN specializes BL;
};

Свойство isNull: BN
invariant (BN x) {
  x.isNull.not
};
```

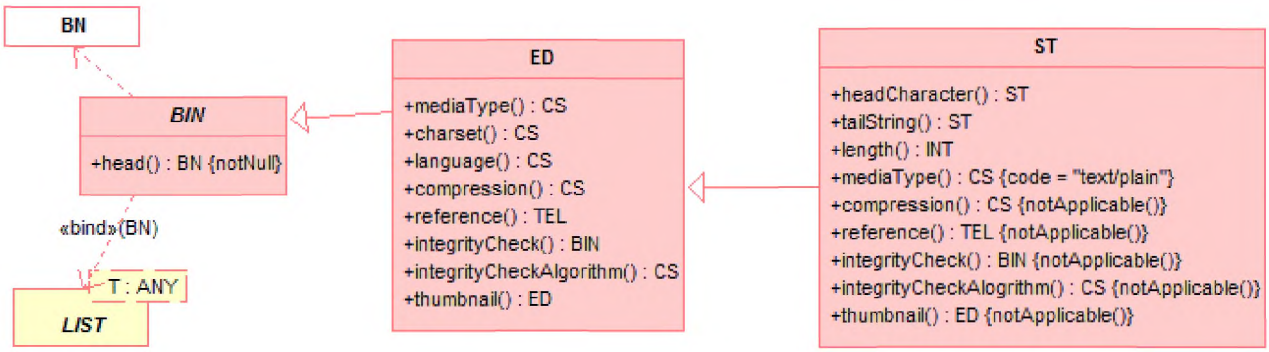


Рисунок В.3 — Обзор мультимедийных и текстовых типов данных

В.2.3 Тип данных BinaryData (BIN) (специализация типа данных LIST<BN>)

Тип данных BIN представляет собой простой блок битов. Он является защищенным, так что не должен объявляться за пределами спецификации типов данных.

Бит семантически эквивалентен непустому значение типа BL. Таким образом, семантически все двоичные данные являются последовательностями непустых значений типа BL.

```
protected type BinaryData alias BIN specializes LIST<BN>;
```

Примечание — конкретное представление произвольных двоичных данных является прерогативой спецификации реализуемой технологии. Как это в ней делается, зависит от базовой технологии реализации (символьная или двоичная) и от представляемых данных. Семантически символьные данные представляются как двоичные данные, однако символьная спецификация реализуемой технологии не должна преобразовывать символьные данные в произвольные двоичные данные, а затем эти двоичные данные преобразовывать в коды символов. В конечном счете при любой символьной технологии будут передаваться двоичные данные.

Пустая последовательность считается не двоичными данными, а пустым значением (NULL). Другими словами, непустые двоичные данные содержат хотя бы один бит. Ни один бит в непустых двоичных данных не может иметь значение NULL.

```
invariant(BIN x)
    where x.nonNull {
  x.notEmpty;
  x.length.greaterThan(0);
};
```

В.2.4 Тип данных EncapsulatedData (ED) (специализация типа данных BIN)

Данные, в основном рассчитанные на чтение человеком или на дальнейшую машинную обработку данных за пределами области применения стандарта HL7. К ним относятся неформатированный или форматированный письменный текст, мультимедийные данные, или структурированная информация, определенная другим стандартом (например, XML-подпись). Вместо самих данных значение типа *ED* может содержать только ссылку на данные (см. описание типа данных TEL). Заметьте, что тип данных ST — это специализация типа данных ED, у которой свойство mediaType имеет фиксированное значение text/plain (только текст).

Таблица В.5 — Сводка свойств типа данных EncapsulatedData

Имя	Тип	Описание
mediaType	CS	Идентифицирует тип инкапсулированных данных и метод их интерпретации или визуализации
charset	CS	Для типов данных, основанных на кодировании символов, это свойство указывает используемый набор символов и его кодировку. Набор символов должен идентифицироваться в соответствии с правилами регистрации наборов символов в организации Internet Assigned Numbers Authority (IANA) [http://www.iana.org/assignments/character-sets] и положениями документа RFC 2978 [http://www.ietf.org/rfc/rfc2978.txt]

Окончание таблицы В.5

Имя	Тип	Описание
language	CS	Для информации, основанной на символах, свойство language (язык) указывает человеческий язык текста
compression	CS	Указывает, являются ли исходные байтовые данные сжатыми и какой алгоритм сжатия использован
reference	TEL	Телекоммуникационный адрес (TEL), например, URL для HTTP или FTP, по которому можно получить в точности те же самые данные, которые могли бы быть представлены внутри значения типа данных ED
integrityCheck	BIN	Контролем целостности является короткое двоичное значение, представляющее криптографически стойкую контрольную сумму, вычисленную по двоичным данным. Это свойство предназначено для использования при передаче ссылок на данные и позволяет позже проверить, не изменились ли ссылочные данные после того, как было создано инкапсулированное значение, ссылающееся на эти данные
integrityCheck Algorithm	CS	Указывает алгоритм вычисления свойства integrityCheck[21)]. В настоящее время промышленным стандартом является алгоритм вычисления криптографически стойкой контрольной суммы Secure Hash Algorithm-1 (SHA-1). Он заменил алгоритм MD5 всего лишь пару лет назад, когда обнаружились некоторые недостатки в безопасности этого алгоритма. В настоящее время становится популярным также алгоритм SHA-256
thumbnail	ED	Сокращенное представление полных данных (эскиз), требующее гораздо меньше ресурсов, нежели полные данные, но при этом обладающее отличительными способностями полных данных. Обычно эскизы используются для ссылочных инкапсулированных данных. Они позволяют пользователю выбирать нужные данные, не загружая по ссылке их полное представление

```
type EncapsulatedData alias ED specializes BIN {
  CS  mediaType;
  CS  charset;
  CS  language;
  CS  compression;
  TEL reference;
  BIN integrityCheck;
  CS  integrityCheckAlgorithm;
  ED  thumbnail;
  BL  equal(ANY x);
};
```

Инкапсулированные данные могут быть представлены в двух формах — вложенные и ссылочные. Вложенные данные передаются или перемещаются как часть значения инкапсулированного типа данных. А ссылочные данные могут находиться в другом (дистанционно удаленном) месте. Независимо от формы данные являются теми же самыми.

В.2.4.1 Свойство mediaType: CS

Идентифицирует тип инкапсулированных данных (тип среды) и метод их интерпретации или визуализации.

Свойство mediaType является обязательным непустым, то есть любой непустой экземпляр типа данных ED должен иметь непустое свойство mediaType.

[21)] В настоящее время промышленным стандартом является алгоритм вычисления криптографически стойкой контрольной суммы Secure Hash Algorithm-1 (SHA-1). Он заменил алгоритм MD5 всего лишь пару лет назад, когда обнаружились некоторые недостатки в безопасности этого алгоритма. В настоящее время по умолчанию для вычисления контрольной суммы используется алгоритм SHA-1. Становится популярным также алгоритм SHA-256.

```
invariant(ED x)
    where x.nonNull {
  x.mediaType.nonNull;
};
```

Организация IANA определяет домены типов среды в стандартах Интернет RFC 2045 [http://www.ietf.org/rfc/rfc2045.txt] и RFC 2046 [http://www.ietf.org/rfc/rfc2046.txt]. Согласно RFC 2046, тип среды состоит из двух частей:

- верхнеуровневый тип среды,
- подтип среды.

Однако эта спецификация трактует полный тип среды как единую строку, формат которой определен организацией IANA, а именно, имя верхнего уровня, за которым следует символ косой черты «/», а за ним — имя подтипа среды. Актуальные зарегистрированные типы среды хранятся в базе данных организации IANA, к которой можно получить доступ по ссылке [http://www.iana.org/assignments/media-types/index.html]. В настоящее время определено свыше разных 160 типов среды MIME и этот список быстро растет. В принципе могут использоваться все типы среды, определенные организацией IANA.

Для обеспечения интероперабельности в настоящем документе некоторые типы среды объявляются более предпочтительными. Это позволяет задать наибольший общий знаменатель, при котором интероперабельность не только возможна, но и может быть достаточно эффективной для обеспечения достаточно развитых потребностей в передаче мультимедийных данных.

В таблице В.6 описан статус некоторых типов среды MIME, который формулируется в следующих терминах:

- **обязательный:** каждое приложение. соответствующее стандарту HL7 и рассчитанное на определенный вид среды, должно как минимум обрабатывать обязательные типы среды. Для каждого вида среды существует один обязательный тип среды. Некоторые типы среды предназначены для специальных целей, что обозначено формулировкой «требуется для ...»;
- **рекомендованный:** другие типы среды рекомендуются для конкретных целей. Для каждой цели должно быть указано совсем немного дополнительных рекомендованных типов среды, при этом обоснование, условия применения и предположения об использовании этих типов должны быть как можно более ясными;
- **индифферентный:** такой статус означает, что комитет HL7 не запрещает, но и не одобряет использование этого типа среды. По умолчанию все типы среды, не указанные в таблице В.6, считаются индифферентными. Поскольку для большинства практических сценариев выделены один обязательный и несколько рекомендованных типов среды, то к применению индифферентных типов среды надо относиться очень консервативно.
- **запрещенный:** запрещенные типы среды не должны использоваться, поскольку либо им присущи недостатки, либо имеются более приемлемые альтернативы, либо их применение влечет за собой определенные риски. К последним могут относиться риски нарушения безопасности, скажем, потенциальный риск переноса компьютерных вирусов в данных этого типа среды. Однако не всякий тип среды, имеющий недостатки или подверженный риску, считается запрещенным. Типы среды, не упомянутые в таблице В.6 и по умолчанию считающиеся индифферентными, также могут обладать подобными недостатками.

Таблица В.6 — Домены типов среды

Код	Имя	Статус	Определение
text/plain	Неформатированный текст	Обязательный	Любой неформатированный текст. Этот тип среды используется по умолчанию. Он эквивалентен строковому типу данных (ST)
text/x-hl7-ft	Текст HL7	Рекомендованный	Этот тип среды предназначен для совмести-мости с типом данных FT, определенном в HL7 v2.x. Рекомендуется использоваит его только для обратной совместимости с систе-мами, где реализован стандарт HL7 v2.x
text/html	Текст HTML	Рекомендованный	Текст, размеченный в сответствии со спецификацией языка разметки гипертекста HTML (Hypertext Mark-up Language). Разметка HTML достаточна для представления боль-шинства типографских документов. Язык HTML является платформенно-независимым и широко используется

Продолжение таблицы В.6

Код	Имя	Статус	Определение
application/ pdf	PDF	Рекомендованный	Формат переносимых документов PDF (Portable Document Format) рекомендован для сверстанного форматированного текста, предназначенного только для чтения. Этот формат имеет открытую спецификацию, является платформенно-независимым, широко распространенным. Существуют свободно распространяемые средства для создания и отображения данных, представленных в этом формате
text/xml	Текст XML	Индифферентный	Предназначен для структурированных строковых данных. Существует определенный риск, что общий формат документов SGML/XML слишком сложен для совместного использования различными приложениями
text/rtf	Текст RTF	Индифферентный	Формат Rich Text (RTF) широко используется для совместного использования документов, подготовленных процессорами текстов. Однако у него есть проблемы совместимости, поскольку он существенно зависит от используемого процессора текстов. Может быть полезен для совместной работы с текстом, подготовленным разными процессорами текстов
application/ msword	MSWORD	Запрещенный	Этот формат весьма подвержен проблемам совместимости. При необходимости совместного редактирования текста вместо него следует использовать обязательный формат text/plain или форматы text/html и text/rtf
audio/basic	Базовый формат аудиозаписи	Обязательный	Формат одноканального звука, закодированного в соответствии со спецификацией 8-бит правила мю ISDN [PCM] с частотой оцифровки 8000 Гц. Этот формат стандартизован организацией CCITT в документе Fascicle III.4 -Recommendation G.711. Pulse Code Modulation (PCM) of Voice Frequencies. Geneva, 1972
audio/mpeg	MPEG audio layer 3	Обязательный	MPEG-1 Audio layer-3 (MP3) представляет собой алгоритм сжатия и формат файла, определенные в стандартах ИСО 11172-3 и ИСО 13818-3. MP3 позволяет использовать настраиваемую частоту оцифровки от телефонии с сильным сжатием до звука с качеством компакт-диска CD
audio/ k32adpcm	K32ADPCM Audio	Индифферентный	Формат ADPCM позволяет сжимать аудиоданные. Он определен в спецификации Интернет RFC 2421 [ftp://ftp.isi.edu/in-notes/rfc2421.txt]. Насколько широко он реализован, не известно
image/png	Изображение в формате PNG	Обязательный	PNG (Portable Network Graphics) [http://www.cdrom.com/pub/png] представляет собой широко используемый стандарт сжатия изображений без потерь. Доступны приложения с открытым кодом, обеспечивающие обработку изображений в формате PNG
image/gif	Изображение в формате GIF	Индифферентный	GIF представляет собой широко используемый популярный формат хранения изображений. Однако он отягощен патентами[22)], поэтому к его использованию надо подходить осторожно

[22)] Согласно https://ru.wikipedia.org/wiki/GIF от 16.09.2014, срок действия последнего патента на GIF истек 11 августа 2006 года. — Прим. перев.

Окончание таблицы В.6

Код	Имя	Статус	Определение
image/jpeg	Изображение в формате JPEG	Обязательный	Этот формат обязателен для применения выского сжатия к фотографией с широкой цветовой гаммой. Он обеспечивает сжатие с потерями, но различие от сжатия без потерь визуально почти не отличимо
application/ dicom	DICOM	Рекомендованный	Тип среды MIME для формата DICOM (Digital Imaging and Communications in Medicine — цифровые изображения и связь в медицине) определен в документе RFC3240 [http://ietf.org/rfc/rfc3240.txt]
image/g3fax	Изображение в формате G3Fax	Рекомендованный	Рекомендуется только для факсимильных изображений
image/tiff	TIFF Image	Индифферентный	Хотя формат TIFF (Tag Image File Format — формат растровых графических изображений) и является международным стандартом, на практике с ним связано большое число проблем интероперабельности. У него слишком много версий, обрабатываемых не всеми программами
video/mpeg	MPEG видео	Обязательный	Международный стандарт MPEG широко используется. Он весьма эффективен для видеосигнала с широкой цветовой гаммой. Доступны приложения с открытым кодом. Степень интероперабельности очень высокая
video/x-avi	X-AVI видео	Запрещенный	Формат файлов AVI представляет собой всего лишь обертку для многих различных кодеков. Он является источником большого числа проблем интероперабельности
model/vrml	VRML Model	Рекомендованный	Это открытый стандартизованный формат трехмерных моделей, который может использоваться для приложений виртуальной реальности, например, при проведении анатомических или биохимических исследований (визуализации пространственной структуры макромолекул)

Перечень обязательных типов среды весьма мал, поэтому на приложения, соответствующую стандарту HL7, в особенности на унаследованные системы, невыполнимых требований не накладывается. В общем случае от таких приложений не требуется поддержки любого другого типа среды, отличающегося от неформатированного текста. Например, многим системам не нужны аудиоданные, поскольку они могут отобразить своим пользователям только неформатированный текст. В заявлении о соответствии таких приложений вполне можно указать, что они не обрабатывают аудиозапись. И только если заявлено об обработке аудиозаписи, такое приложение должно обеспечивать обработку обязательного типа среды для аудиозаписи.

B.2.4.2 Свойство charset: CS

Для типов кодирования символов это свойство указывает используемый набор символов и используемую кодировку. Набор символов должен идентифицироваться в соответствии с правилами регистрации наборов символов в организации Internet Assigned Numbers Authority (IANA) [http://www.iana.org/assignments/character-sets] и положениями документа RFC 2978 [http://www.ietf.org/rfc/rfc2978.txt].

Домен наборов символов ведется организацией Internet Assigned Numbers Authority (IANA) [http://www.iana.org/assignments/character-sets]. В базе данных IANA указаны имена и различные псевдонимы для большинства наборов символов. Для целей стандартов HL7 использование нескольких различных имен одного и того же набора символов не допускается. В них должно использоваться то имя, которое помечено организацией IANA как «предпочтительное для MIME.» Если ни один из псевдонимов не имеет такой пометки, то для стандартов HL7 должно использоваться основное имя набора символов.

В таблице B.7 D:\ТК 468\Стандарты ISO\27932\ГОСТ Р ИСО 27932\infrastructure\datatypes\datatypes.htm - domain-Charset перечислены некоторые из наборов символов, определенных организацией IANA и представляющих интерес для текущих членов комитета HL7.

Таблица В.7 — Домен наборов символов

Код	Имя	Определение
EBCDIC	EBCDIC	Использование этого набора символов индифферентно для стандартов HL7
ISO-10646-UCS-2	ISO-10646-UCS-2	Запрещен для использования в стандартах HL7
ISO-10646-UCS-4	ISO-10646-UCS-4	Запрещен для использования в стандартах HL7
ISO-8859-1	ISO-8859-1	Использование этого набора символов индифферентно для стандартов HL7
ISO-8859-2	ISO-8859-2	Использование этого набора символов индифферентно для стандартов HL7
ISO-8859-5	ISO-8859-5	Использование этого набора символов индифферентно для стандартов HL7
JIS-2022-JP	JIS-2022-JP	Использование этого набора символов индифферентно для стандартов HL7
US-ASCII	US-ASCII	Обязателен для использования в стандартах HL7
UTF-7	UTF-7	Использование этого набора символов индифферентно для стандартов HL7
UTF-8	UTF-8	Обязателен для поддержки кодировки Unicode

Примечание — Перечень, приведенный в таблице В.7, не является полным и исключительным. Национальные филиалы комитета HL7 могут предложить собственные рекомендации по наборам символов, используемых на их территории. Эти рекомендации могут добавить дополнительные наборы символов и указать иные статусы рекомендаций для наборов, перечисленных в таблице В.7.

Свойство charset должно быть известно, если значение типа ED представляет собой символьные данные в любой форме. Если данные вложены в значение, то набор символов должен быть известен. Если эти данные предоставляются по ссылке, и метод доступа не обеспечивает идентификацию набора символов, обычно возвращаемую в заголовке MIME, то эта идентификация должна быть указана как компонент типа данных *ED*.

Более полное обсуждение набора символов и сопутствующих вопросов можно найти в публикации «Character Model for the World Wide Web» [http://www.w3.org/TR/charmod/].

Свойство language: CS

Применительно к символьной информации свойство language указывает человеческий язык текста.

Необходимость указания кода языка для текстовых данных документирована в публикации RFC 2277 «IETF Policy on Character Sets and Languages» [http://www.ietf.org/rfc/rfc2277.txt]. Дополнительную базовую информацию можно найти в документе «Using International Characters in Internet Mail» [http://www.imc.org/mail-i18n.html] (меморандум консорциума Internet Mail Consortium).

Принципы, положенные в основу домена значений этого свойства изложены в стандарте Интернет RFC 3066 [http://www.ietf.org/rfc/rfc3066.txt]. Система кодирования, описанная в этом документе, конструируется из основного компонента, содержащего тег с кодом языка, определенным в стандарте ИСО 639, и двух кодов расширения языков, не предусмотренных в этом стандарте. Код может содержать дополнительный компонент, содержащий тег с двухбуквенным кодом страны, определенным в стандарте ИСО 3166, или с расширением кода языка, определенным организацией Internet Assigned Names Authority [http://www.iana.org/assignments/language-tags][23].

В то время как теги языка обычно изменяют смысл текста, код языка не меняет смысл символов текста. does not alter the meaning of the characters in the text[24].

Примечание — Представление тегов языка в тексте существенно зависит от спецификации реализуемой технологии. В ней может использоваться нативный способ указания тегов языка, предусмотренной в ее целевой технологии реализации. В некоторых технологиях информация о языке может быть указана в отдельном компо-

[23] Система кодирования, описанная в документе RFC 3066 [http://www.ietf.org/rfc/rfc3066.txt], одобрена комитетом HL7 для всех ссылок на человеческие языки, используемых в типах данных и других местах.

[24] По этой причине система или место реализации, которые не способны обрабатывать многоязычный текст или имена, могут спокойно игнорировать свойство language.

ненте, например, в XML язык строки может быть указан в теге xml:lang. В других коды языка могут быть частью двоичного представления строк символов, как это специфицировано в стандарте ИСО 10646 (Unicode) и его тегах «plane-14».

Если в технологии реализации тег языка не обязателен, то не должно быть обязательным и свойство language. Семантически объявление языка в строках следует логике умолчаний. Если на данной территории реализации может поддерживаться несколько языков, то эта территория сама устанавливает правила задания языка в случае, когда никакой язык не указан. Если никаких правил не задано, то подразумевается местный язык читателя. Если язык указан для всего сообщения или документа, то этот язык используется по умолчанию. Если язык указан для любого элемента информации, стоящего выше в синтаксической иерархии, то этот язык используется по умолчанию во всех подчиненных текстовых значениях.

Если теги языка присутствуют в начале кодированного двоичного текста (например, в тегах plane-14 кодировки Unicode), то они являются источником значения свойства language для инкапсулированных данных.

B.2.4.4 Свойство compression : CS

Указывает, являются ли байтовые данные результатом сжатия исходных данных и какой алгоритм сжатия был использован.

Таблица B.8 — Домен алгоритмов сжатия

Код	Имя	Статус	Определение
DF	deflate	обязательный	Формат дефляционного сжатия, предложенный в документе RFC 1951 [http://www.ietf.org/rfc/rfc1951.txt]
GZ	gzip	индифферентный	Формат сжатия данных, совместимый с широко используемой утилитой GZIP и описанный в документе RFC 1952 [http://www.ietf.org/rfc/rfc1952.txt] (использует алгоритм дефляции)
ZL	zlib	индифферентный	Сжатый формат данных, также использующий алгоритм дефляции. Описан в документе RFC 1950 [http://www.ietf.org/rfc/rfc1952.txt]
Z	compress	запрещенный	Оригинальный алгоритм сжатия и формат файла, предложенный в операционной системе UNIX и использующий алгоритм LZC (вариант алгоритма LZW). Отягощен патентами и менее эффективен по сравнению с алгоритмов дефляции

Значения типа данных ST никогда не должны быть сжатыми.

B.2.4.5 Свойство reference: TEL

Телекоммуникационный адрес (TEL), например, URL для HTTP или FTP, по которому находятся ровно те же самые двоичные данные, которые могли бы быть вложенными в инкапсулированные данные.

Семантическое значение инкапсулированных данных одно и то же вне зависимости, являются ли они вложенными или ссылочными. Однако инкапсулированные данные, не являющиеся вложенными, имеют другое поведение, поскольку любая попытка использования этих данных требует загрузки из источника, на который показывает ссылка. Инкапсулированные данные могут одновременно содержать и вложенные данные, и ссылку.

Ссылка должна указывать на те же самые данные, что и вложены. Если данные, полученные по ссылке, не проходят проверку на целостность, не совпадают с вложенными данными или с теми данными, которые были извлечены по ссылке ранее и сохранены в кэше, то это должно признаваться ошибкой.

Свойство reference может содержать компонент usablePeriod (период доступности), указывающий, что данным могут быть доступны по ссылке только ограниченный период времени. Независимо от того, ограничена ли доступность ссылки значением usablePeriod, содержание ссылочных данных должно оставаться фиксированным. Любое приложение, использующее ссылку, должно получить те же самые данные. Ссылка не может быть повторно использована для получения другой версии этих данных или других данных.

В зависимости от атрибута или компонента, объявляющего инкапсулированные данные, использование ссылочных данных может быть запрещено. Значения типа ST должно всегда быть вложенным.

B.2.4.6 Свойство integrityCheck: BIN

Свойство integrityCheck содержит короткое двоичное значение, представляющее криптографически стойкую контрольную сумму, вычисленную по двоичным данным. Это свойство предназначено для использования при передаче ссылок на данные и позволяет позже проверить, не изменились ли ссылочные данные после того, как было создано инкапсулированное значение, ссылающееся на эти данные.

Если данные, извлеченные по ссылке, не проходят проверку целостности, это должно признаваться ошибкой.

Контрольная сумма integrityCheck вычисляется с помощью алгоритма, указанного в свойстве integrityCheckAlgorithm. По умолчанию должен использоваться алгоритм *Secure Hash Algorithm-1* (SHA-1). Контрольная сумма представляется в двоичном виде в соответствии с правилами, принятыми в алгоритме контрольного суммирования.

Контрольная сумма вычисляется по двоичным данным, содержащимся в компоненте данных или доступным по ссылке. Перед вычислением контрольной суммы никаких преобразований не делается. Если данные сжаты, то контрольная сумма вычисляется по сжатым данным.

B.2.4.7 Свойство integrityCheckAlgorithm: CS

Указывает алгоритм, использованный для вычисления значения свойства integrityCheck[25].

Таблица B.9 — Домен значений свойства integrityCheckAlgorithm

Код	Имя	Определение
SHA-1	secure hash algorithm — 1	Этот алгоритм определен в документе FIPS PUB 180-1: Secure Hash Standard (по состоянию на 17 апреля 1995 г.)
SHA-256	secure hash algorithm — 256	Этот алгоритм определен в документе FIPS PUB 180-2: Secure Hash Standard

B.2.4.8 Свойство thumbnail : ED

Определение: сокращенное представление полных данных (эскиз), требующее гораздо меньше ресурсов, нежели полные данные, но при этом обладающее отличительными способностями полных данных. Обычно эскизы используются для ссылочных инкапсулированных данных. Они позволяют пользователю выбирать нужные данные, не загружая по ссылке их полное представление.

Первоначально термин «эскиз» (thumbnail) использовался для обозначения изображения в более низком разрешении (или меньшего размера) по сравнению с другим изображением. Однако это понятие может быть метафорически использоваться для других типов данных, кроме изображений. Например, видеозапись может быть представлена коротким клипом, аудиоклип может быть представлен другим аудиоклипом, более коротким, имеющим меньшую частоту оцифровки или использующим искажающее сжатие.

В зависимости от атрибута или компонента, объявляющего инкапсулированные данные, использование эскизов может быть запрещено. Значения типа ST никогда не имеют эскизов, и эскиз не может содержать вложенный эскиз.

```
invariant(ED x)
    where x.thumbnail.nonNull {
  x.thumbnail.thumbnail.isNull;
};
```

Примечание — В спецификациях реализуемой технологии следует рассматривать случай, когда и эскиз, и оригинал имеют одни и те же свойства типа, набора символов и сжатия. Тогда эти свойства можно не задавать для эскиза, а считать «унаследованными» от соответствующих свойств основных инкапсулированных данных.

B.2.4.9 Свойство equality: BL, унаследовано от типа данных ANY

Два значения типа *ED* равны в том и только том случае, если у них одинаковы типы среды и данные. Если значения типа *ED* содержат ссылочные или сжатые (упакованные) данные, то в проверке на равенство должны участвовать данные, извлеченные по ссылке и распакованные. Свойства сжатия compression, эскиза thumbnail и ссылки reference в проверке на равенство не участвуют. Кроме того, из этой проверки исключается свойство языка language, поскольку сложно понять, о какой язык подразумевается, если он не указан. Если типом среды (свойство mediaType) являются символьные данные и свойства набора символов (charset) не одинаковы, то перед проверкой надо отобразить данные на общий набор символов.

Свойства контрольной суммы integrityCheck и алгоритма контрольного суммирования integrityCheckAlgorithm исключаются из проверки на равенство. Однако в случае, если алгоритмы контрольного суммирования одинаковы, вполне практичной является проверка совпадения контрольных сумм, поскольку оно может служить признаком равенства.

B.2.5 Тип данных CharacterString (ST) (специализация типа данных ED)

Строковый тип данных, предназначенный для представления текста, предназначенного в основном для машинной обработки (например, сортировка, запросы, индексирование и т. д.). Используется для представления имен, символов и формальных выражений.

Тип данных ST является ограничением типа данных ED, у которого свойство ED.mediaType имеет фиксированное значение «text/plain» (неформатированный текст), а данные являются вложенными и не сжатыми. Поэтому свойства compression, reference, integrityCheck, integrityCheckAlgorithm и thumbnail к типу данных ST не применимы.

---

[25] В настоящее время промышленным стандартом является алгоритм вычисления криптографически стойкой контрольной суммы Secure Hash Algorithm-1 (SHA-1). Он заменил алгоритм MD5 всего лишь пару лет назад, когда обнаружились некоторые недостатки в безопасности этого алгоритма. В настоящее время по умолчанию для вычисления контрольной суммы используется алгоритм SHA-1. Становится популярным также алгоритм SHA-256.

Строковый тип данных используется, когда внешний вид текста не имеет значения, что справедливо для формальных выражений и всех типов имен.

Таблица В.10 — Сводка свойств строкового типа данных CharacterString

Имя	Тип	Описание
mediaType	CS	Идентифицирует тип инкапсулированных данных и метод их интерпретации или визуализации
charset	CS	Для типов данных, основанных на кодировании символов, это свойство указывает используемый набор символов и его кодировку. Набор символов должен идентифицироваться в соответствии с правилами регистрации наборов символов в организации Internet Assigned Numbers Authority (IANA) [http://www.iana.org/assignments/character-sets] и положениями документа RFC 2978 [http://www.ietf.org/rfc/rfc2978.txt]
language	CS	Для информации, основанной на символах, свойство language (язык) уазывает человеческий язык текста

Тип данных ST интерпретирует инкапсулированные данные как символьные (а не битовые) в зависимости от свойства charset инкапсулированного типа данных.

```
type CharacterString alias ST specializes ED {
  INT  length;
  ST   headCharacter;
  ST   tailString;
};
```

Примечание — Поскольку в данном случае многие свойства инкапсулированных данных имеют значения по умолчанию, то в спецификации реализуемой технологии эти свойства вообще не надо предоставлять. В действительности, если кодировка символов также фиксирована, то достаточно представлять только закодированные символьные данные.

Свойства headCharacter (голова) и tailString (хвост) определяют тип данных *ST* как последовательность элементов, каждый из которых уникально идентифицирует один символ из объединения наборов всех символов, известных во всех языках мира[26)].

Голова типа данных ST представляет собой строку, состоящую ровно из одного символа. Значение типа ST должно содержать хотя бы один символ, в противном случае оно считается пустым. Значение типа ST, имеющее нулевую длину, является исключительным значением (NULL), а не допустимым значением.

```
invariant(ST x)
    where x.nonNull {
  x.headCharacter.notEmpty;
  x.headCharacter.length.equal(1);
```

[26)] Стандарт ИСО/МЭК 10646-1:1993 определяет символ как «Член множества элементов, используемых для организации, управления или представления данных». В техническом отчете ИСО/МЭК ТО 15285 «Информационные технологии. Операционная модель применения графических символов и глифов» обсуждаются проблемы, связанные с определением понятия символов. Примечательно, что символы рассматриваются как абстрактные единицы информации, не зависящие от шрифта или языка. В стандарте ИСО 10646 (UNICODE [http://www.unicode.org]) (в Японии JIS X0221) описан глобально применимый набор символов, уникально идентифицирующий все символы всех языков мира.

В настоящей спецификации стандарт ИСО 10646 используется как семантическая модель строк символов. Важный момент состоит в том, что в ней нет понятия отдельных наборов символов и переключения от одного набора к другому. Наборы символов и их кодировки рассматриваются на уровне спецификации реализуемой технологии. Формальное определение указывает на этот эффект, поскольку каждый символ сам по себе является значением типа ST, имеющим свойство набора символов charset. Поэтому двоичное кодирование каждого символа всегда осуществляется в контексте определенного набора символов. Это вовсе не означает, что спецификация реализуемой технологии всегда должна представлять строку символов как последовательность полновесных значений типа данных ED. Это значит лишь то, что на прикладном уровне понятие кодировки символов не имеет значения для обсуждения строк символов.

```
  x.headCharacter.tailString.isEmpty;
  x.tailString.isEmpty.implies(x.length.equal(1));
  x.tailString.notEmpty.implies(x.length.equal(x.tailString.length.successor));
};
```

Длиной значения типа данных ST является число символов, а не число кодированных байтов. Байтовое кодирование является предметом спецификации реализуемой технологии, а не прикладного уровня.

К пробельным символам, содержащимся в значениях типа данных ST, применяются следующие правила:

- пробельными символами считаются TAB, пробел и символ конца строки;
- как ведущие, так и концевые пробельные символы считаются значащими;
- различные пробельные символы не являются взаимозаменяющими.

Различные представления конца строки нормализуются в соответствии с методом, описанным в спецификации языка XML [Section 2.11 End-of-Line Handling].

Последовательности пробельных символов не могут сжиматься для получения более коротких последовательностей.

Требование

Тип данных ST является специализацией типа данных ED, поэтому каждый атрибут модели RIM, имеющий тип ED, может быть ограничен до типа данных ST. Наиболее важным случаем является атрибут Act.text, который имеет тип данных ED, позволяющий указывать ссылки и использовать мультимедийные данные. Однако тип этого атрибута нередко ограничивается до неформатированного текста.

B.2.5.1 Свойство mediaType: CS (унаследовано от типа данных ED)

```
invariant(ST x)
    where x.nonNull {
  x.mediaType.equal("text/plain");
};
```

Это свойство имеет фиксированное значение «text/plain».

B.2.5.2 Свойство charset: CS (унаследовано от типа данных ED)

Для значений типа данных ST должен быть известен набор символов.

B.2.5.3 Свойство language: CS (унаследовано от типа данных ED)

Применительно к символьной информации свойство language указывает человеческий язык текста.

Необходимость указания кода языка для текстовых данных документирована в публикации RFC 2277 «IETF Policy on Character Sets and Languages» [http://www.ietf.org/rfc/rfc2277.txt]. Дополнительную базовую информацию можен найти в документе «Using International Characters in Internet Mail» [http://www.imc.org/mail-i18n.html] (меморандум консорциуима Internet Mail Consortium).

Принципы, положенные в основу домена значений этого свойства изложены в стандарте Интернет RFC 3066 [http://www.ietf.org/rfc/rfc3066.txt]. Система кодирования, описанная в этом документе, конструируется из основного компонента, содержащего тег с кодом языка, определенным в стандарте ИСО 639, и двух кодов расширения языков, не предусмотренных в этом стандарте. Код может содержать дополнительный компонент, содержащий тег с двухбуквенным кодом страны, определенным в стандарте ИСО 3166, или с расширением кода языка, определенным организацией Internet Assigned Names Authority [http://www.iana.org/assignments/language-tags][27].

В то время как теги языка обычно изменяют смысл текста, код языка не меняет смысл символов текста. does not alter the meaning of the characters in the text[28].

Примечание — Представление тегов языка в тексте существенно зависит от спецификации реализуемой технологии. В ней может использоваться нативный способ указания тегов языка, предусмотренной в ее целевой технологии реализации. В некоторых технологиях информация о языке может быть указана в отдельном компоненте, например, в XML язык строки может быть указан в теге xml:lang. В других коды языка могут быть частью двоичного представления строк символов, как это специфицировано в стандарте ИСО 10646 (Unicode) и его тегах «plane-14».

Если в технологии реализации тег языка не обязателен, то не должно быть обязательным и свойство language. Семантически объявление языка в строках следует логике умолчаний. Если на данной территории реализации может поддерживаться несколько языков, то эта территория сама устанавливает правила задания языка в случае, когда никакой язык не указан. Если никаких правил не задано, то подразумевается местный язык читателя.

---

[27] Система кодирования, описанная в документе RFC 3066 [http://www.ietf.org/rfc/rfc3066.txt], одобрена комитетом HL7 для всех ссылок на человеческие языки, используемых в типах данных и других местах.

[28] По этой причине система или место реализации, которые не способны обрабатывать многоязычный текст или имена, могут спокойно игнорировать свойство language.

Если язык указан для всего сообщения или документа, то этот язык используется по умолчанию. Если язык указан для любого элемента информации, стоящего выше в синтаксической иерархии, то этот язык используется по умолчанию во всех подчиненных текстовых значениях.

Если теги языка присутствуют в начале кодированного двоичного текста (например, в тегах plane-14 кодировки Unicode), то они являются источником значения свойства language для инкапсулированных данных.

B.2.5.4 Свойство compression: CS (фиксированное)

```
invariant(ST x)
    where x.nonNull {
  x.compression.notApplicable;
};
```

Значения типа ST не могут быть сжатыми.

B.2.5.5 Свойство reference: TEL (фиксированное)

```
invariant(ST x)
    where x.nonNull {
  x.reference.notApplicable;
};
```

Значения типа ST не могут быть ссылаться на содержание, находящееся в каком-то другом месте.

B.2.5.6 Свойство integrityCheck: BIN (фиксированное)

```
invariant(ST x)
    where x.nonNull {
  x.integrityCheck.notApplicable;
};
```

Контрольная сумма значений типа ST не вычисляется.

B.2.5.7 Свойство integrityCheckAlgorithm: CS (фиксированное)

```
invariant(ST x)
    where x.nonNull {
  x.integrityCheckAlgorithm.notApplicable;
};
```

К значениям типа ST алгоритм контрольного суммирования не применяется.

B.2.5.8 Свойство thumbnail: ED (фиксированное)

```
invariant(ST x)
    where x.nonNull {
  x.thumbnail.notApplicable;
};
```

Эскизы для значений типа ST не могут быть определены.

B.2.5.9 Литеральная форма

Определены два варианта литералов типа данных ST: форма токена и строка в кавычках[29]. Форма токена состоит из строчных и прописных символов латинского алфавита, десяти цифр и подчеркивания. Строка в кавычках может содержать между кавычками любой символ. Кавычки предотвращают интерпретировать строку символов как литерал другого типа. Форма токена используется для представления имен и ключевых слов, разбираемых в языке спецификации типов данных.

---

[29] Литерал типа данных ST literal представляет собой преобразование строки символов в другой тип данных. Очевидно, литералы типа данных ST для символьных строк представляют собой циклическое, если не избыточное образование. Поэтому литеральная форма в основном указывает, каким образом осуществляется разбор строк символов на языке определения типов данных.

```
ST.literal ST {
  ST : /"[^]+"/        { $.equal($1); }  /* quoted string */
     | /[a-zA-Z0-9_]+/  { $.equal($1); }; /* token form */
};
```

Примечание — Поскольку литералы типа данных ST играют фундаментальную роль в технологии реализации, в большинстве спецификаций реализуемой технологии будет использоваться некоторая модифицированная форма литерала строки символов. Однако разработчики этих спецификаций должны учитывать возможность пересечения литеральной формы типа данных ST с литеральными формами, определенными для других типов данных. Такое пересечение особенно критично, если литеральная форма другого типа данных структурирована в виде основных компонентов, разделенных специальными символами (например, вещественное число, физическая величина, литералы множества и списка, и т. д.).

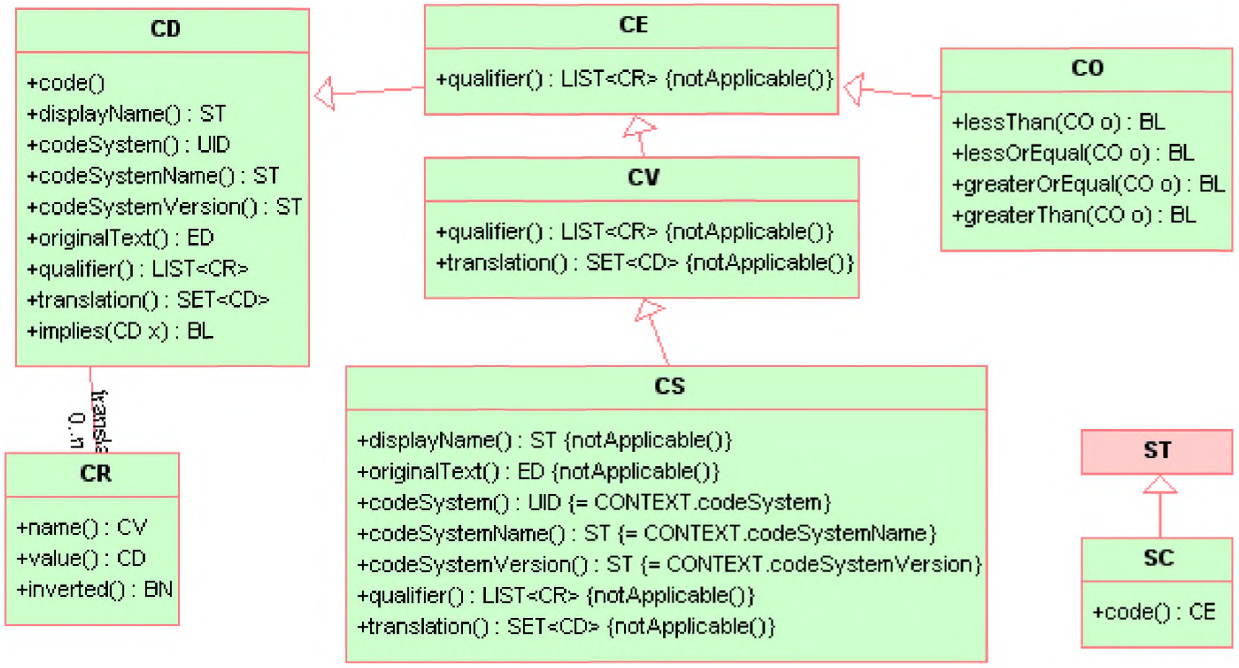


Рисунок В.4 — Информационная модель дескриптора понятия

В.2.6 Тип данных ConceptDescriptor (CD) (специализация типа данных ANY)

Обычно тип данных CD представляет любой вид понятия с помощью кода, определенного в системе кодирования. Значение типа данных CD может содержать исходный текст или фразу, используемые как основа для кодирования, а также одно или несколько преобразований в другие системы кодирования. Значение типа данных CD может содержать квалификаторы, позволяющие описать, к примеру, понятие «левая нога» как посткоординированный термин, составленный из основного кода «НОГА» и квалификатора «ЛЕВЫЙ». В исключительных значениях типа данных CD код отсутствует, а присутствует только исходный текст, описывающий данное понятие.

Таблица В.11 — Сводка свойств типа данных ConceptDescriptor

Имя	Тип	Описание
code	ST	Символ кода, определенный в системе кодирования. Например, «784.0» является символом кода головной боли «784.0», определенным в системе кодирования МКБ-9
codeSystem	UID	Указывает систему кодирования, в которой определен код
codeSystemName	ST	Общее имя системы кодирования

Окончание таблицы В.11

Имя	Тип	Описание
codeSystemVersion	ST	Если применим, дескриптор версии, определенный специально для данной системы кодирования
displayName	ST	Имя или название кода, под которым система-отправитель показывает значение кода своим пользователям
originalText	ED	Текст или фраза, используемые в качестве основы для кодирования
translation	SET<CD>	Множество других дескрипторов понятия, которые преобразуют данный дескриптор понятия в другие системы кодирования
qualifier	LIST<CR>	Указывает дополнительные коды, которые повышают специфичность основного кода

```
type ConceptDescriptor alias CD specializes ANY {
        ST          code;
        ST          displayName;
        UID         codeSystem;
        ST          codeSystemName;
        ST          codeSystemVersion;
        ED          originalText;
        LIST<CR>    qualifier;
        SET<CD>     translation;
        BL          equal(ANY x);
        BL          mplies(CD x);
 demotion           ED;
};
```

Тип данных CD обычно используется в одной из своих ограниченных, или «профилированных» форм CS, CE, CV. Тип данных полного дескриптора понятия используется не часто. Для его применения необходимы осознанное решение и документированное обоснование. Во всех остальных случаях используется одно из ограничений типа данных CD[30)].

Все ограничения на тип данных CD накладываются на определенные свойства. Эти свойства могут быть ограничены до такой степени, что допустимым окажется единственное значение, после чего упоминание этого свойства станет избыточным. Ограничение свойства до единственного значения называется подавлением этого свойства. Хотя концептуально подавленное свойство продолжает быть семантически приемлемым, в любом интерфейсе HL7 можно вполне безопасно без проверки предполагать значение по умолчанию.

Примечание — Вообще говоря, сказанное верно для многих типов данных, определенных в настоящей спецификации, однако чаще всего возникают вопросы по поводу потомков типа данных CD.

В.2.6.1 Свойство code: ST

Определение: символ кода, определенный в системе кодирования. Например, «784.0» является символом кода головной боли «784.0», определенным в системе кодирования МКБ-9.

Не исключительное значение типа данных CD имеет непустое свойство code, значением которого является строка символов, определенная в системе кодирования, идентифицированной в свойстве codeSystem. И наоборот, значение типа CD, у которого свойство code не имеет значения или имеет значение, не принадлежащее указанной системе кодирования, считается исключительным значением (пустым значением (NULL) с типом пустоты «other»).

```
invariant(CD x)
    where x.nonNull {
  x.code.nonNull;
};
```

[30)] Преимуществом типа данных CD является его выразительность. Но постоянное использование всех его свойств, например, исключений из кодирования, текста, преобразований и квалификаторов во всех реализациях становится очень затруднительным и небезопасным. Поэтому в большинстве случаев тип данных CD используется в ограниченной форме с меньшим числом свойств.

B.2.6.2 Свойство codeSystem: UID

Указывает систему кодирования, в которой определено значение свойства code.

Идентификатор системы кодирования должен иметь тип данных UID, позволяющий однозначно указать стандартные системы кодирования HL7, другие стандартные системы кодирования, а также местные системы кодирования. Комитет HL7 должен присваивать идентификатор типа UID каждой из своих таблиц кодов, а также внешним стандартным системам кодирования, которые используются в стандартах HL7. На местах должны использовать свои объектные идентификаторы ИСО (тип данных OID), с помощью которых можно сконструировать глобально уникальные идентификаторы местных систем кодирования.

Под ветвью комитета HL7, 2.16.840.1.113883, подветви 5 и 6 содержат соответственно идентификаторы стандартных систем кодирования HL7 и внешних систем кодирования. Эти ветви ведутся техническим комитетом HL7 Vocabulary Technical Committee.

Не исключительное значение типа CD (то есть значение типа CD с непустым свойством code) имеет непустое свойство codeSystem, указывающее систему понятий, в которой определено значение свойства code. Другими словами, если есть код, должна быть и система кодирования.

Примечание — Хотя для каждого непустого значения типа CD определена конкретная система кодирования, в некоторых обстоятельствах представление значения типа CD в соответствии со спецификацией реализуемой технологии не нуждается в явном упоминании системы кодирования. Например, когда контекст подразумевает одну и только одну систему кодирования, то ее явное указание стало бы избыточным. Однако в таком случае свойство codeSystem принимает контекстно-зависимое значение по умолчанию и не является пустым.

```
invariant(CD x)
    where x.code.nonNull {
  x.codeSystem.nonNull;
};
```

Причина пустоты «other» у исключительного значения типа CD указывает, что понятие не может быть закодировано в указанной системе кодирования. Эта система кодирования, в которой нет такого исключительного понятия, должна быть указана в свойстве codeSystem.

Некоторые домены кодов квалифицированы таким образом, что они могут включать в себя некоторые части подходящей местной системы кодирования, не являющиеся парафразами стандартной системы кодирования (coded with extensibility, CWE — кодированные с расширением). Если поле с квалификатором «CWE» действительно содержит такой местный код, то в свойстве системы кодирования должен быть указан идентификатор местной системы кодирования, из которой взят этот код. Однако в доменах с квалификатором «CWE» местный код является допустимым членом домена, поэтому использование местного кода не является ни ошибкой, ни исключительным значением (с причиной пустоты «other») в смысле настоящей спецификации.

```
invariant(CD x)
    where x.other {
  x.code.other;
  x.codeSystem.nonNull;
};
```

B.2.6.3 Свойство codeSystemName: ST

Общее имя системы кодирования.

Имя системы кодирования не используется для вычислений. Оно может быть указано для облегчения интерпретации человеком значений свойств code и codeSystem. Предполагается, хотя и не является обязательным, что в спецификациях реализуемой технологии будет предусматриваться указание значения свойства codeSystemName в качестве аннотации к идентификатору UID, рассчитанной на восприятие человеком.

Системы, соответствующие стандарту HL7, не должны функционально полагаться на значение свойства codeSystemName. Это значение не должно модифицировать значение свойства codeSystem и не может существовать без значения свойства codeSystem.

```
invariant(CD x) {
  x.codeSystemName.nonNull.implies(x.codeSystem.nonNull);
};
```

B.2.6.4 Свойство codeSystemVersion: ST

Если применим, дескриптор версии, определенный специально для данной системы кодирования.

Для каждой внешней системы кодирования в стандарте HL7 будет указано, как формируется строка со значением версии. Если для конкретной системы кодирования такое указание отсутствует, то для такой системы обозначение версии не имеет определенного значения.

Различные версии одной и той же системы кодирования должны быть совместимыми. Если система кодирования изменена несовместимым образом, то она представляет собой другую систему кодирования, а не другую версию, как бы издатель это не называл.

Например, издатель классификаций МКБ-9 и МКБ-10 назвал эти системы кодирования «9-м пересмотром» и «10-м пересмотром». Однако МКБ-10 представляет собой полное изменение кодов МКБ, и не является обратно совместимой. Поэтому в целях настоящей спецификации МКБ-9 и МКБ-10 рассматриваются как разные системы кодирования, а не как разные версии одной системы кодирования. Напротив, когда версия «1.0j» системы кодирования LOINC была обновлена до версии «1.0k», комитет HL7 рассматривал это как смену версий, поскольку новая версия была обратно совместимой с предыдущей.

```
invariant(CD x) {
  x.codeSystemVersion.nonNull.implies(x.codeSystem.nonNull);
};
```

B.2.6.5 Свойство displayName: ST

Имя или название кода, под которым система-отправитель показывает значение кода своим пользователям.

Свойство displayName включено как для удобства интерпретации человеком значения кода, так и для документирования имени, используемого для изображения понятия пользователю. Оно не имеет функционального значения, не может существовать без кода и никогда не должно модифицировать смысл кода.

Примечания

1 В своих словарных доменах стандарты HL7 предусматривают атрибут «печатаемого имени» (print name). Значения этого атрибута могут использоваться в качестве значений свойства displayName.

2 Имена, предусмотренные для кодов, не могут менять смысл кодированного значения. Поэтому они не должны предоставляться пользователю прикладной системы-получателя, пока не будет уверенности в том, что такое имя адекватно отражает понятие, соответствующее кодированному значению. При коммуникациях нельзя просто полагаться на имя кодированного значения. Основная цель этого имени в обеспечении возможности отладки единиц данных в протоколах HL7 (например, сообщений).

```
invariant(CD x) {
  x.displayName.nonNull.implies(x.code.nonNull);
};
```

B.2.6.6 Свойство originalText: ED

Текст или фраза, используемые в качестве основы для кодирования.

Исходный текст появляется в тех случаях, когда код был присвоен информации не ее источником, а специальным кодировщиком уже после ее появления. Тогда при создании дескриптора понятия исходный текст может существовать без кода.

Примечание — Хотя кодировка пост-фактум часто осуществляется по информации, представленной как свободный текст, например, в форме документов, сканированных изображений или диктовки, мультимедийные данные явным образом не разрешены в качестве исходного текста. Кроме того, свойство исходного текста originalText не может означать ссылку на весь исходный документ. Связи между различными артефактами медицинской информации (например, документом и кодированным результатом) не входят в область применения настоящей спецификации и рассматриваются в других стандартах HL7. Исходный текст должен представлять собой извлечение из оригинального источника, а не точную копию его содержания или указатель на этот источник. Поэтому исходный текст должен представляться в неформатированном виде.

Значение типа CD могут иметь непустое свойство исходного текста, даже если свойство code имеет пустое значение. В этом случае свойство originalText является именем или описанием понятия, которое не было закодировано. Такие исключительные значения типа CD могут также содержать преобразования translation. Такие преобразования обеспечивают непосредственное кодирование понятия, описанного в свойстве originalText.

Тип данных CD может быть понижен до типа данных ST. В этом случае значение типа ST будет представлено только свойством originalText значения типа CD.

```
invariant(CD x)
    where x.originalText.nonNull {
  ((ST)x).equal(x.originalText);
};
```

B.2.6.7 Свойство translation: SET<CD>

Множество других дескрипторов понятия, которые преобразуют данный дескриптор понятия в другие системы кодирования.

Свойство translation представляет собой множество других значений типа CD, каждое из которых преобразует первое значение типа CD в другую систему кодирования. Каждый элемент множества преобразований был получен с помощью преобразования первого значения типа CD. Каждое преобразование в свою очередь может также содержать преобразования. Таким образом, если код многократно преобразуется, то информация о том, какой код послужил входным параметром каждого преобразования, сохранится.

Примечание — Преобразования являются квазисинонимами одного понятия реального мира. Предполагается, что каждое преобразование в этом множестве выражает то же самое понятие «другими словами». Однако между двумя структурно различными системами кодирования редко существует точная синонимия. Поэтому не все преобразования будут абсолютно точными.

B.2.6.8 Свойство qualifier: LIST<CR>

Указывает дополнительные коды, которые повышают специфичность основного кода.

Основной код и все его квалификаторы (qualifier) в совокупности идентифицируют одно понятие. Значение типа CD с квалификаторами называется также кодовой фразой или посткоординируемым выражением.

Квалификаторы ограничивают значение основного кода, но их применение не должно приводить к отрицанию значения основного кода или к такому его изменению, при котором будет получено другое значение из основной системы кодирования.

Квалификаторы могут использоваться только в сочетании с хорошо определенными правилами посткоординации. Значение типа CD может иметь квалификаторы только в том случае, когда его система кодирования допускает использование или когда существует третья система кодирования, в которой указано, каким образом могут сочетаться другие системы кодирования.

Например, номенклатура SNOMED CT позволяет конструировать понятие с помощью сочетания нескольких кодов. В этой номенклатуре определено понятие «целлюлит (нарушение)» (128045006), атрибут «локализация» (363698007) и другое понятие «структура ноги (структура тела)» (56459004). Номенклатура SNOMED CT позволяет создать кодовую фразу из этих кодов.

Примеры

```
1 <observation>
  ...
  <value code="128045006" codeSystem="&amp;amp;SNOMED-CT;" displayName="целлюлит (нарушение">
    <qualifier code=»56459004» displayName=»структура ноги (структура тела)»>
      <name code="363698007" displayName="локализация"/>
    </qualifier>
  </value>
  ...
</observation>
```

В данном примере в одной системе кодирования, SNOMED-CT, определены и основной код, и все квалификаторы, а также правила их сочетания. Поэтому в нем не требуется отдельно упоминать систему кодирования codeSystem для имени и значений квалификаторов (значение свойства codeSystem наследуется от основного кода).

Важно отметить, что все допустимые квалификаторы указываются системой кодирования. Например, в номенклатуре SNOMED CT определено множество квалифицирующих атрибутов, при этом только исследования (Finding) и нарушения (Disorder) могут квалифицироваться атрибутом «локализация» (finding site). Использование квалификаторов за пределами границ, установленными системой кодирования, является несовместимым использованием типа данных CD. Следуя правилам, описанным в системе кодирования, можно сравнить посткоординированные выражения с пре-координированными понятиями (например, можно сравнить приведенную выше кодовую фразу с пре-координированным понятием «целлюлит ноги (нарушение)» (128276007), которое определено в номенклатуре SNOMED CT как относящееся к локализации «структура ноги»). В типе данных CD не предусмотрена нормализация композиции кодов, поэтому могут создаваться неоднозначные кодовые фразы. Следует иметь в виду, что для создания однозначных представлений данных с помощью композиции кодов с типом CD, необходимо наложить на композицию дополнительные ограничения. Иначе может оказаться, что извлечение полного набора записей, удовлетворяющих определенному запросу, окажется невозможным.

Другим общим примером могут служить коды процедур, используемые организацией U.S. Centers for Medicare and Medicaid Services (CMS) (ранее известной под названием Health Care Financing Administration, HCFA). Коды процедур, используемые организацией CMS (HCPCS) основаны на классси-

фикации CPT-4 и добавляют к ней различные квалификаторы. Например, пациенту с указанным выше нарушением (страдающему также заболеванием периферических артерий, сахарным диабетом и хроническим поражением кожи на большом пальце левой стопы) могли ампутировать этот палец. Для кодирования этой процедуры используется понятие «Ампутация, плюснефаланговый сустав стопы» (код понятия 28820), определенное в классификации CPT-4 (28820) и квалификатор «большой палец левой стопы» с кодом TA из системы HCPCS. Ниже показан пример сочетания этого кода с квалификатором.

```
2 <procedure>
  ...
   <cd code=»28820» codeSystem=»&amp;CP4;» displayName=» Ампутация, плюснефаланговый сустав стопы»>
     <qualifier code=»TA» codeSystem=»&amp;HCP;» displayName=»большой палец левой стопы»/>
   </cd>
  ...
</procedure>
```

В этом примере система кодирования квалификатора (HCPCS) отличается от системы кодирования основного кода (CPT-4.) Этот квалификатор может использоваться только в силу наличия хорошо определенных правил сочетания этих кодов. Учтите также, что имя роли квалификатора не является обязательным, и для кодов HCPCS имена ролей не назначаются.

Порядок применения квалификаторов сохраняется, особенно в случае, когда система кодирования позволяет посткоординацию, но не определяет имена ролей (как это имеет место для некоторых кодов клинической модификации МКБ-9 в США (ICD-9CM) или для «многоосевого» кодирования в старой версии номенклатуры SNOMED «multiaxial»).

B.2.6.9 Свойство equality: BL (унаследовано от типа данных ANY)

Основными применениями дескрипторов понятий являются индексирование, запросы и принятие решения на основе кодированного значения. Поэтому для семантически однозначной спецификации кодированных значений требуется точное определение того, что означает равенство дескрипторов понятий и как можно сравнивать значения типа CD. (Дополнительные сведения о сравнении пре- и посткоординированных выражений см. в публикации Dolin RH, Spackman KA, Markwell D. Selective Retrieval of Pre- and Post-coordinated SNOMED Concepts. Fall AMIA 2002; 210-14, или в руководстве SNOMED CT Implementation Guide, July 2003.)

Равенство двух значений типа CD определяется исключительно по значения свойств code и codeSystem. Свойство codeSystemVersion из проверки на равенство исключается[31].Если присутствуют квалификаторы, то они учитываются при проверке на равенство. Преобразования не включаются в проверку[32]. Исключительные значения типа CD не считаются равными даже в том случае, если они имеют одну и ту же причину пустоты или один и тот же исходный текст[33].

```
invariant(CD x, y)
    where x.nonNull.and(y.nonNull) {
  x.equal(y).equal(x.code.equal(y.code)
         .and(x.codeSystem.equal(y.codingSystem))
         .and(x.qualifier.equal(y.qualifier)));
};
```

---

[31] Свойство codeSystemVersion не учитывается при проверке равенства, поскольку по определению символ кода должен иметь одно и то же значение во всех версиях системы кодирования. При переходе к другой версии код может быть объявлен устаревшим, но не может быть удален или повторно использован для другого понятия.

[32] Преобразования не включаются в проверку на равенство по соображениям безопасности. Альтернативой могло быть признание двух значений типа CD равными, если равны какие-либо из их преобразований. Однако некоторые преобразования могут оказаться одинаковыми, поскольку их система кодирования является очень грубой. Более интеллектуальные сравнения дескрипторов зависят от приложений и не обсуждаются в настоящей спецификации.

[33] Пустые значения (NULL) являются исключительными значениями, а не допустимыми понятиями. Было бы небезопасно приравнивать два значения только на том основании, что они оба исключительны (например, не закодированы или не известны). Аналогично, нет никакой гарантии, что исходный текст представляет собой уникальное описание понятия, поэтому совпадение исходных текстов не означает равенство понятий. Верно и обратное: так как одно и то же понятие может быть описано с помощью разных исходных текстов, то несовпадение текстов не означает, что понятия различаются.

В некоторых системах кодирования определены разные стили представления кодовых значений. Например, в системе идентификации лекарственных средств U.S. National Drug Code (NDC) предусмотрены две формы: с дефисами и без дефисов. Примером формы с дефисами служит код 1234-5678-90, а формы без дефисов — 01234567890. Другим примером подобной проблемы служат некоторые таблицы кодов ИСО или ANSI, в которых определены необязательные алфавитно-цифровые и числовые формы кода из двух или трех символов для всех кодируемых значений.

В том случае, когда система кодирования предусматривает несколько представлений, комитетом HL7 должно быть принято решение о предпочтительной форме. Комитет должен документировать это решение, если им рекомендована соответствующая внешняя система кодирования. Решение о предпочтительной форме должно приниматься с учетом практичности и широты использования. Если четких критериев практичности и широты использования нет, то должно быть отдано предпочтение наиболее безопасной, расширяемой и наименее стилизованной (наименее декорированной) форме[34)].

B.2.6.10 Свойство implies : BL

Указывает, является ли данное значение типа CD специализацией операнда типа CD.

Естественно, понятия могут делаться более узкими или более широкими, что исключить или охватить другие понятия. Во многих системах кодирования обеспечиваются явные указания специализации и обобщения понятий. Принципы построения словаря HL7 также предусматривают специализацию понятий в наборах значений, определенных в стандартах HL7. Свойство implies (импликация) представляет собой предикат, устанавливающий, является ли одно понятие специализацией другого понятия, и, следовательно, выводится из этого другого понятия.

При составлении предикатов (например, условных операторов), сравнивающих два кода, обычно проверяется не равенство кодов, а возможность вывода одного из другого.

Например, в таблице B.12D:\ТК 468\Стандарты ISO\27932\ГОСТ Р ИСО 27932\infrastructure\datatypes\datatypes.htm - domain-TelecommunicationAddressUse, описывающей понятия «использования телекоммуникаций» определены следующие значения: «W» (рабочий), «H» (домашний), «HP» (основной домашний) и «HV» (домашний на время отпуска). Очевидно, что из обоих значений «HP» и «HV вытекает значение «H». При поиске какого-либо номера домашнего телефона необходимо установить, вытекает ли из данного значения кода использования телекоммуникаций значение «H». Если бы выполнялась проверка равенства коду «H», то были бы найдены только домашние номера без специализации, а не номер основного домашнего телефона.

С вычислительной точки зрения импликация может быть установлена одним из двух способов. Литералы в системе кодирования могут быть устроены таким образом, что иерархия может задаваться в самом литерале кода (как это имеет место в классификации МКБ-9). Но за исключением таких особых случаев для вычисления выражений, содержащих импликацию, потребуется терминологическая база знаний и алгоритм проверки поглощения понятий. Для посткоординированных систем кодирования разработка такого алгоритма представляет собой нетривиальную задачу[35)].

B.2.7 Тип данных ConceptRole (CR) (специализация типа данных ANY)

Код квалификатора понятия с необязательной именованной ролью. Как роль квалификатора, так и коды значений должны быть определены в системе кодирования значений типа CD, содержащей квалификатор понятия. Например, если в номенклатуре SNOMED RT определены понятие «нога», отношение роли «имеет-сторону» и еще одно понятие «левая», то отношение роли понятия позволяет добавить квалификатор «имеет-сторону: левая» к основному коду «нога» и тем самым получить понятие «левая нога».

Таблица B.12 — Сводка свойств типа данных ConceptRole

Имя	Тип	Описание
name	CV	Указывает способ, которым роль понятия вносит вклад в значение кодовой фразы. Например, если в номенклатуре SNOMED RT определены понятие «нога», отношение роли «имеет-сторону» и еще одно понятие «левая», то отношение роли понятия позволяет добавить квалификатор «имеет-сторону: левая» к основному коду «нога» и тем самым получить понятие «левая нога». В данном примере свойство name (имя) имеет значение «имеет-сторону»

[34)] Это решение, принимаемое на этапе конструирования, необходимо, чтобы интерфейсы, предлагаемые в стандартах HL7, не были отягощены необходимостью реализации преобразований различных стилей литералов кодов во время исполнения, хотя это и может привести к ситуации, когда некоторым приложениям может потребоваться выполнять преобразования из одной формы представления кодов в другую, если они рассчитаны на вариант представления, не выбранный комитетом HL7.

[35)] Это одна из причин, почему использование квалификаторов в посткоординированных системах кодирования должно быть умеренным и осторожным. Другая проблема посткоординации состоит в том, что общее правило равенства может вообще не существовать.

Окончание таблицы В.12

Имя	Тип	Описание
value	CD	Понятие, которое модифицирует основной код кодовой фразы с помощью отношения роли. Например, если в номенклатуре SNOMED RT определены понятие «нога», отношение роли «имеет-сторону» и еще одно понятие «левая», то отношение роли понятия позволяет добавить квалификатор «имеет-сторону: левая» к основному коду «нога» и тем самым получить понятие «левая нога». В этом примере свойство value имеет значение «левая»
inverted	BN	Указывает, что смысл имени меняется на противоположный. Такое обращение смысла можно использовать в тех случаях, когда в соответствующей системе кодирования определено обращение, но не предусмотрена взаимно противоположная пара имен. По умолчанию свойство inverted имеет значение «false»

Применение квалификаторов строго контролируется используемой системой кодирования. Определение типа данных CD не разрешает использовать квалификаторы вместе с системами кодирования, где они не предусмотрены (например, с предкоординированными системами, скажем, LOINC, ICD-10 PCS).

```
protected type ConceptRole alias CR specializes ANY {
  CV name;
  BN inverted;
  CD value;
};
```

B.2.7.1 Свойство name: CV

Указывает способ, которым роль понятия вносит вклад в значение кодовой фразы. Например, если в номенклатуре SNOMED RT определены понятие «нога», отношение роли «имеет-сторону» и еще одно понятие «левая», то отношение роли понятия позволяет добавить квалификатор «имеет-сторону: левая» к основному коду «нога» и тем самым получить понятие «левая нога». В данном примере свойство name (имя) имеет значение «имеет-сторону».

Если родительская система кодирования CD.codeSystem позволяет посткоординацию, но не предусматривает имена ролей (например, SNOMED), то свойство name может быть пустым (NULL).

B.2.7.2 Свойство value: CD

Понятие, которое модифицирует основной код кодовой фразы с помощью отношения роли. Например, если в номенклатуре SNOMED RT определены понятие «нога», отношение роли «имеет-сторону» и еще одно понятие «левая», то отношение роли понятия позволяет добавить квалификатор «имеет-сторону: левая» к основному коду «нога» и тем самым получить понятие «левая нога». В этом примере свойство value имеет значение «левая».

Свойство value имеет тип данных CD и, следовательно, само может иметь квалификаторы. Таким образом, квалификаторы могут быть вложенными. Квалификаторы могут использоваться только в тех случаях, когда возможность их применения указана в соответствующей системе кодирования. Никакие виды квалификаторах нельзя применять для систем кодирования, в описании которых явно не указаны применение квалификаторов и правила их применения.

```
invariant(CR x)
    where x.nonNull {
  x.value.nonNull;
};
```

B.2.7.3 Свойство inverted: BN

Указывает, что смысл имени меняется на противоположный. Такое обращение смысла можно использовать в тех случаях, когда в соответствующей системе кодирования определено обращение, но не предусмотрена взаимно противоположная пара имен. По умолчанию свойство inverted имеет значение «false».

Например, в система кодирования могут быть определены отношение роли «вызывает» и понятия «Streptococcus pneumoniae» и «Пневмония». Если в этой системе кодирования разрешено обращение ролей, то можно сконструировать посткоординированное понятие «Пневмококковая пневмония» с помощью кодовой фразы «Пневмония», обращение «вызывает», «Streptococcus pneumoniae».

Роли могут быть обращены, если в соответствующей системе кодирования определено такое обращение. Но если система кодирования определяет роли в обращаемых парах или намеренно не определяет некоторые обращения, то вместо обращения надо использовать подходящий код роли (например, «вызывается» вместо обращения роли «вызывает»). Обязательно должно быть известно, принимает ли свойство inverted значение «true» или «false», поскольку если оно пусто, то роль нельзя интерпретировать.

Примечание — Свойство inverted должно передаваться как атрибут признака, который по умолчанию имеет значение «false». В этом случае признак обращения роли не должен передаваться, если роль не обращена.

B.2.8 Тип данных CodedSimpleValue (CS) (специализация типа данных CV)

Кодированные данные в их простейшей форма, в которой только свойство code не имеет предопределенных значений. Система кодирования и версия системы кодирования однозначно определяются контекстом значения типа CS.Тип данных CS используется для кодированных атрибутов, значения которых берутся из одного набора данных, определенного в стандарте HL7.

Таблица B.13 — Сводка свойств типа данных CodedSimpleValue

Имя	Тип	Описание
code	ST	Символ кода, определенный в системе кодирования. Например, «784.0» является символом кода головной боли «784.0», определенным в системе кодирования МКБ-9

```
type CodedSimpleValue alias CS specializes CV {
        ST  code;
  literal  ST;
};
```

Тип данных CS может использоваться только в следующих случаях:

- для кодированного атрибута, с которым связана единственная система кодирования, определенная комитетом HL7, и добавление кодов в которую требует от комитета формальных действий (например, гармонизации). Такие кодированные атрибуты должны иметь тип данных CS;
- для свойства, определенного в настоящей спецификации, которому присвоена единственная система кодирования, указанная в данной спецификации либо определенная вне комитета HL7 органом, ответственным за это понятие и ведение системы кодирования.

Например, организация IETF ведет систему кодирования типов среды MIME, используемую в типе данных ED. Таким образом, спецификация этого типа данных допускает также и предусмотренный этой организацией механизм расширения типов среды MIME (к примеру, «application/x-myapp»).

Для значений типа CS квалификатор расширяемости всегда имеет значение «CNE» (coded, non-extensible — кодируемый, не расширяемый) и по контексту определяется, какие можно использовать значения, определенные в стандарте HL7[36)].

B.2.8.1 Свойство code : ST (унаследовано от типа данных CD)

```
invariant(CS x)
    where x.nonNull {
  x.code.nonNull;
};
```

B.2.8.2 Свойство codeSystem: UID, (фиксированное)

С каждым непустым значением типа данных CS связана определенная система кодирования. Она не обязана быть явно указана в представлении типа данных CS, используемом в спецификации реализуемой технологии, поскольку по контексту определяется одна и только одна используемая система кодирования. Ее явное указание избыточно. Но при этом для нее подразумевается определенное непустое контекстно-специфическое значение.

```
invariant(CS x)
    where x.code.nonNull {
  x.codeSystem.nonNull;
  x.codeSystem.equal(CONTEXT.codeSystem);
};
```

36) Это не противоречит тому, что используемый по ссылке внешний домен, например, типы среды IETF MIME, могут иметь механизм расширения. Эти расширенные коды типов среды MIME не должны рассматриваться как «расширения», нарушающие квалификатор расширяемости CNE. Нарушение будет иметь место в том случае, если будет сделана попытка использовать другую систему кодирования (посредством свойства CD.codeSystem), что невозможно сделать для типа данных CS.

Исключительное значение типа CS с кодом причины пустоты «other» означает, что данное понятие не может быть закодировано в указанной системе кодирования. В этих случаях свойство code должно иметь пустое значение NULL.

```
invariant(CS x)
    where x.other {
  x.code.isNull;
  x.codeSystem.nonNull;
};
```

B.2.8.3 Свойство codeSystemName: ST (фиксированное)

```
invariant(CS x) {
  x.codeSystemName.equal(CONTEXT.codeSystemName);
};
```

B.2.8.4 Свойство codeSystemVersion: ST (фиксированное)

```
invariant(CS x) {
  x.codeSystemVersion.equal(CONTEXT.codeSystemVersion);
};
```

B.2.8.5 Свойство displayName: ST (фиксированное)

```
invariant(CS x) {
  x.displayName.notApplicable;
};
```

B.2.8.6 Свойство originalText: ED (фиксированное)

```
invariant(CS x) {
  x.originalText.notApplicable;
};
```

B.2.8.7 Свойство translation: SET<CD> (фиксированное)

```
invariant(CS x) {
  x.translation.notApplicable;
};
```

B.2.8.8 Свойство qualifier: LIST<CR>, (фиксированное)

```
invariant(CS x) {
  x.qualifier.notApplicable;
};
```

B.2.8.9 Литеральная форма

```
CS.literal ST {
  ST : /[a-zA-Z0-9_]+/  { $.equal($1); };
};
```

Строковая форма литеральной формы типа данных CS определена в основном для целей настоящей спецификации. Она является строковым представлением кода из системы кодирования, заданной контекстом, в котором используется тип данных CS. По самому литералу нельзя определить систему кодирования codeSystem или ее версию codeSystemVersion, поэтому литерал можно использовать, только если известен контекст.

B.2.9 Тип данных CodedValue (CV) (специализация типа данных CE)

Кодированные данные, для которых указаны только код, система кодирования, а также необязательные имя для вывода на экран и исходный текст. Используется только как тип свойств других типов данных.

Таблица B.14 — Сводка свойств типа данных CodedValue

Имя	Тип	Описание
code	ST	Символ кода, определенный в системе кодирования. Например, «784.0» является символом кода головной боли «784.0», определенным в системе кодирования МКБ-9
codeSystem	UID	Указывает систему кодирования, в которой определен код
codeSystemName	ST	Общее имя системы кодирования
codeSystemVersion	ST	Если применим, дескриптор версии, определенный специально для данной системы кодирования
displayName	ST	Имя или название кода, под которым система-отправитель показывает значение кода своим пользователям
originalText	ED	Текст или фраза, используемые в качестве основы для кодирования

```
type CodedValue alias CV specializes CE {
  ST  code;
  UID  codeSystem;
  ST  codeSystemName;
  ST  codeSystemVersion;
  ST  displayName;
  ED  originalText;
};
```

В наиболее приемлемом сценарии использования типа данных CV требуется возвратить единственный код. Поэтому этот тип данных не должен использоваться, если для данного значения желательно передать несколько альтернативных кодов. Этот тип данных может использоваться как с квалификатором расширяемости домена «CNE» (coded, non-extensible — кодируемый, не расширяемый), так с квалификатором CWE (coded — кодируемый, расширяемый).

B.2.9.1 Свойство code: ST (унаследовано от типа данных CD)

Определение: символ кода, определенный в системе кодирования. Например, «784.0» является символом кода головной боли «784.0», определенным в системе кодирования МКБ-9.

Не исключительное значение типа данных CV имеет непустое свойство code, значением которого является строка символов, определенная в системе кодирования, идентифицированной в свойстве codeSystem. И наоборот, значение типа CV, у которого свойство code не имеет значения или имеет значение, не принадлежащее указанной системе кодирования, считается исключительным значением (пустым значением (NULL) с типом пустоты «other»).

```
invariant(CV x)
    where x.nonNull {
  x.code.nonNull;
};
```

B.2.9.2 Свойство codeSystem: UID (унаследовано от типа данных CD)

Указывает систему кодирования, в которой определено значение свойства code.

Идентификатор системы кодирования должен иметь тип данных UID, позволяющий однозначно указать стандартные системы кодирования HL7, другие стандартные системы кодирования, а также местные системы кодирования. Комитет HL7 должен присваивать идентификатор типа UID каждой из своих таблиц кодов, а также внешним стандартным системам кодирования, которые используются в стандартах HL7. На местах должны использовать свои объектные идентификаторы ИСО (тип данных OID), с помощью которых можно сконструировать глобально уникальные идентификаторы местных систем кодирования.

Под ветвью комитета HL7, 2.16.840.1.113883, подветви 5 и 6 содержат соответственно идентификаторы стандартных систем кодирования HL7 и внешних систем кодирования. Эти ветви ведутся техническим комитетом HL7 Vocabulary Technical Committee.

Не исключительное значение типа CD (то есть значение типа CD с непустым свойством code) имеет непустое свойство codeSystem, указывающее систему понятий, в которой определено значение свойства code. Другими словами, если есть код, должна быть и система кодирования.

Примечание — Хотя для каждого непустого значения типа CD определена конкретная система кодирования, в некоторых обстоятельствах представление значения типа CD в соответствии со спецификацией реализуемой технологии не нуждается в явном упоминании системы кодирования. Например, когда контекст подразумевает одну и только одну систему кодирования, то ее явное указание стало бы избыточным. Однако в таком случае свойство codeSystem принимает контекстно-зависимое значение по умолчанию и не является пустым.

```
invariant(CV x)
    where x.code.nonNull {
  x.codeSystem.nonNull;
};
```

Причина пустоты «other» у исключительного значения типа CD указывает, что понятие не может быть закодировано в указанной системе кодирования. Эта система кодирования, в которой нет такого исключительного понятия, должна быть указана в свойстве codeSystem.

Некоторые домены кодов квалифицированы таким образом, что они могут включать в себя некоторые части подходящей местной системы кодирования, не являющиеся парафразами стандартной системы кодирования (coded with extensibility, CWE — кодированные с расширением.) Если поле с квалификатором «CWE» действительно содержит такой местный код, то в свойстве системы кодирования должен быть указан идентификатор местной системы кодирования, из которой взят этот код. Однако в доменах с квалификатором «CWE» местный код является допустимым членом домена, поэтому использование местного кода не является ни ошибкой, ни исключительным значением (с причиной пустоты «other») в смысле настоящей спецификации.

```
invariant(CV x)
    where x.other {
  x.code.other;
  x.codeSystem.nonNull;
};
```

B.2.9.3 Свойство codeSystemName: ST (унаследовано от типа данных CD)

Общее имя системы кодирования.

Имя системы кодирования не используется для вычислений. Оно может быть указано для облегчения интерпретации человеком значений свойств code и codeSystem. Предполагается, хотя и не является обязательным, что в спецификациях реализуемой технологии будет предусматриваться указание значения свойства codeSystemName в качестве аннотации к идентификатору UID, рассчитанной на восприятие человеком.

Системы, соответствующие стандарту HL7, не должны функционально полагаться на значение свойства codeSystemName. Это значение не должно модифицировать значение свойства codeSystem и не может существовать без значения свойства codeSystem.

```
invariant(CV x) {
  x.codeSystemName.nonNull.implies(x.codeSystem.nonNull);
};
```

B.2.9.4 Свойство codeSystemVersion: ST (унаследовано от типа данных CD)

Если применим, дескриптор версии, определенный специально для данной системы кодирования.

Для каждой внешней системы кодирования в стандарте HL7 будет указано, как формируется строка со значением версии. Если для конкретной системы кодирования такое указание отсутствует, то для такой системы обозначение версии не имеет определенного значения.

Различные версии одной и той же системы кодирования должны быть совместимыми. Если система кодирования изменена несовместимым образом, то она представляет собой другую систему кодирования, а не другую версию, как бы издатель это не называл.

Например, издатель классификаций МКБ-9 и МКБ-10 назвал эти системы кодирования «9-м пересмотром» и «10-м пересмотром». Однако МКБ-10 представляет собой полное изменение кодов МКБ, и не является обратно совместимой. Поэтому в целях настоящей спецификации МКБ-9 и МКБ-10 рассматриваются как разные системы кодирования, а не как разные версии одной системы кодирования. Напротив, когда версия «1.0j» системы кодирования LOINC была обновлена до версии «1.0k», комитет HL7 рассматривал это как смену версий, поскольку новая версия была обратно совместимой с предыдущей.

```
invariant(CV x) {
  x.codeSystemVersion.nonNull.implies(x.codeSystem.nonNull);
};
```

B.2.9.5 Свойство displayName: ST (унаследовано от типа данных CD)

Имя или название кода, под которым система-отправитель показывает значение кода своим пользователям.

Свойство displayName включено как для удобства интерпретации человеком значения кода, так и для документирования имени, используемого для изображения понятия пользователю. Оно не имеет функционального значения, не может существовать без кода и никогда не должно модифицировать смысл кода.

Примечания

1 В своих словарных доменах стандарты HL7 предусматривают атрибут «печатаемого имени» (print name). Значения этого атрибута могут использоваться в качестве значений свойства displayName.

2 Имена, предусмотренные для кодов, не могут менять смысл кодированного значения. Поэтому они не должны предоставляться пользователю прикладной системы-получателя, пока не будет уверенности в том, что такое имя адекватно отражает понятие, соответствующее кодированному значению. При коммуникациях нельзя просто полагаться на имя кодированного значения. Основная цель этого имени в обеспечении возможности отладки единиц данных в протоколах HL7 (например, сообщений).

```
invariant(CV x) {
  x.displayName.nonNull.implies(x.code.nonNull);
};
```

B.2.9.6 Свойство originalText: ED

Текст или фраза, используемые в качестве основы для кодирования.

Исходный текст появляется в тех случаях, когда код был присвоен информации не ее источником, а специальным кодировщиком уже после ее появления. Тогда при создании дескриптора понятия исходный текст может существовать без кода.

Примечание — Хотя кодировка пост-фактум часто осуществляется по информации, представленной как свободный текст, например, в форме документов, сканированных изображений или диктовки, мультимедийные данные явным образом не разрешены в качестве исходного текста. Кроме того, свойство исходного текста originalText не может означать ссылку на весь исходный документ. Связи между различными артефактами медицинской информации (например, документом и кодированным результатом) не входят в область применения настоящей спецификации и рассматриваются в других стандартах HL7. Исходный текст должен представлять собой извлечение из оригинального источника, а не точную копию его содержания или указатель на этот источник. Поэтому исходный текст должен представляться в неформатированном виде.

Значение типа CV могут иметь непустое свойство исходного текста, даже если свойство code имеет пустое значение и, следовательно, является исключительным. В этом случае свойство originalText является именем или описанием понятия, которое не было закодировано. Такие исключительные значения могут также содержать преобразования translation. Такие преобразования обеспечивают непосредственное кодирование понятия, описанного в свойстве originalText.

Тип данных CV может быть понижен до типа данных ST. В этом случае значение типа ST будет представлено только свойством originalText значения типа CV.

```
invariant(CV x)
    where x.originalText.nonNull {
  ((ST)x).equal(x.originalText);
};
```

B.2.9.7 Свойство translation: SET<CD> (фиксированное)

```
invariant(CV x) {
  x.translation.notApplicable;
};
```

B.2.9.8 Свойство qualifier: LIST<CR> (фиксированное)

```
invariant(CV x) {
  x.qualifier.notApplicable;
};
```

B.2.10 Тип данных CodedOrdinal (CO) (специализация типа данных CV)

Кодированные данные, принадлежащие системе кодирования с упорядоченными кодами. В типе данных CO добавлена семантика упорядоченности; таким образом, в моделях, использующих такие домены, могут быть определены элементы, содержащие утверждения о порядке следования терминов в домене.

```
type CodedOrdinal alias CO specializes CV {
  BL lessOrEqual(CO o);
  BL lessThan(CO o);
  BL greaterThan(CO o);
  BL greaterOrEqual(CO o);
};
```

Относительные порядок значений типа CO не обязан очевидно вытекать из их литерального представления. Предполагается, что приложение будет определять упорядоченность этих значений по некоторой таблице.

B.2.10.1 Свойство lessOrEqual: BL (отношение «меньше или равно»)

Отношение порядка основано на свойстве lessOrEqual, которое считается примитивным в настоящей спецификации.

Из него могут быть выведены все другие отношения порядка. Так как свойство lessOrEqual является примитивным, то оно может задавать и частичную упорядоченность.

Отношения порядка обычно распространяются на одну систему кодирования.

B.2.10.2 Свойство lessThan: BL (отношение «меньше»)

```
invariant(CO x, y)
    where x.nonNull.and(y.nonNull) {
  x.lessThan(y).equal(y.lessOrEqual(x).and(x.equal(y).not));
};
```

B.2.10.3 Свойство greaterThan: BL (отношение «больше»)

```
invariant(CO x, y)
    where x.nonNull.and(y.nonNull) {
  x.greaterThan(y).equal(y.lessThan(x));
};
```

B.2.10.4 Свойство greaterOrEqual: BL (отношение «больше или равно»)

```
invariant(CO x, y)
    where x.nonNull.and(y.nonNull) {
  x.greaterOrEqual(y).equal(y.lessOrEqual(x));
};
```

B.2.11 Тип данных CodedWithEquivalents (CE) (специализация типа данных CD)

Кодированные данные, состоящие из кодированного значения и из необязательных кодированных значений, идентифицирующих то же самое понятие. Этот тип данных используется, когда могут существовать альтернативные коды.

Таблица B.15 — Сводка свойств типа данных CodedWithEquivalents

Имя	Тип	Описание
code	ST	Символ кода, определенный в системе кодирования. Например, «784.0» является символом кода головной боли «784.0», определенным в системе кодирования МКБ-9
codeSystem	UID	Указывает систему кодирования, в которой определен код
codeSystemName	ST	Общее имя системы кодирования
codeSystemVersion	ST	Если применим, дескриптор версии, определенный специально для данной системы кодирования

Окончание таблицы В.15

Имя	Тип	Описание
displayName	ST	Имя или название кода, под которым система-отправитель показывает значение кода своим пользователям
originalText	ED	Текст или фраза, используемые в качестве основы для кодирования
translation	SET<CD>	Множество других дескрипторов понятия, которые преобразуют данный дескриптор понятия в другие системы кодирования

```
type CodedWithEquivalents alias CE specializes CD {
  ST          code;
  UID         codeSystem;
  ST          codeSystemName;
  ST          codeSystemVersion;
  ST          displayName;
  ED          originalText;
  SET<CV>     translation;
};
```

Тип данных CE используется в сценариях, когда могут существовать альтернативные коды, полезные для получателя. Экземпляр типа данных CE содержит основное значение кода и может содержать множество альтернативных или эквивалентных представлений.

В.2.11.1 Свойство code: ST (унаследовано от типа данных CD)

Символ кода, определенный в системе кодирования. Например, «784.0» является символом кода головной боли «784.0», определенным в системе кодирования МКБ-9.

Не исключительное значение типа данных CD имеет непустое свойство code, значением которого является строка символов, определенная в системе кодирования, идентифицированной в свойстве codeSystem. И наоборот, значение типа CD, у которого свойство code не имеет значения или имеет значение, не принадлежащее указанной системе кодирования, считается исключительным значением (пустым значением (NULL) с типом пустоты «other»).

```
invariant(CE x)
    where x.nonNull {
  x.code.nonNull;
};
```

В.2.11.2 Свойство codeSystem: UID (унаследовано от типа данных CD)

Указывает систему кодирования, в которой определено значение свойства code.

Идентификатор системы кодирования должен иметь тип данных UID, позволяющий однозначно указать стандартные системы кодирования HL7, другие стандартные системы кодирования, а также местные системы кодирования. Комитет HL7 должен присваивать идентификатор типа UID каждой из своих таблиц кодов, а также внешним стандартным системам кодирования, которые используются в стандартах HL7. На местах должны использовать свои объектные идентификаторы ИСО (тип данных OID), с помощью которых можно сконструировать глобально уникальные идентификаторы местных систем кодирования.

Под ветвью комитета HL7, 2.16.840.1.113883, подветви 5 и 6 содержат соответственно идентификаторы стандартных систем кодирования HL7 и внешних систем кодирования. Эти ветви ведутся техническим комитетом HL7 Vocabulary Technical Committee.

Не исключительное значение типа CE (то есть значение типа CE с непустым свойством code) имеет непустое свойство codeSystem, указывающее систему понятий, в которой определено значение свойства code. Другими словами, если есть код, должна быть и система кодирования.

Примечание — Хотя для каждого непустого значения типа CE определена конкретная система кодирования, в некоторых обстоятельствах представление значения типа CE в соответствии со спецификацией реализуемой технологии не нуждается в явном упоминании системы кодирования. Например, когда контекст подразумевает одну и только одну систему кодирования, то ее явное указание стало бы избыточным. Однако в таком случае свойство codeSystem принимает контекстно-зависимое значение по умолчанию и не является пустым.

```
invariant(CE x)
    where x.code.nonNull {
  x.codeSystem.nonNull;
};
```

Причина пустоты «other» у исключительного значения типа CE указывает, что понятие не может быть закодировано в указанной системе кодирования. Эта система кодирования, в которой нет такого исключительного понятия, должна быть указана в свойстве codeSystem.

Некоторые домены кодов квалифицированы таким образом, что они могут включать в себя некоторые части подходящей местной системы кодирования, не являющиеся парафразами стандартной системы кодирования (coded with extensibility, CWE — кодированные с расширением). Если поле с квалификатором «CWE» действительно содержит такой местный код, то в свойстве системы кодирования должен быть указан идентификатор местной системы кодирования, из которой взят этот код. Однако в доменах с квалификатором «CWE» местный код является допустимым членом домена, поэтому использование местного кода не является ни ошибкой, ни исключительным значением (с причиной пустоты «other») в смысле настоящей спецификации.

```
invariant(CE x)
    where x.other {
  x.code.other;
  x.codeSystem.nonNull;
};
```

B.2.11.3 Свойство codeSystemName: ST (унаследовано от типа данных CD)

Общее имя системы кодирования.

Имя системы кодирования не используется для вычислений. Оно может быть указано для облегчения интерпретации человеком значений свойств code и codeSystem. Предполагается, хотя и не является обязательным, что в спецификациях реализуемой технологии будет предусматриваться указание значения свойства codeSystemName в качестве аннотации к идентификатору UID, рассчитанной на восприятие человеком.

Системы, соответствующие стандарту HL7, не должны функционально полагаться на значение свойства codeSystemName. Это значение не должно модифицировать значение свойства codeSystem и не может существовать без значения свойства codeSystem.

```
invariant(CE x) {
  x.codeSystemName.nonNull.implies(x.codeSystem.nonNull);
};
```

B.2.11.4 Свойство codeSystemVersion: ST (унаследовано от типа данных CD)

Если применим, дескриптор версии, определенный специально для данной системы кодирования.

Для каждой внешней системы кодирования в стандарте HL7 будет указано, как формируется строка со значением версии. Если для конкретной системы кодирования такое указание отсутствует, то для такой системы обозначение версии не имеет определенного значения.

Различные версии одной и той же системы кодирования должны быть совместимыми. Если система кодирования изменена несовместимым образом, то она представляет собой другую систему кодирования, а не другую версию, как бы издатель это не называл.

Например, издатель классификаций МКБ-9 и МКБ-10 назвал эти системы кодирования «9-м пересмотром» и «10-м пересмотром». Однако МКБ-10 представляет собой полное изменение кодов МКБ, и не является обратно совместимой. Поэтому в целях настоящей спецификации МКБ-9 и МКБ-10 рассматриваются как разные системы кодирования, а не как разные версии одной системы кодирования. Напротив, когда версия «1.0j» системы кодирования LOINC была обновлена до версии «1.0k», комитет HL7 рассматривал это как смену версий, поскольку новая версия была обратно совместимой с предыдущей.

```
invariant(CE x) {
  x.codeSystemVersion.nonNull.implies(x.codeSystem.nonNull);
};
```

B.2.11.5 Свойство displayName: ST (унаследовано от типа данных CD)

Имя или название кода, под которым система-отправитель показывает значение кода своим пользователям.

Свойство displayName включено как для удобства интерпретации человеком значения кода, так и для документирования имени, используемого для изображения понятия пользователю. Оно не имеет функционального значения, не может существовать без кода и никогда не должно модифицировать смысл кода.

Примечания

1 В своих словарных доменах стандарты HL7 предусматривают атрибут «печатаемого имени» (print name). Значения этого атрибута могут использоваться в качестве значений свойства displayName.

2 Имена, предусмотренные для кодов, не могут менять смысл кодированного значения. Поэтому они не должны предоставляться пользователю прикладной системы-получателя, пока не будет уверенности в том, что такое имя адекватно отражает понятие, соответствующее кодированному значению. При коммуникациях нельзя просто полагаться на имя кодированного значения. Основная цель этого имени в обеспечении возможности отладки единиц данных в протоколах HL7 (например, сообщений).

```
invariant(CE x) {
  x.displayName.nonNull.implies(x.code.nonNull);
};
```

B.2.11.6 Свойство originalText: ED (унаследовано от типа данных CD)

Текст или фраза, используемые в качестве основы для кодирования.

Исходный текст появляется в тех случаях, когда код был присвоен информации не ее источником, а специальным кодировщиком уже после ее появления. Тогда при создании дескриптора понятия исходный текст может существовать без кода.

Примечание — Хотя кодировка пост-фактум часто осуществляется по информации, представленной как свободный текст, например, в форме документов, сканированных изображений или диктовки, мультимедийные данные явным образом не разрешены в качестве исходного текста. Кроме того, свойство исходного текста originalText не может означать ссылку на весь исходный документ. Связи между различными артефактами медицинской информации (например, документом и кодированным результатом) не входят в область применения настоящей спецификации и рассматриваются в других стандартах HL7. Исходный текст должен представлять собой извлечение из оригинального источника, а не точную копию его содержания или указатель на этот источник. Поэтому исходный текст должен представляться в неформатированном виде.

Значение типа CD могут иметь непустое свойство исходного текста, даже если свойство code имеет пустое значение. В этом случае свойство originalText является именем или описанием понятия, которое не было закодировано. Такие исключительные значения типа CD могут также содержать преобразования translation. Такие преобразования обеспечивают непосредственное кодирование понятия, описанного в свойстве originalText.

Тип данных CE может быть понижен до типа данных ST. В этом случае значение типа ST будет представлено только свойством originalText значения типа CE.

```
invariant(CE x)
    where x.originalText.nonNull {
  ((ST)x).equal(x.originalText);
};
```

B.2.11.7 Свойство translation: SET<CD> (унаследовано от типа данных CD)

Множество других дескрипторов понятия, которые преобразуют данный дескриптор понятия в другие системы кодирования.

Свойство translation представляет собой множество других значений типа CD, каждое из которых преобразует первое значение типа CD в другую систему кодирования. Каждый элемент множества преобразований был получен с помощью преобразования первого значения типа CD. Каждое преобразование в свою очередь может также содержать преобразования. Таким образом, если код многократно преобразуется, то информация о том, какой код послужил входным параметром каждого преобразования, сохранится.

Примечание — Преобразования являются квазисинонимами одного понятия реального мира. Предполагается, что каждое преобразование в этом множестве выражает то же самое понятие «другими словами». Однако между двумя структурно различными системами кодирования редко существует точная синонимия. Поэтому не все преобразования будут абсолютно точными.

B.2.11.8 Свойство qualifier: LIST<CR> (фиксированное)

```
invariant(CE x) {
  x.qualifier.notApplicable;
};
```

B.2.12 Тип данных CharacterStringWithCode (SC) (специализация типа данных ST)

Строка символов, с которой может быть связан необязательный код. Если код присутствует, обязательно должен присутсвовать и текст. В качестве кода нередко используется местный код.

Таблица B.16 — Сводка свойств типа данных CharacterStringWithCode

Имя	Тип	Описание
code	CE	Код, представляющий строку символов. Например, такая строка может быть сообщением пользователю, выбираемым из каталога сообщений, а код представляет собой идентификатор сообщения в этом каталоге

```
type CharacterStringWithCode alias SC specializes ST {
  CE code;
};
```

Тип данных SC используется в случаях, когда кодирование является скорее исключением, чем правилом (например, сообщения пользователю являются в основном текстовыми, и пользователю важно знать печатаемое сообщение. Однако иногда сообщения берутся из каталога заранее заготовленных текстов, ссылки на которые содержатся в значении типа SC.

Любое непустое значение типа SC МОЖЕТ иметь код, но код НЕ ДОЛЖЕН быть указан без текста.

```
invariant(SC x)
    where x.nonNull {
  x.code.nonNull.implies(x.notEmpty);
};
```

B.2.12.1 Свойство code: CE

Код, представляющий строку символов. Например, такая строка может быть сообщением пользователю, выбираемым из каталога сообщений, а код представляет собой идентификатор сообщения в этом каталоге.

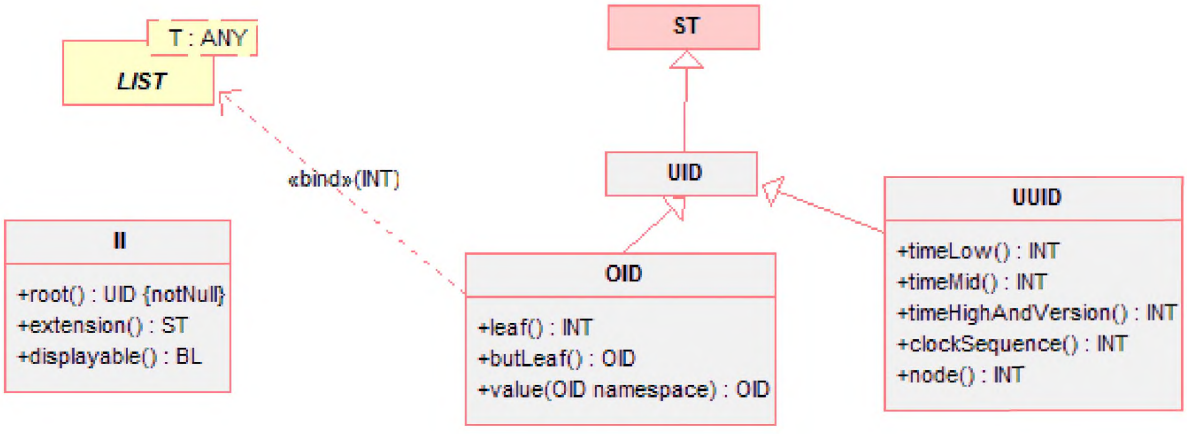


Рисунок B.5 — Типы данных экземпляров идентификаторов

B.2.13 Тип данных UniqueIdentifierString (UID) (специализация типа данных ST)

Тип данных UniqueIdentifierString описывает строку, являющуюся идентификатором объекта, обладающим свойством глобальной уникальности и не меняющимся с течением времени. Допустимые форматы, значения и процедуры этого типа данных строго контролируются комитетом HL7. В настоящее время пользовательскими идентификаторами могут быть определенные символьные представления объектных идентификаторов ИСО (тип данных OID) и универсально уникальные идентификаторы Распределенного компьютерного окружения DCE (тип данных UUID). Комитет HL7 также оставляет за собой право присваивать другие формы универсально уникальных идентификаторов (резервные схемы идентификаторов RUID), например, мнемонические идентификаторы систем кодирования.

Основным назначением уникальных идентификаторов (УИД) является идентификация объектов, обладающая свойством глобальной уникальности и не меняющаяся с течением времени. Формы УИД, будь то значения типов данных OID, UUID или RUID, совершенно не существенны. Пока они используются в стандартах HL7, единственное, что можно делать с УИД, это обозначать им объект, которому он присвоен. Сравнение УИД осуществляется по их литеральным формам, то есть если два УИД имеют идентичные литералы, то эти УИД считаются обозначениями одного и того же объекта. Если литералы двух УИД не идентичны, то они МОГУТ не обозначать один и тот же объект.

```
type UniqueIdentifierString alias UID specializes ST { };
```

Между различными допустимыми формами УИД нет никаких семантических различий. Разные формы не отличаются по компоненту, входящему в строку идентификатора или смежному с ней.

Хотя в настоящей спецификации и не проводится семантическое различие между разными формами уникальных идентификаторов, существуют определеннные отличия в том, как эти идентификаторы образуются и как ими управляют, что и служит единственной причиной выделения подтипов УИД для каждого варианта.

B.2.14 Тип данных объектного идентификатора ИСО ObjectIdentifier (OID) (специализация типа данных UID)

Глобально уникальная строка, представляющая объектный идентификатор ИСО (ОИД) в форме, образованной только из цифр и точек (например, «2.16.840.1.113883.3.1»). Согласно стандарту ИСО, идентификаторы ОИД являются путями в структуре дерева, где крайнее левое число означает корень дерева, а крайнее правое число — его лист.

Каждая ветвь под корнем соответствует организации, уполномоченной присваивать идентификаторы. В свою очередь, каждая из этих организаций может назначить свою совокупность уполномоченных организаций, действующих под их надзором, и так далее. В конечном счете одна из этих организаций присваивает уникальный (для себя) номер, соответствующий корневому узлу дерева. Этот лист может представлять уполномоченную организацию (в данном случае корневой ОИД, идентифицирующий организацию) или экземпляр объекта. Каждая уполномоченная организация владеет пространством имен, образованным ее поддеревом.

ОИД является предпочтительной схемой уникальных идентификаторов. Идентификаторы ОИД должны использоваться во всех случаях, за исключением ситуаций, когда выполняются критерии применимости других схем.

В разделе 28 стандарта ИСО/МЭК 8824:1990(E) объектный идентификатор определен следующим образом:

«28.9 Семантика значения объектного идентификатора определяется ссылкой на дерево объектных идентификаторов. Корень этого дерева соответствует стандарту [ИСО/МЭК 8824], а вершины (узлы) — уполномоченным по регистрации, отвечающим за выделение дуг [то есть ветвей], исходящих из этой вершины. Каждая дуга этого дерева помечается компонентом объектного идентификатора, являющегося [целым значением]. Каждому идентифицируемому объекту выделяется строго одна вершина (обычно являющаяся листом), которая не выделяется другим объектам (того же или другого типа). Таким образом, объект однозначно и недвусмысленно идентифицируется последовательностью целых значений (значений компонентов объектного идентификатора), помечающих дуги на пути от корня до выделенной объекту вершины.

28.10 Значение объектного идентификатора является семантически упорядоченным списком значений его компонентов. Начиная с корня дерева объектных идентификаторов, каждое значение компонента объектного идентификатора объекта отождествляет дугу в этом дереве. Последнее значение компонента объектного идентификатора отождествляет дугу, ведущую к вершине, которая была назначена объекту. Это и есть объект, который идентифицируется значением объектного идентификатора. [...]»

```
type ObjectIdentifier alias OID specializes UID, LIST<INT> {
        INT  leaf;
        OID  butLeaf;
        OID  value(namespace OID);
  literal ST;
};
```

В соответствии со стандартом ИСО/МЭК 8824, объектный идентификатор является упорядоченным списком значений его компонентов, представленных целыми числами. Эти значения компонентов упорядочены таким образом, что корень дерева объектных идентификаторов является головой списка, а за ним следуют все дуги вплоть до листа, представляющего собой информационный объект, идентифицируемый полученным ОИД. Тот факт, что тип данных объектного идентификатора OID является специализацией типа данных LIST<INT>, отражает представление пути от корня до листа, образованного значениями компонентов объектного идентификатора.

Свойства leaf и butLeaf представляют собой взгляд на объектный идентификатор с другого конца. Свойство leaf является последним из значений компонентов объектного идентификатора в списке, а свойство butLeaf (кроме листа) представляет собой весь ОИД, за исключением листа, указанного в свойстве leaf. В известном смысле

значение свойства leaf является значением идентификатора, а значение свойства butLeaf — идентификатором пространства имен, в котором значение этого идентификатора является уникальным и осмысленным.

Однако то, какая часть ОИД рассматривается как идентификатор значения, а какая — как идентификатор пространства имен, может трактоваться по-разному. В общем случае любая левая часть последовательности компонентов ОИД может рассматриваться как идентификатор пространства имен, а оставшаяся правая часть — как уникальное и осмысленное значение идентификатора. Эта точка зрения представлена свойством value в качестве аргумента в пространстве имен ОИД.[37)].

```
invariant(OID x)
    where x.nonNull {
  x.notEmpty;
  x.tail.isEmpty.implies(x.leaf.equal(x.tail));
  x.tail.notEmpty.implies(x.leaf.equal(x.tail.leaf);
  x.tail.isEmpty.implies(x.butLeaf.isNull);
  x.tail.notEmpty.implies(x.butLeaf.head.equal(x.head)
      .and(x.butLeaf.tail.equal(x.butLeaf(x.tail))));
  forall(OID v; OID n) where v.equal(x.value(n)) {
    n.isEmpty.implies(v.equal(x));
    n.notEmpty.implies(v.equal(x.value(n.tail)));
    };
};
```

B.2.14.1 Объектные идентификаторы, присваиваемые комитетом HL7

Комитет HL7 должен создать регистр ОИД и по запросу присваивать ОИД в своей ветви пользователям и поставщикам, сотрудничающим с комитетом. Кроме того, комитет HL7 должен присвоить ОИД публичным организациям, присваивающим идентификаторы на национальном уровне в США (например, U.S. State driver license bureaus, U.S. Social Security Administration, HIPAA Provider ID registry и т. д.) и на международном уровне (например, органам социального страхования или регистрам граждан других стран). Для этих организаций должны использоваться ОИД, присвоенные комитетом HL7, даже если этим организациям присвоены ОИД из других источников.

Присваивая ОИД третьим сторонам или сущностям, комитет HL7 длжен определить, не присвоены ли этим сущностям ОИД из других источников. В случае наличия таковых комитет HL7 должен занести соответствующий ОИД в свой каталог, но не присваивать собственный ОИД в ветви, контролируемой комитетом. По возможности комитет HL7 должен информировать третью сторону о том, что ей присвоен ОИД в ветви комитета HL7.

Хотя комитет HL7 должен провести тщательный поиск ОИД, уже присвоенных третьим сторонам, в отсутствие глобального механизма регистрации ОИД нельзя быть абсолютно уверенным, что третьей стороне еще не присвоен ОИД. Кроме того, повторное присвоение может произойти из других источников. В случае, когда комитету HL7 становится известно о наличии другого ОИД у третьей стороны, он должен предпринять определенные усилия для разрешения этой ситуации. Тем не менее для обеспечения интероперабельности тот ОИД, что присвоен комитетом HL7, должен быть предпочтительным.

Хотя большинство владельцев ОИД «конструируют» поддерево своего пространства имен, используя определенные принципы, общего способа придания смысла отдельным компонентам ОИД не существует. Комитет HL7 не стандартизует структуру поддеревьев пространства имен и не требует этого от других организаций. Тем не менее владелец ОИД или иное лицо, владеющее информацией о логической структуре компонентов ОИД может использовать это знание для извлечения сведений об ассоциированом объекте. Однако такой подход не может быть обобщен.

Пример дерева ОИД показан на рисунке .6–. Комитету HL7 присвоен ОИД 2.16.840.1.113883.

Интерфейс, предложенный в стандарте HL7, не должен полагаться на какое-либо знание о структуре ОИД, присваивание которого комитет HL7 не может контролировать.

[37)] Представление объектного идентификатора ISO в виде пары значение/пространство имен отражает важную семантическую связь, а именно, отношение между идентификатором и организацией, присваивающей идентификаторы (= пространство имен), являющееся очень важны в информационных системах здравоохранения вообще и в стандарте HL7 v2.x в частности.

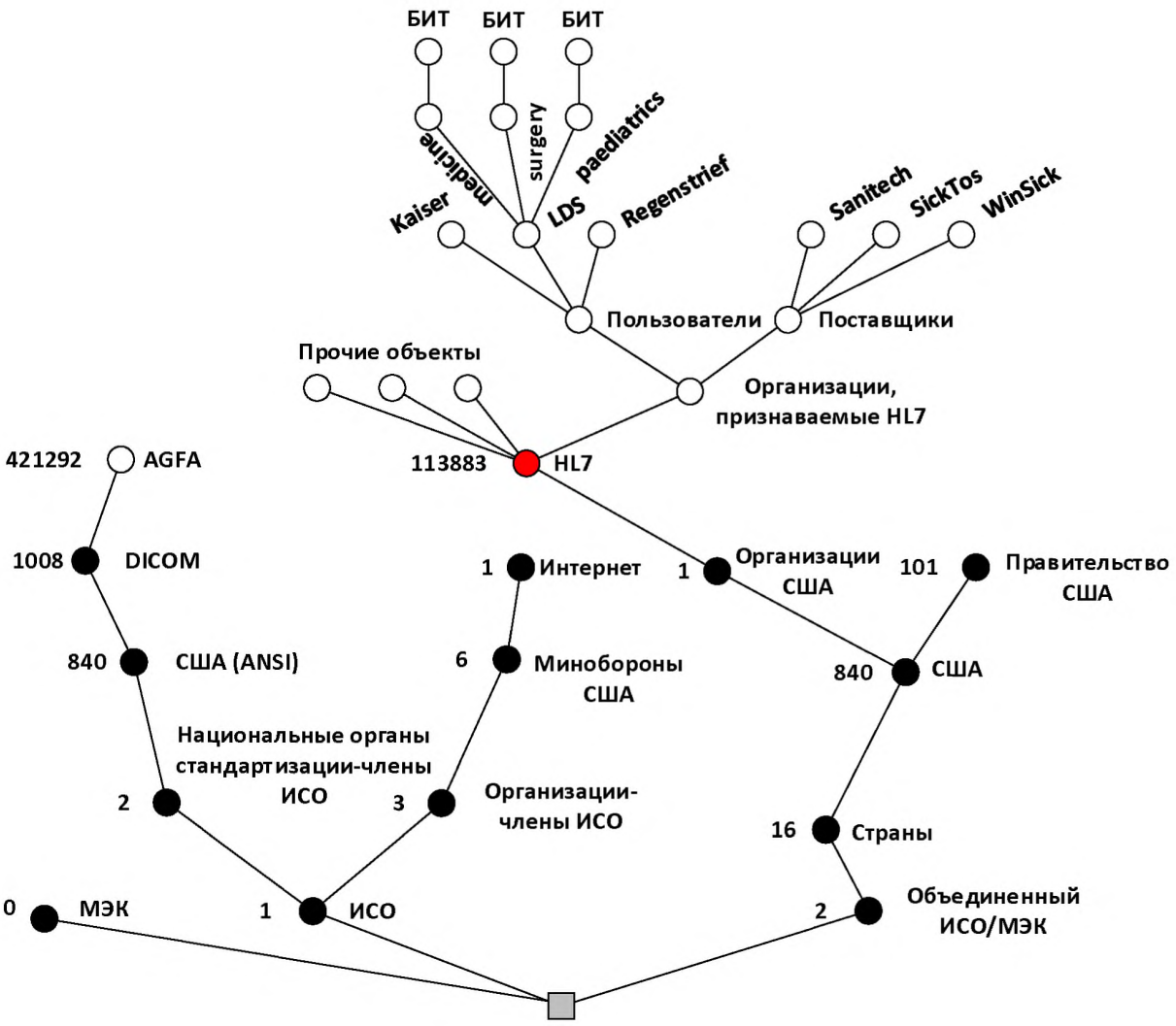


Рисунок В.6 — Пример дерева ОИД

В.2.14.2 Литеральная форма

Структурное определение типа данных ОИД представлено только в целях соответствия спецификации объектных идентификаторов. В стандартах HL7 идентификаторы ОИД используются только как строки УИД, то есть строка литерала является единственным передаваемым объектом, и получатель только ее и должен рассматривать при работе с идентификаторами ОИД в рамках спецификации HL7.

```
OID.literal ST {
  OID : INT "." OID { $.head.equal($1);
                      $.tail.equal($3); }
      | INT         { $.head.equal($1);
                      $.tail.isEmpty; }
};
```

Для совместимости со стандартом DICOM длина литеральной формы ОИД не должна превышать 64 символов (см. раздел 9 части 5 стандарта DICOM).

В.2.15 Тип данных UniversalUniqueIdentifier (UUID) (специализация типа данных UID)

Глобально уникальная строка, представляющая Универсальный уникальный идентификатор (UUID) Распределенной вычислительной среды (DCE) в общем формате, состоящем из пяти групп шестнадцатеричных цифр, имеющих соответственно 8, 4, 4, 4 и 12 позиций и разделенных дефисами.

Как идентификатор UUID, так и его строковое представление определены в документе CDE 1.1 Remote Procedure Call specification, Appendix A организации Open Group.

Идентификаторы UUID присваиваются на основе MAC-адресов Ethernet, момента времени создания и некоторого случайного компонента. Предполагается, что это сочетание позволяет генерировать достаточно уникальные идентификаторы без какой-либо организационной политики присваивания идентификаторов (если не считать организацию присвоения MAC-адресов).

Идентификаторы UUID не являются предпочтительной схемой для использования в качестве УИД в стандартах HL7. Они могут использоваться, когда идентификаторы присваиваются отдельным объектам (например, идентификаторы экземпляров сущностей, идентификаторы событий действия и т. д.). Для объектов, описывающих классы сущностей или событий (например, каталога предметов), предпочтительной схемой идентификации являются ОИД.

```
type UniversalUniqueIdentifier alias UUID specializes UID {
  INT timeLow;
  INT timeMid;
  INT timeHighAndVersion;
  INT clockSequence;
  INT node;
};
```

B.2.15.1 Литеральная форма

Структурное определение идентификаторов UUID представлено только в целях соответствия спецификации универсально уникальных идентификаторов. В стандартах HL7 идентификаторы UUID используются только как строки УИД, то есть строка литерала является единственным передаваемым объектом, и получатель только ее и должен рассматривать при работе с идентификаторами UUID в рамках спецификации HL7.

Литеральная форма идентификаторов UUID определена в соответствии с их исходной спецификацией. Следует иметь в виду, что в стандартах HL7 уникальные идентификаторы чувствительны к регистру, поэтому при использовании в соответствии со стандартами HL7 шестнадцатеричные цифры A-F в этих идентификаторах должны быть приведены к верхнему регистру.

```
UUID.literal ST {
  UUID      : hex8 "-" hex4 "-" hex4 "-" hex4 "-" hex12 {
                $.timeLow.equal($1);
                $.timeMid.equal($3);
                $.timeHighAndVersion.equal($5);
                $.clockSequence.equal($7);
                $.node.equal($9);
                }

  INT hex4  : hexDigit hexDigit hexDigit hexDigit {
                $.equal($1.times(16).plus($2)
                .times(16).plus($3)
                .times(16).plus($4);
                }

  INT hex8  : hexDigit hexDigit hexDigit hexDigit
              hexDigit hexDigit hexDigit hexDigit {
                $.equal($1.times(16).plus($2)
                .times(16).plus($3)
                .times(16).plus($4)
                .times(16).plus($5)
                .times(16).plus($6)
                .times(16).plus($7)
                .times(16).plus($8);
                }
```

```
INT hex12: hexDigit hexDigit hexDigit hexDigit
           hexDigit hexDigit hexDigit hexDigit
           hexDigit hexDigit hexDigit hexDigit {
         $.equal($1.times(16).plus($2)
         .times(16).plus($3)
         .times(16).plus($4)
         .times(16).plus($5)
         .times(16).plus($6)
         .times(16).plus($7)
         .times(16).plus($8)
         .times(16).plus($9)
         .times(16).plus($10)
         .times(16).plus($11)
         .times(16).plus($12);
         }

INT hexDigit : "0"   { $.equal(0); }
             | "1"   { $.equal(1); }
             | "2"   { $.equal(2); }
             | "3"   { $.equal(3); }
             | "4"   { $.equal(4); }
             | "5"   { $.equal(5); }
             | "6"   { $.equal(6); }
             | "7"   { $.equal(7); }
             | "8"   { $.equal(8); }
             | "9"   { $.equal(9); }
             | "A"   { $.equal(10); }
             | "B"   { $.equal(11); }
             | "C"   { $.equal(12); }
             | "D"   { $.equal(13); }
             | "E"   { $.equal(14); }
             | «F»   { $.equal(15); }
};
```

Примечание — Выходные данные программ и функций, использующих идентификаторы UUID, могут содержать всевозможные варианты форм их представления — в верхнем регистре, в нижнем регистре, с дефисами и без дефисов между группами цифр. Для соответствия спецификации HL7 эти выходные данные должны быть приведены к стандартному виду, то есть между группами цифр 8-4-4-4-12 должны быть указаны дефисы и все шестнадцатеричные цифры должны быть в верхнем регистре.

B.1.1 Тип данных HL7 ReservedIdentifierScheme (RUID) (специализация типа данных UID)

Глобально уникальная строка, определенная исключительно в интересах комитета HL7. Идентификаторы в этой схеме утверждаются путем голосования спецификаций HL7. Местные сообщества или системы никогда не должны использовать такие зарезервированные идентификаторы на основе двусторонних соглашений.

Зарезервированные идентификаторы HL7 представляют собой строки, состоящие только из букв набора символов US-ASCII, цифр и дефисов, причем первый символ идентификатора должен быть буквой. Комитет HL7 может использовать эти зарезервированные идентификаторы как мнемонику основных понятий в сфере интересов комитета.

B.1.2 Тип данных InstanceIdentifier (II) (специализация типа данных ANY)

Идентификатор экземпляра, являющийся уникальным идентификатором предмета или объекта. Примерами служат идентификаторы объектов в модели HL7 RIM, номера медицинских карт, идентификаторы заказов, идентификаторы услуг в прейскуранте, идентификационные номера автомобилей VIN (Vehicle Identification Number) и т. д. Определения идентификаторов экземпляров основаны на объектных идентификаторах ИСО.

Таблица B.17 — Сводка свойств типа данных InstanceIdentifier

Имя	Тип	Описание
root	UID	Уникальный идентификатор, гарантирующий глобальную уникальность идентификатора экземпляра. Значение свойства root может само по себе быть идентификатором экземпляра
extension	ST	Строка символов, являющаяся уникальным идентификатором в пространстве имен, заданном значением свойства root
assigningAuthorityName	ST	Человеко-читаемое название или мнемоника уполномоченной организации по присвоению идентификаторов. Не имеет вычислительного значения. Это свойство предназначено для помощи человеку, интерпретирующему значение типа II. Примечание: никакая автоматизированная обработка не должна зависеть от названия уполномоченной организации, в какой бы форме оно не было представлено
displayable	BL	Указывает, предназначен ли идентификатор для вывода на экран и ввода данных (displayable = «true») или же используется только для машинной обработки (displayable = «false»)

```
type InstanceIdentifier alias II specializes ANY {
  ST  extension;
  UID root;
  ST  assigningAuthorityName;
  BL  equal(ANY x);
};
```

B.2.15.2 Свойство root : UID

Уникальный идентификатор, гарантирующий глобальную уникальность идентификатора экземпляра. Значение свойства root может само по себе быть идентификатором экземпляра.

При наличии непустого свойства extension значение свойства root обычно трактуется как «уполномоченная организации по присвоению идентификаторов», то есть предполагается, что это значение каким-то образом указывает организацию, присваивающие идентификаторы, передаваемые в свойстве extension. Однако значение свойства root не обязано представлять УИД организации, оно может также быть специально зарегистрированным уникальным идентификатором схемы идентификации[38)].

```
invariant(II x)
    where x.nonNull {
  root.nonNull;
};
```

B.2.15.3 Свойство extension : ST

Строка символов, являющаяся уникальным идентификатором в пространстве имен, заданном значением свойства root.

Значение свойства extension представляет собой строку символов, уникальную в пространстве имен, заданном значением свойства root. Если оно не является пустым, то свойство root задает пространство имен (иногда называемое «уполномоченной организацией по присвоению идентификаторов» или «типом идентификатора».) Свойство extension может быть пустым, и в таком случае значение свойства root само будет идентификатором экземпляра.

Схема идентификации, обеспечиваемая свойствами root и extension, фактически означает, что конкатенация значений свойств root и extension должна быть уникальным идентификатором объекта, идентифицируемым данным значением типа II.

[38)] Объекты DICOM идентифицируются только с помощью УИД. С точки зрения интеграции стандартов DICOM и HL7, было бы неразумно со стороны комитета HL7 требовать, чтобы свойство extension было обязательным и рассматривать УИД только как идентификатор организации, присваивающей идентификаторы. Поскольку идентификаторы УИД проще и не подвержены рискам незначащего декорирования, то комитет HL7 поддерживает практику использования в системах простых идентификаторов УИД в качестве ссылок на их объекты.

Для внешних идентификаторов передаваемых объектов рекомендуется использовать схему OID. Свойство extension в основном предназначено для передачи унаследованных алфавитно-цифровых идентификаторов.

В некоторых схемах идентификации используется определенный стиль представления соответствующих им значений. Например, номер карточки социального страхования в США SSN (Social Security Number) обычно записывается с дефисами по шаблону «123-12-1234». Однако дефисы не являются значащими символами и номер SSN может быть равным образом представлен без дефисов как «123121234».

В случае, если схемы идентификации предусматривают несколько разных представлений, комитетом HL7 должно быть принято решение о предпочтительной форме. Комитет должен документировать это решение, если им рекомендована соответствующая внешняя система идентификации. Решение о предпочтительной форме должно приниматься с учетом практичности и широты использования. Если четких критериев практичности и широты использования нет, то должно быть отдано предпочтение наиболее безопасной, расширяемой и наименее стилизованной (наименее декорированной) форме[39)].

Комитет HL7 может также принять решение об отображении распространенных внешних идентификаторов в значения свойства II.root типа OID. Например, указанный выше номер SSN может быть представлен как 2.16.840.1 .113883.4.1.123121234. В каждом конкретном случае решение комитета HL7 будет приниматься с учетом практичности и широты использования.

B.2.15.4 Свойство assigningAuthorityName: ST

Человеко-читаемое название или мнемоника уполномоченной организации по присвоению идентификаторов. Не имеет вычислительного значения. Это свойство предназначено для помощи человеку, интерпретирующему значение типа II. Примечание: никакая автоматизированная обработка не должна зависеть от названия уполномоченной организации, в какой бы форме оно не было представлено.

B.2.15.5 Свойство displayable : BL

Указывает, предназначен ли идентификатор для вывода на экран и ввода данных (displayable = «true») или же используется только для машинной обработки (displayable = «false»).

B.2.15.6 Свойство equality: BL (унаследовано от типа данных ANY)

Два идентификатора экземпляра равны в том и только том случае, если равны значения их свойств root и extension.

```
invariant(II x, y)
    where x.nonNull.and(y.nonNull) {
  x.equal(y).equal(x.root.equal(y.root)
        .and(x.extension.equal(y.extension)));
};
```

Рисунок B.7 — Типы данных URL и TEL

[39)] Это решение, принимаемое на этапе конструирования, необходимо, чтобы интерфейсы, предлагаемые в стандартах HL7, не были отягощены необходимостью реализации преобразований различных стилей литералов кодов во время исполнения, хотя это и может привести к ситуации, когда некоторым приложениям может потребоваться выполнять преобразования из одной формы представления кодов в другую, если они рассчитаны на вариант представления, не выбранный комитетом HL7.

На основании практического опыта рекомендуется, чтобы алфавитно-цифровые значения свойства extension типа данных II не содержали ведущих нулей (если в них присутствуют нули), поскольку эти нули нередко ошибочно вырезаются. Значения «000123» и «123» свойства extension должны рассматриваться как совершенно разные, но это правило может игнорироваться, что приводит к ошибочным совпадениям и дубликатам записей в базах данных. Но приложения должны учитывать все ведущие нули, обнаруженные в значениях свойства extension. Ведущие нули запрещены в идентификаторах ОИД, но в идентификаторах UUID они могут случаться, и это надо учитывать.

Отдельного свойства контрольной цифры нет. Контрольные цифры используются при ручном вводе данных и лучше всего работают, если они полностью прозрачны. Значения свойства extension типа данных II может содержать контрольные цифры в любом месте, а конкретная схема контрольного суммирования (если таковая есть) должна однозначно вытекать из значения свойства II.root. Но отдельное свойство контрольной цифры намеренно не включено в настоящую спецификацию.

B.1.3 Тип данных UniversalResourceLocator (URL) (специализация типа данных ANY)

Универсальный указатель ресурсов, представляющий телекоммуникационный адрес, соответствующий стандарту Интернет RFC 2396 [http://www.ietf.org/rfc/rfc2396.txt]. Этот стандарт описывает идентификатор URI, задающий протокол и точку контакта, определяемую этим протоколом для ресурса. Важными примерами использования типа данных URL являются номера телефонов и факсов, адреса электронной почты, гипертекстовые ссылки, ссылки FTP и т. д.

Стандарт Интернет RFC 2396 [http://www.ietf.org/rfc/rfc2396.txt] определяет адрес URL следующим образом:

Поскольку существует много различных методов доступа к ресурсам, существует и несколько схем описания местонахождения таких ресурсов. Общий синтаксис адресов URL служит базой для определения новых схем, основанных на использовании протоколов, отличающихся от тех, что описаны в этом документе.

Адреса URL используется для указания «местонахождения» ресурсов, предоставляя абстрактную идентификация местонахождения ресурса. Получив информацию о местонахождении ресурса, система может выполнить различные операции над этим ресурсом, которые могут быть охарактеризованы такими словами. как «доступ», «изменение», «замена», «поиск атрибутов». В общем случае для любой схемы URL достаточно указать только метод «доступ».

По местным соглашениям вместо адресов URL можно использовать идентификаторы URI. В этих случаях все же ожидается, что к идентифицированному ресурсу обеспечивается доступ некоторым согласованным методом. Идентификаторы URI широко используются для ссылки на вложения в конверты SOAP.

```
protected type UniversalResourceLocator alias URL specializes ANY {
    CS  scheme;
    ST  address;
  literal ST;
};
```

B.2.15.7 Свойство scheme: CS

Идентифицирует схему URL — протокол, используемый для интерпретации адресной строки и для доступа к ресурсу, адресованному таким способом.

Некоторые схемы URL регистрируются организацией Internet Assigned Numbers Authority (IANA) [http://www.iana.org], однако эта организация регистрирует только те схемы URL, которые описаны в документах Интернет RFC. В действительности есть целый ряд схем URL, определенных не в документах RFC. Часть из них зарегистрирована консорциумом World Wide Web Consortium (W3C)[40].

Аналогично типу среды ED.mediaType, в стандарте HL7 даются некоторые рекомендации для значений свойства scheme, классифицирую их как обязательные, рекомендованные, прочие и запрещенные. Любая не упомянутая в стандарте схема имеет статус «прочая».

Таблица B.18 — Домен значений свойства scheme

Код	Имя	Определение
fax	Fax (факс)	Телефонный номер факсимильного устройства [http://www.ietf.org/rfc/rfc3966.txt и http://www.ietf.org/rfc/rfc2806.txt]
file	File (файл)	Имена файлов, специфичные для места хранения [RCF 1738]. Обратите внимание, что схема file имеет смысл только для местных файлов. Передача таких имен другим системам мало востребована, поскольку система-получатель вряд ли будет способна получить непосредственный доступ к этому файлу
ftp	FTP	Протокол передачи файлов FTP (File Transfer Protocol) [http://www.ietf.org/rfc/rfc1738.txt]
http	HTTP	Протокол передачи гипертекста HTTP (Hypertext Transfer Protocol) [http://www.ietf.org/rfc/rfc2368.txt]
mailto	Mailto	Адрес электронной почты [http://www.ietf.org/rfc/rfc2368.txt].

[40] Свойство scheme имеет тип данных CS и для целей стандартов HL7 имеет нерасширяемый домен (с квалификатором CNE). С учетом того факта, что ни одного официального списка схем URL не существует и нередко используются местные схемы, это может показаться странным. Однако комитет HL7 не может позволить расширение схемы URL с использованием механизма местных альтернативных систем кодирования, поэтому технически свойству scheme присвоен тип данных CS.

Окончание таблицы В.18

Код	Имя	Определение
mllp	MLLP	Традиционный протокол MLLP (Minimal Lower Layer Protocol — протокол минимально возможного уровня), используемый для передачи сообщений по стандартам HL7. Адрес URL для такого протокола имеет ту же форму, что и IP-адрес, например, mllp://<host>:<port>/, где <host> является IP-адресом или именем узла DNS, а <port> является номером порта, обслуживающего протокол MLLP
modem	Modem (модем)	Телефонный номер модема [http://www.ietf.org/rfc/rfc3966.txt и http://www.ietf.org/rfc/rfc2806.txt]
nfs	NFS	Протокол сетевой файловой системы NFS (Network File System) [http://www.ietf.org/rfc/rfc2224.txt]. Серверы NFS используются в некоторых местах для общего доступа к файлам
tel	Telephone (телефон)	Телефонный номер для передачи голоса [http://www.ietf.org/rfc/rfc3966.txt и http://www.ietf.org/rfc/rfc2806.txt].
telnet	Telnet	Ссылка на диалоговые сеансы [http://www.ietf.org/rfc/rfc1738.txt]. В некоторых местах (например, в лабораториях) используются компьютерные терминалы для удаленных сеансов запросов по протоколу Telnet

Учтите, что настоящая спецификация явно ограничивает этот тип данных адресами URL. Универсальные имена ресурсов URN (Universal Resource Name) в ней не разрешены. Имена URN представляют собой вид схемы идентификации, используемой для ресурсов, к которым не обеспечивается доступ. А настоящая спецификация распространяется только на ресурсы, к которым может быть получен доступ и адресуемым с помощью URL.

B.2.15.8 Свойство address : ST

Адрес представляет собой строку символов, формат которой полностью определяется значением свойства scheme.

B.2.15.9 Литеральная форма

В то время как концептуально значения типа URL имеют свойства scheme и address, внешний вид адреса URL представляет собой строковый литерал, сформированный в соответствии со стандартом Internet. Общий синтаксис литерала URL определен следующим образом:

```
URL.literal ST {
  URL : /[a-z0-9+.-]+/ ":" ST   { $.scheme.equal($1);
                                  $.address.equal($3); }
};
```

Номера факсов и телефонов

Обратите внимания что для телефонных номеров нет специального типа данных. Они имеют тип данных TEL и специфицированы как адреса URL.

Адреса URL телефонных номеров определены в документе Интернет RFC 2806 [http://www.ietf.org/rfc/rfc2806.txt]. Это определение кратко излагается в настоящем подразделе. Он не предназначено для переопределения или изменения каких-либо правил, описанных в указанном выше документе.

Адреса URL голосовых телефонов начинаются с приставки tel:, а факсимильных телефонов — с приставки «fax:».

Свойство address содержит телефонный номер, соответствующий спецификации ITU-T E.123 Telephone Network and ISDN Operation, Numbering, Routing and Mobile Service: Notation for National and International Telephone Numbers (1993). Пока комитет HL7 не дополняет и не удаляет спецификацию адреса URL, предпочтительное подмножество синтаксиса телефонного номера, передаваемого в свойстве address, имеет следующее описание:

```
protected type TelephoneURL specializes URL {
  literal ST {
    URL                 : /(tel)|(fax)/ ":" address   {
$.scheme.equal($1);
$.address.equal($3); };
    ST address          : "+" phoneDigits
    ST phoneDigits      : digitOrSeparator phoneDigits
                        | digitOrSeparator
```

```
    ST digitOrSeparator  : digit
                         | separator;
    ST digit             : /[0..9]/;
    ST separator         : /[().-]/;
    };
};
```

Предпочтительными являются абсолютно глобальные телефонные номера, начинающиеся со знака «+» и кода страны. Разделяющие символы служат для удобства чтения, но не придают никакого особого смысла телефонному номеру. Например, строки «tel:+13176307960» и «tel:+1(317)630-7960» описывают один и тот же телефонный номер; строки «fax:+49308101724» и «fax:+49(30)8101-724» описывают один и тот же номер факса.

B.1.4 Тип данных TelecommunicationAddress (TEL) (специализация типа данных URL)

Телефонный номер (голосового телефона или факса), адрес электронной почты или иной указатель ресурса, воспринимаемый телекоммуникационным оборудованием. Этот телекоммуникационный адрес указывается как указатель URL, дополненный спецификациями времени и кодами использования, помогающими определить, какой адрес следует использовать в данное время и для данной цели.

Таблица B.19 — Сводка свойств типа данных TelecommunicationAddress

Имя	Тип	Описание
useablePeriod	GTS	Указывает периоды времени, в течение которых можно пользоваться данным телекоммуникационным адресом. Для телефонного номера это могут быть периоды времени, в течение которого нужному абоненту можно звонить по этому номеру. Для веб-адреса это могут быть периоды времени, в течение которого содержание сайта предполагается доступным по этому адресу
use	SET<CS>	Один или несколько кодов, информирующих систему или пользователя о том, какой из телекоммуникационных адресов следует выбрать для данной цели контакта

Семантика телекоммуникационного адреса состоит в том, что взаимодействующая по нему сущность (ответчик) ожидает запросы по этому адресу и готов на них ответить, и, следовательно, инициатор взаимодействия может установить с ним контакт.

Ответчиком по данному телекоммуникационному адресу может быть автоматическая служба, возвращающая необходимую информацию (например, службы FTP или HTTP). В этом случае телекоммуникационный адрес является ссылкой на информацию, доступную по этому адресу. Таким образом, по значению телекоммуникационного адреса можно получить некоторую информацию (в форме инкапсулированных данных типа ED).

```
type TelecommunicationAddress alias TEL specializes URL {
  GTS        useablePeriod;
  SET<CS>    use;
  BL         equal(ANY x);
};
```

Тип данных телекоммуникационного адреса представляет собой расширение типа данных UniversalResourceLocator (URL), определенного в соответствии со стандартом Интернет RFC 2396 [http://www.ietf.org/rfc/rfc2396.txt]. Тип данных URL указывает протокол и точку контакта, определенную этим протоколом для доступа к ресурсу. Наиболее примечательными приложениями типа данных телекоммуникационного адреса являются номера телефонов и факсов, адреса электронной почты, гипертекстовые ссылки, FTP-адреса и т. д.

B.2.15.10 Свойство useablePeriod: GTS

Указывает периоды времени, в течение которых можно пользоваться данным телекоммуникационным адресом. Для телефонного номера это могут быть периоды времени, в течение которого нужному абоненту можно звонить по этому номеру. Для веб-адреса это могут быть периоды времени, в течение которого содержание сайта предполагается доступным по этому адресу.

B.2.15.11 Свойство use: SET<CS>

Один или несколько кодов, информирующих систему или пользователя о том, какой из телекоммуникационных адресов следует выбрать для данной цели контакта.

Таблица В.20 — Домен значений свойства use

Код	Имя	Определение
H	home address (домашний адрес)	Домашний телекоммуникационный адрес; попытка контакта по этому адресу для деловых целей может явиться нарушением неприкосновенности личной жизни, и вместо нужного лица может ответить член семьи или иной житель. Обычно используется в экстренных случаях, или если других способов контакта нет
HP	primary home (основной домашний адрес)	Основной домашний телекоммуникационный адрес. Используется для контакта с лицом во внерабочее время
HV	vacation home (домашний на время отпуска)	Домашний телекоммуникационный адрес на время отпуска. Используется для контакта с лицом во время его отпуска
WP	work place (служебный)	Служебный адрес. Первый, по которому должны начинаться попытки контакта в рабочее время
DIR	direct (прямой)	Адрес рабочего места или телекоммуникационный адрес, по которому можно связаться с лицом без посредников. У телефона такой номер часто называется «прямым»
PUB	public (публичный)	«Стандартный» адрес рабочего места или телекоммуникационный адрес, по которому можно связаться со справочной, общим почтовым ящиком или иным посредником, обеспечивающим контакт с нужным лицом.
BAD	bad address (плохой адрес)	Признак «плохого», бесполезного адреса
TMP	temporary address (временный адрес)	Временный адрес, может быть приемлемым для визита или корреспонденции. История смены адресов может представить более детальную информацию
AS	answering service (автоответчик)	Автоответчик, используемый для менее срочных контактов, а также в ситуация, когда основной целью контакта является оставить сообщение или прослушать автоматическое объявление
EC	emergency contact (экстренный контакт)	Контакт, специально указанный для связи в экстренных случаях. Первый, по которому должны начинаться попытки контакта в экстренных случаях вне зависимости от наличия адресов с другими кодами использования
MC	mobile contact (мобильный контакт)	Мобильное телекоммуникационное устройство, которое владелец носит с собой. Может иметь характеристики других кодов использования, пригоден для срочных контактов, но не является первым, по которому должны начинаться попытки обычных деловых контактов
PG	pager (пейджер)	Пейджер, позволяющий инициировать обратный вызов или оставить очень короткое сообщение

Коды использования телекоммуникаций не являются полной классификацией типов оборудования или местонахождений. Их основной целью является предложить использование конкретного телекоммуникационного адреса или предостеречь от его использования. Не так-то просто предложить точные правила, управляющие выбором телекоммуникационного адреса.

В.2.15.12 Свойство equal: BL (унаследовано от типа данных ANY)

Два значения телекоммуникационного адреса считаются равными, если равны их URL. Свойства use и useablePeriod из проверки на равенство исключаются.

```
invariant(TEL x, y)
    where x.nonNull.and(y.nonNull) {
  x.equal(y).equal(((URL)x).equal((URL)y));
};
```

Типы данных почтового адреса и имен сущностей (фамилия, имя, отчество лица, название организации, тривиальное имя) основаны на расширении строкового типа данных ST (рисунок В.8).

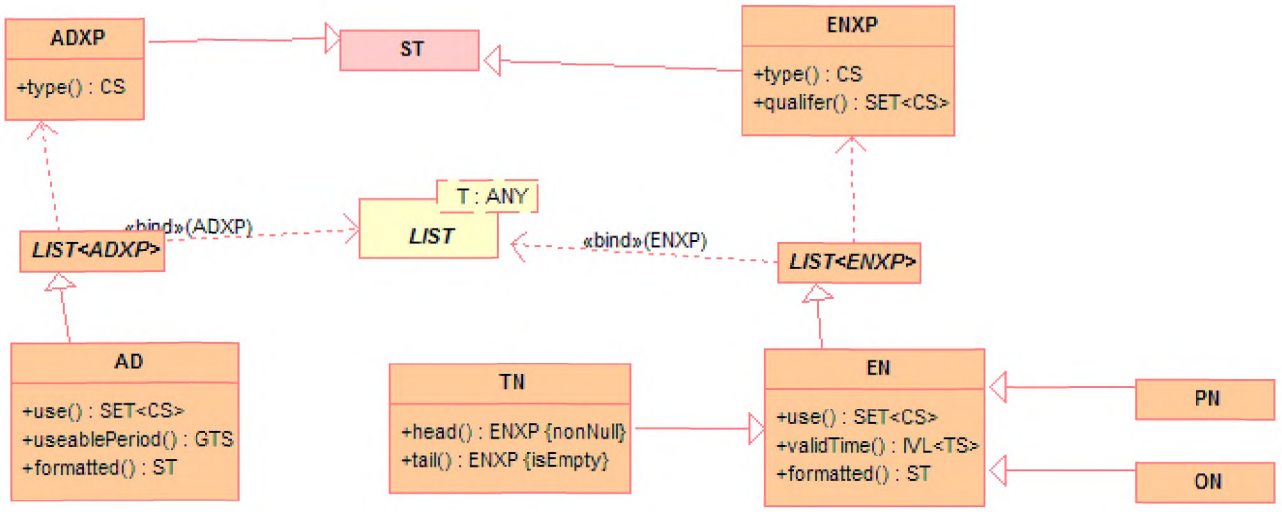


Рисунок В.8 — Типы данных почтового адреса и имен сущностей

В 1.5 Тип данных AddressPart (ADXP) (специализация типа данных ST)

Строка символов, которая может иметь маркировку типа, указывающую ее роль в адресе. Типичными компонентами, присутствующими почти в каждом адресе, являются название улицы, номер дома или почтового ящика, почтовый индекс, город, страна. Другие роли могут быть определены на региональном, национальном или даже на ведомственном уровне (например, адреса воинских частей). Обычно адреса разбиваются на строки, обозначаемыми специальными признаками разрыва строки (например, DEL).

Таблица В.21 — Сводка свойств типа данных AddressPart

Код	Имя	Определение
partType	CS	Указывает, означает ли компонент адреса улицу город, страну, почтовый индекс, почтовый ящик и т. д. Если это свойство имеет пустое значение, то компонент адреса является не классифицированным и должен представлять собой неструктурированный адрес

```
protected type AddressPart alias ADXP specializes ST {
  CS  type;
};
```

В.2.15.13 Свойство partType: CS

Указывает, означает ли компонент адреса улицу город, страну, почтовый индекс, почтовый ящик и т. д. Если это свойство имеет пустое значение, то компонент адреса является не классифицированным и должен представлять собой неструктурированный адрес.

Таблица В.22 — Домен значений свойства partType

Код	Имя	Определение
ADL	additional locator (дополнительный указатель)	Компонент с таким свойством может обозначать некоторую единицу адреса, например, номер квартиры, комнаты или этажа. В одном адресе может быть несколько таких компонентов (например, «3-й этаж, кв. 342»). Это может быть обозначением места, находящегося в стороне от адреса, а не меньшей единицей внутри некоторой большей (например, в Нидерландах обозначение «t.o.» означает «напротив» и используется для указания жилых лодок, стоящих напротив фасадов домов, выходящих на улицу)
UNID	unit identifier (идентификатор единицы)	Номер или название конкретной части здания или комплекса, присвоенные в этом здании или комплексе

Продолжение таблицы В.22

Код	Имя	Определение
UNIT	unit designator (обозначение единицы)	Указывает тип конкретной части здания или комплекса, например, квартира, этаж
DAL	delivery address line (строка адреса доставки)	Строка адреса доставки часто используется вместо выделения способа доставки, почтового накопителя и т. д. Обычно в адресе указывают либо строку с адресом улицы, либо строку с адресом доставки, но не обе вместе
DINST	delivery installation type (тип почтового накопителя)	Указывает тип почтового накопителя (места, в которое письмо должно быть доставлено перед окончательной передачей с помощью способа доставки.) Примеры почтовое отделение, почтовый узел, центральный телеграф, станция и т. д.
DINSTA	delivery installation area (территория почтового накопителя)	Местонахождение почтового накопителя, обычно поселок или город. Требуется только в тех случаях, когда оно отличается от муниципального образования. Территория, на которой осуществляется доставка почты этим накопителем, например, доставка почтальоном, доставка по сельскому или городскому маршруту
DINSTQ	delivery installation qualifier (квалификатор почтового накопителя)	Номер, буква или имя, выделяющие почтовый накопитель. Например, для «Станции А» квалификатором, выделяющим этот накопитель, будет буква «А»
DMOD	delivery mode (метод доставки)	Тип предлагаемой услуги, метод доставки. Например, почтовый ящик (в почтовом отделении), сельский маршрут, общая доставка и т. д.
DMODID	delivery mode identifier (идентификатор метода доставки)	Идентификатор метода доставки, например, номер маршрута доставки. Конкретизирует метод доставки (например, указывает номер почтового ящика или сельского маршрута)
SAL	street address line (строка адреса улицы)	
BNR	building number (номер здания)	Номер здания, дома или участка, находящегося на улице. Называется также «основным уличным номером». Это скорее номер здания, нежели улицы
BNN	building number numeric (числовой компонент номера здания)	Числовой компонент номера здания
BNS	building number suffix (суффикс номера здания)	Любая буква, дробь или иной текст, который может быть указан после числового компонента номера здания
STR	street name (название улицы)	Название улицы
STB	street name base (базовое название улицы)	Базовое название проезда по улице, присвоенное муниципалитетом (исключая тип и направление улицы)
STTYP	street type (тип улицы)	Тип улицы (например, улица, проспект, площадь и т. д.)
DIR	direction (направление)	Направление улицы (например, «Северная», «Южная», «Западная», «Восточная»)
CAR	care of (через)	Название получателя почтового отправления, который находится по этому адресу и обеспечит передачу отправления его адресату
CEN	census tract (переписной район)	Переписной район, выделяемый в демографических целях
CNT	country (страна)	Страна

Окончание таблицы В.22

Код	Имя	Определение
CPA	county or parish (графство или приход)	Административная единицы штата или провинции. (В 49 штатах США используется термин «county» (графство), в Луизиане –»parish» (приход))
CTY	municipality (муниципалитет)	Название города, поселка, деревни или иного муниципального образования
DEL	delimiter (разделитель)	Разделители печатаются без окаймления пробельными символами. Если этот компонент пуст, то разделителем служит переход на следующую строку
POB	post box (почтовый ящик)	Пронумерованный ящик в почтовом отделении
PRE	precinct (избирательный участок)	Структурная единица муниципального образования
STA	state or province (штат или провинция)	Административная единица страны, которая в федеральных государствах сохраняет ограниченный суверенитет
ZIP	postal code (почтовый индекс)	Почтовый индекс, обозначающий район, обслуживаемый почтовым отделением

B.2.16 Тип данных PostalAddress (AD) (специализация типа данных LIST<ADXP>)

Почтовый, домашний или служебный адрес. Представляет собой последовательность компонентов адреса, например, названия улицы или номера почтового ящика, города, почтового индекса, страны и т. д.

Тип данных AD прежде всего используется для передачи данных, позволяющих печатать этикетки с адресами, позволяющими лицу физически посетить адресата. Тип данных почтового адреса AD не предназначен служить контейнером для дополнительной информации, которая может быть полезной для поиска географического местонахождения (например, для GPS-координат) или для проведения эпидемиологических исследований. Такая дополнительная информация может передаваться в других, более подходящих элементах, определенных в стандартах HL7.

Таблица B.23 — Сводка свойств типа данных PostalAddress

Имя	Тип	Описание
use	SET<CS>	Один или несколько кодов, информирующих систему или пользователя о том, какой из адресов следует выбрать для данной цели
useablePeriod	GTS	Общая спецификация времени GTS (General Timing Specification), указывающая периоды времени, в течение которых можно использовать данный адрес. Используется для указания разных адресов в зависимости от дня недели или от года
isNotOrdered	BL	Булевское значение, указывающее, известен ли порядок компонентов адреса или нет. Хотя компоненты адреса всегда передаются в определенном порядке, тем не менее порядок, в котором они должны быть представлены пользователю, может быть, а может и не быть известен. Если он не известен, то это может быть указано с помощью свойства isNotOrdered
formatted	ST	Строковое значение адреса, форматированное по строкам и с нужными пробельными символами. Это только семантическое свойство, определяющее назначение некоторых типов компонентов адреса[41)]

Адреса концептуально рассматриваются как текст с добавленной логической разметкой. Такая разметка может разбивать адрес на строки и может детально описывать роль каждого компонента адреса, если она известна. Компоненты адреса присутствуют в нем в том порядке, как они печатаются на этикетке почтового отправления. Этот подход похож на разметку текста HTML или XML (но технически не сводится к представлению XML).

41) Учтите, что семантические свойства свободны от любой семантики потока управления. Свойство formatted может быть реализовано как «процедура», «возвращающая» форматированный адрес, но обычно оно не является переменной, которой можно присвоить форматированный адрес. Однако стандарт HL7 определяет не приложения, а только семантику передаваемых значений данных. Поэтому семантическая модель абстрагируется от понятий наподобие «процедуры», «возвращения» и «присваивания», а оперирует только свойством и значением.

По существу адреса являются последовательностями своих компонентов, но при этом имеют дополнительные свойства use и useablePeriod, указывающие, как и когда адрес может использоваться для конкретной цели.

```
type PostalAddress alias AD specializes LIST<ADXP> {
  SET<CS>  use;
  GTS      useablePeriod;
  BL       isNotOrdered;
  BL       equal(ANY x);
  ST       formatted;
};
```

B.2.16.1 Свойство use: SET<CS>

Один или несколько кодов, информирующих систему или пользователя о том, какой из адресов следует выбрать для данной цели.

Таблица B.24 — Домен значений свойства use типа данных PostalAddress

Код	Имя	Определение
H	home address (домашний адрес)	Домашний адрес; попытка контакта по этому адресу для деловых целей может явиться нарушением неприкосновенности личной жизни, и вместо нужного лица по этому адресу может находиться член семьи или иной житель. Обычно используется в экстренных случаях, или если других способов контакта нет
HP	primary home (основной домашний адрес)	Основной домашний адрес. Используется для контакта с лицом во внерабочее время
HV	vacation home (домашний на время отпуска)	Домашний адрес на время отпуска. Используется для контакта с лицом во время его отпуска
WP	work place (служебный)	Служебный адрес. Первый, по которому должны начинаться попытки контакта в рабочее время
DIR	direct (прямой)	Адрес рабочего места или телекоммуникационный адрес, по которому можно связаться с лицом без посредников. У телефона такой номер часто называется «прямым»
PUB	public (публичный)	«Стандартный» адрес рабочего места или телекоммуникационный адрес, по которому можно связаться со справочной, общим почтовым ящиком или иным посредником, обеспечивающим контакт с нужным лицом
BAD	bad address (плохой адрес)	Признак «плохого», бесполезного адреса
TMP	temporary address (временный адрес)	Временный адрес, может быть приемлемым для визита или корреспонденции. История смены адресов может представить более детальную информацию
Следующие коды указывают различные представления имен. Способ представления может влиять на использование имени (например, для формальных коммуникаций может требоваться идеографическое представление)		
ABC	Alphabetic (алфавитное)	Алфавитное представление имени (romaji в Японии)
IDE	Ideographic (идеографическое)	Идеографическое представление имени (kanji в Японии, китайские иероглифы)
SYL	Syllabic (силлабическое)	Силлабическая транскрипция имени (kana в Японии, hangul в Корее)
PHYS	physical visit address (адрес физического визита)	Используемый в основном для посещения этого адреса
PST	postal address (почтовый адрес)	Используется для отправки письма

Адрес, для которого свойство use не указано, может по умолчанию использоваться для любых целей, но адрес с конкретным значением этого свойства должен предпочитаться для соответствующей цели.

B.2.16.2 Свойство useablePeriod: GTS

Общая спецификация времени GTS (General Timing Specification), указывающая периоды времени, в течение которых можно использовать данный адрес. Используется для указания разных адресов в зависимости от дня недели или от года.

B.2.16.3 Свойство useablePeriod: BL

Булевское значение, указывающее, известен ли порядок компонентов адреса или нет. Хотя компоненты адреса всегда передаются в определенном порядке, тем не менее порядок, в котором они должны быть представлены пользователю, может быть, а может и не быть известен. Если он не известен, то это может быть указано с помощью свойства isNotOrdered.

B.2.16.4 Свойство equal: BL (унаследовано от типа данных ANY)

Два значения адреса считаются равными, если они содержат одинаковые компоненты адреса (без учета порядка следования). Свойства use и useablePeriod из проверки на равенство исключаются.

```
invariant(AD x, y)
    where x.nonNull.and(y.nonNull) {
  x.equal(y).equal((
    forall(ADXP p) where x.contains(p) {
        y.contains(p);
        }).and.(
       forall(ADXP p) where x.contains(p) {
            y.contains(p);
            }));
};
```

B.2.16.5 Свойство formatted: ST

Строковое значение адреса, форматированное по строкам и с нужными пробельными символами. Это только семантическое свойство, определяющее назначение некоторых типов компонентов адреса[42)].

Тип данных AD прежде всего используется для передачи данных, по которым можно печатать этикетки с адресами, позволяющими лицу физически посетить адресата. Люди воспринимают адреса в напечатанном виде, например, на этикетках. В определении типа данных AD точно указано, как форматируются адреса[43)].

Адреса являются упорядоченными списками компонентов адреса. Каждый компонент адреса печатается слева направо в том порядке, как он указан в списке (или в ином направлении, специфичном для языка; определение этого направления не входит в область применения настоящей спецификации). Печатается значение каждого компонента адреса. Большинство компонентов окаймляются пробельными символами. Следующие шесть правил регламентируют использование пробельных элементов:

1 Пробельные символы не аккумулируются, то есть два смежных пробельных символа эквивалентны одному. Смежные переходы на другую строку могут быть заменены одним переходом. Пробельные символы вокруг перехода на другую строку не являются значащими.

2 Литералы могут явно содержать пробельные элементы, к которым могут применяться те же самые правила сокращения пробелов. В литерале, указанном внутри текста одного компонента адреса, переходы на новую строку лишены смысла.

3 Явные ведущие и концевые пробельные символы не являются значащими во всех компонентах адреса, за исключением компонентов, у которых свойство partType имеет значение «DEL» (разделитель).

4 По умолчанию компонент адреса окаймляется неявными пробельными символами.

---

42) Учтите, что семантические свойства свободны от любой семантики потока управления. Свойство formatted может быть реализовано как «процедура», «возвращающая» форматированный адрес, но обычно оно не является переменной, которой можно присвоить форматированный адрес. Однако стандарт HL7 определяет не приложения, а только семантику передаваемых значений данных. Поэтому семантическая модель абстрагируется от понятий наподобие «процедуры», «возвращения» и «присваивания», а оперирует только свойством и значением.

43) Правила форматирования адреса являются частью семантики адресов, поскольку адреса в первую очередь являются изображаемым или печатаемым текстом, предназначенным для восприятия человеком. Другие использования (например, в эпидемиологии) являются вторичными. Хотя они и не запрещены, использование типа данных AD в этих случаях может оказаться не идеальным решением. В стандартах HL7 определены более лучшие способы удовлетворения таких сценариев. Обратите внимание, что правила форматирования адреса не относятся к вопросам технологий реализации, поскольку они применяются для представления адреса людям, а спецификации технологии реализации описывают адреса для целей обмена между компьютерами.

5 Компоненты адреса, у которых свойство partType имеет значение «DEL» (разделитель), неявными пробельными символами не окаймляются.

6 Явные ведущие и концевые пробельные символы являются значащими в компонентах адреса, у которых свойство partType имеет значение «DEL» (разделитель).

Это означает, что в общем случае все компоненты адреса окаймляются пробельными символами, но эти символы не аккумулируются. Компоненты разделителей никогда не окаймляются неявными пробельными символами, и каждый пробельный символ, которым заканчивается предшествующий компонент или начинается следующий компонент, удаляется вне зависимости от того, явный он или неявный.

Ниже показаны примеры представления адреса в форме, определяемой спецификацией реализуемой технологии на языке XML[44)].

Адрес

1050 W Wishard Blvd,

RG 5th floor,

Indianapolis, IN 46240.

может быть закодирован в любой из указанных ниже форм.

Первая форма используется системой, которая хранит адреса в форме свободного текста или в форме списка строк (строка 1, строка 2 и т. д.).

Пример 3 —

```
<addr use="WP">
  1050 W Wishard Blvd,
  RG 5th floor,
  Indianapolis, IN 46240
</addr>
```

Вторая форма содержит более специфичные компоненты адреса по сравнению с первой.

Пример 4 —

```
<addr use="WP">
  <streetAddressLine>1050 W Wishard Blvd</streetAddressLine>,
  <streetAddressLine>RG 5th floor</streetAddressLine>,
  <city>Indianapolis</city>,
  <state>IN</state>
  <postalCode>46240</postalCode>
</addr>
```

Эта форма типична для США, где адрес улицы иногда разделяется на компоненты, но город, штат и почтовый индекс всегда разделены.

Третья форма еще более специфична.

Пример 5 —

```
<addr use="WP">
  <houseNumber>1050</houseNumber>
  <direction>W</direction>
  <streetName>Wishard Blvd</streetName>,
  <additionalLocator>RG 5th floor</additionalLocator>,
  <city>Indianapolis</city>,
  <state>IN</state>
  <postalCode>46240</postalCode>
</addr>
```

---

44) Кодирование на языке XML в соответствии со спецификацией реализуемой технологии XML показано здесь лишь для того, чтобы избежать описания других нотаций. Это не означает, что функция работает только на языке XML или что XML является предпочтительным представлением.

Такая форма в США не используется. Однако она полезна в Германии, где многие системы выделяют номер дома в отдельный компонент. Например, германский адрес

Windsteiner Weg 54a,

D-14165 Berlin

скорее всего, будет представлен в описанной ниже форме[45)].

Пример 6 —

```
<addr use="HP">
  <streetName>Windsteiner Weg</streetName>
  <houseNumber>54a</houseNumber>,
  <country>D</country>-
  <postalCode>14165</postalCode>
  <city>Berlin</city>
</addr>
```

B.2.17 Тип данных EntityNamePart (ENXP) (специализация типа данных ST)

Строковое значение, представляющее компонент именования сущности (фамилии, имени, отчества лица). Может иметь свойство partType, обозначающее роль этого компонента в полном именовании, и квалификатор qualifier, детализирующий эту роль. Типичными компонентами именований лиц являются имена, фамилии, обращения и т. д.

Таблица B.25 — Сводка свойств типа данных EntityNamePart

Имя	Тип	Описание
partType	CS	Тип компонента указывает, является ли он именем, фамилией, префиксом, суффиксом и т. д.
qualifier	SET<CS>	Квалификатор содержит ряд кодов, каждый из которых указывает определенную подкатегорию компонента именования, дополняющую основной тип компонента. Например, можно указать, что имя является прозвищем, фамилия является псевдонимом или предназначена для использования в общедоступных записях

```
protected type EntityNamePart alias ENXP specializes ST {
  CS      type;
  SET<CS>  qualifier;
};
```

B 2.22.1 Свойство partType: CS

Тип компонента partType указывает, является ли он именем, фамилией, префиксом, суффиксом и т. д.

Таблица B.26 — Домен значений свойства partType типа данных EntityName

Код	Имя	Определение
FAM	family (фамилия)	Имя семьи, то есть имя, связывающее с генеалогией. В некоторых культурах (например, в Эритрее) фамилией сына является первое имя его отца
GIV	given (имя)	Имя (не следует называть его «первым именем», поскольку в написании именования оно не всегда идет первым)
PFX	prefix (префикс)	Префикс имеет жесткую связь с компонентом, непосредственно следующим за ним. У префикса нет неявных концевых пробельных символов (хотя он может иметь неявный ведущий пробельный символ). Учтите, что префиксы могут быть обращены в суффиксы

[45)] Этот пример демонстрирует эффективность применения разметки адреса. Типичная германская система, которая хранит номер дома и название улицы в разных полях, должна напечатать сначала название улицы, а затем номер дома. Это будет неправильным в США, поскольку там номер дома пишется перед названием улицы. Разметка адреса позволяет хранить компоненты адреса в естественном порядке и тем не менее понимать их роль.

Окончание таблицы В.26

Код	Имя	Определение
SFX	suffix (суффикс)	Суффикс имеет жесткую связь с компонентом, непосредственно предшествующим ему. У суффикса нет неявных ведущих пробельных символов (хотя он может иметь неявный концевой пробельный символ). Суффиксы не могут быть обращены в префиксы.
DEL	delimiter	Разделитель служит только в качестве литерала, печатаемого в данном представлении именования. Он не имеет неявных ведущих и концевых пробельных символов

Не каждый компонент именования должен иметь код типа. Если этот код не известен, не применим или просто не определен, то это выражается пустым значением (partType.isNull). Например, по именованию «Rogan Sulma» нельзя сделать уверенное заключение, что является именем, а что фамилией. Может даже оказаться, что «Rogan» является титулом.

Концептуально именования сущностей являются размеченным текстом. Разметка может детально описывать роль каждого компонента именования, если она известна. Компоненты именования присутствуют в нем в том порядке, как они печатаются на этикетке почтового отправления. Этот подход похож на разметку текста HTML или XML.

В.17.1 Свойство qualifier: SET<CS>

Квалификатор содержит ряд кодов, каждый из которых указывает определенную подкатегорию компонента именования, дополняющую основной тип компонента. Например, можно указать, что имя является прозвищем, фамилия является псевдонимом или предназначена для использования в общедоступных записях.

Таблица В.27 — Домен значений свойства qualifier типа данных EntityNamePart

Код	Имя	Определение
LS	legal status (юридическая форма собственности)	Для организаций этот суффикс указывает юридическую форму собственности, например, «Inc.», «Co.», «AG», «GmbH», «B.V.» «S.A.», «Ltd.» и т. д.
AC	academic (ученая степень или звание)	Указывает префикс наподобие «Др.», «M.D.» или «Ph.D.», обозначающий ученую степень или звание
NB	nobility (дворянин)	В Европе и Азии все еще есть люди с дворянскими титулами (аристократы). Германское «von» в большей мере является титулом, нежели приставкой. Другими примерами служат «Барон» или «Его Величество Король...» и т. д. В наши дни используется редко, но в некоторых системах все еще присутствует
PR	professional (профессиональная принадлежность)	Люди, воспитанные культурой Британской Империи, нередко используют в качестве суффиксов аббревиатуры своей профессиональной организации
VV	voorvoegsel (приставка)	Приставка (по-голландски «voorvoegsel») наподобие «van» или «de» в прошлом могла указывать дворянское происхождение, но в настоящее время это уже не так. Аналогичные приставки существуют и в других языках, например, в испанском, французском и португальском
AD	adopted (при усыновлении или удочерении)	Именование лица, данное ему при усыновлении (удочерении)
BR	birth (урожденный)	Именование лица, данное ему в краткий период после рождения. Обычно относится к фамилии, но может присваиваться и именам, если они были позже изменены
SP	Spouse (супружеское)	Именование, полученное от партнера в семейных отношениях (отсюда «М»). Обычно фамилия супруга. Учтите, что из существования супружеской фамилии нельзя сделать определенный вывод относительно пола супруга
CL	callme (зовите меня)	Именование (обычно имя), предпочтительное при прямом обращении к лицу

Окончание таблицы В.27

Код	Имя	Определение
IN	initial (инициал)	Указывает, что компонент именования является инициалом. В инициалы обычно не включается заключительная точка, поскольку это может быть не принято в языках, не использующих латинский алфавит. Инициалы могут содержать более одной буквы, например, инициал «Ph.» может соответствовать имени «Philippe», а «Th.» — имени «Thomas»
TITLE	title (титул)	Указывает, что префикс или суффикс является титулом, который применяется к полному именованию, а не только к смежному компоненту

В.2.18 Тип данных EntityName (EN) (специализация типа данных LIST<ENXP>)

Именования физического лица, организации, места или предмета. Последовательность компонентов, например, имени или фамилии, префикса, суффикса и т. д. Примерами именований служат «Джим Боб Уолтон мл.», «Health Level Seven, Inc.», «Озеро Тахо» и т. д. Именование может представлять собой простую строку или состоять из нескольких компонентов, например, «Джим», «Боб», «Уолтон» и «мл.», «Health Level Seven» и «Inc.», «Озеро» и «Тахо».

Таблица В.28 — Сводка свойств типа данных EntityName

Имя	Тип	Описание
use	SET<CS>	Один или несколько кодов, информирующих систему или пользователя о том, какое из именований следует выбрать для данной цели
validTime	IVL<TS>	Интервал времени, в течение которого данное именование используется или было использовано для этой сущности. Отражает тот факт, что именования людей, мест и предметов могут с течением времени изменяться
formatted	ST	Строковое значение адреса, форматированное по строкам и с нужными пробельными символами. Это только семантическое свойство, определяющее назначение некоторых типов компонентов адреса[46)]

Именования концептуально рассматриваются как текст с добавленной логической разметкой. Компоненты именования присутствуют в нем в том порядке, как они будут отображаться, а не в порядке, определяемом именем компонента. Упорядочение компонентов именования является важной характеристикой, которая заменяет необходимость в отдельном свойстве «именование для распечатки». Приложения могут изменить этот порядок в целях учета пожеланий своих пользователей. Этот подход похож на разметку текста HTML или XML (но технически не сводится к представлению XML).

По существу, именования являются последовательностями своих компонентов, но при этом имеют дополнительные свойства use и useablePeriod, указывающие, как и когда именование может использоваться для конкретной цели.

```
type EntityName alias EN specializes LIST<ENXP> {
  SET<CS>  use;
  IVL<TS>  validTime;
  BL       equal(ANY x);
  ST       formatted;
};
```

В.2.18.1 Свойство use: SET<CS>

Один или несколько кодов, информирующих систему или пользователя о том, какое из именований следует выбрать для данной цели.

[46)] Учтите, что семантические свойства свободны от любой семантики потока управления. Свойство formatted может быть реализовано как «процедура», «возвращающая» форматированный адрес, но обычно оно не является переменной, которой можно присвоить форматированный адрес. Однако стандарт HL7 определяет не приложения, а только семантику передаваемых значений данных. Поэтому семантическая модель абстрагируется от понятий наподобие «процедуры», «возвращения» и «присваивания», а оперирует только свойством и значением.

Таблица В.28 — Домен значений свойства use типа данных EntityName

Код	Имя	Определение
C	License (вписано в лицензию)	Именование, вписанное в лицензию, запись, сертификат и т. д. (только если отличается от юридического именования)
I	Indigenous/Tribal (племенное)	Например, «Вождь Красное Облако»
L	Legal (юридическое)	Известно как традиционное, которое можно использовать в юридических отношениях
P	pseudonym (псевдоним)	Самоприсвоенное именование, которое лицо использует или использовало
A	Artist/Stage (сценическое)	Псевдоним писателя, сценическое именование и т. д.)
R	Religious (религиозное)	Например, сестра Мэри Браун, брат Джон
SRCH	search (поисковое)	Именование, используемое для поиска или определения совпадения
PHON	phonetic (фонетическое)	Произношение наименования
SNDX	Soundex	Свертка именования в соответствии с алгоритмом SoundEx
ABC	Alphabetic (алфавитное)	Алфавитная транскрипция именования (romaji в Японии)
SYL	Syllabic (силлабическое)	Силлабическая транскрипция именования (например, kana в Японии, hangul в Корее)
IDE	Ideographic (идеографическое)	Идеографическое представление именования (kanji в Японии, китайские иероглифы)

Именование, для которого свойство use не указано, может по умолчанию использоваться для любых целей, но именование с конкретным значением этого свойства должно предпочитаться для соответствующей цели.

B.2.18.2 Свойство validTime: IVL<TS>

Интервал времени, в течение которого данное именование используется или было использовано для этой сущности. Отражает тот факт, что именования людей, мест и предметов могут с течением времени изменяться.

Тип данных EN соответствует расширению типа данных HistoryItem (HXIT).

B.2.18.3 Свойство equal: BL (унаследовано от типа данных ANY)

Два значения именования считаются равными, если они содержат одинаковые компоненты именования (без учета порядка следования). Свойства use и validTime из проверки на равенство исключаются.

```
invariant(EN x, y)
    where x.nonNull.and(y.nonNull) {
  x.equal(y).equal((
    forall(ENXP p) where x.contains(p) {
                y.contains(p);
                }).and.(
               forall(ENXP p) where x.contains(p) {
                    y.contains(p);
                    }));
```

B.2.18.4 Свойство formatted: ST

Строковое значение адреса, форматированное по строкам и с нужными пробельными символами. Это только семантическое свойство, определяющее назначение некоторых типов компонентов адреса [47)].

[47)] Учтите, что семантические свойства свободны от любой семантики потока управления. Свойство formatted может быть реализовано как «процедура», «возвращающая» форматированное именование, но обычно оно не является переменной, которой можно присвоить форматированное именование. Однако стандарт HL7 определяет не приложения, а только семантику передаваемых значений данных. Поэтому семантическая модель абстрагируется от понятий наподобие «процедуры», «возвращения» и «присваивания», а оперирует только свойством и значением.

Тип данных EN прежде всего используется для передачи именований физических лиц, мест и предметов (сущностей), позволяющих устно и письменно использовать их и ссылаться на них. Люди воспринимают именования в напечатанном виде, например, на этикетках. В определении типа данных EN точно указано, как форматируются именования[48)].

Именования являются упорядоченными списками компонентов именования. Каждый компонент именования печатается слева направо в том порядке, как он указан в списке (или в ином направлении, специфичном для языка). Печатается значение каждого компонента именования. Большинство компонентов окаймляются пробельными символами. Следующие шесть правил регламентируют использование пробельных элементов:

1 Пробельные символы не аккумулируются, то есть два смежных пробельных символа эквивалентны одному. Смежные переходы на другую строку могут быть заменены одним переходом. Пробельные символы вокруг перехода на другую строку не являются значащими.

2 Литералы могут явно содержать пробельные элементы, к которым могут применяться те же самые правила сокращения пробелов.

3 Явные ведущие и концевые пробельные символы не являются значащими во всех компонентах именования, за исключением компонентов, у которых свойство partType имеет значение «PFX» (префикс), «SFX» (суффикс), «DEL» (разделитель).

4 Компоненты именования, у которых свойство partType имеет значение «DEL» (разделитель), неявными пробельными символами не окаймляются. Ведущие и концевые явные пробельные символы являются в этих компонентах значащими.

5 Компонент именования, у которого свойство partType имеет значение «PFX» (префикс), может иметь только ведущий неявный пробельный символ, но не концевой. Явный концевой пробельный символ в этом компоненте является значащим.

6 Компонент именования, у которого свойство partType имеет значение «SFX» (суффикс), может иметь только концевой неявный пробельный символ, но не ведущий. Явный ведущий пробельный символ в этом компоненте является значащим.

Это означает, что в общем случае все компоненты именования окаймляются пробельными символами, но эти символы не аккумулируются. Компоненты разделителей никогда не окаймляются неявными пробельными символами, и каждый пробельный символ, которым заканчивается предшествующий компонент или начинается следующий компонент, удаляется вне зависимости от того, явный он или неявный.

B.2.18.5 Примеры

Ниже показан очень простой пример кодирования именования «Adam A. Everyman».

Пример 7 —

```
<name>
  <given>Adam</given>
  <given>A.</given>
  <family>Everyman</family>
</name>
```

Не должен упоминаться никакой специальный квалификатор, если его значение не известно или не применимо. Ниже показано интенсивное применение нескольких имен, префиксов, суффиксов, ученых степеней, дворянских титулов, приставок («van») и обозначений профессиональной принадлежности.

Пример 8 —

```
<name>
  <prefix qualifier="AC">Dr. phil. </prefix>
  <given>Regina</given>
  <given>Johanna</given>
  <given>Maria</given>
  <prefix qualifier="NB">Grafin </prefix>
  <family qualifier="BR">Hochheim</family>-<family qualifier="SP">Weilenfels</family>
  <suffix qualifier="PR">NCFSA</suffix>
</name>
```

---

48) Правила форматирования именования являются частью семантики именований, поскольку компоненты именований в первую очередь являются изображаемым или печатаемым текстом, предназначенным для восприятия человеком. Обратите внимание, что правила форматирования именований не относятся к вопросам технологий реализации, поскольку они применяются для представления именований людям, а спецификации технологии реализации описывают именования для целей обмена между компьютерами.

В следующем примере название организации «Health Level Seven, Inc.» показано в форме простой строки.

Пример 9 —

<name>Health Level Seven, Inc.</name>

В следующем примере это же название приведено в полностью структурированной форме.

Пример 10 —

<name>Health Level Seven, <suffix qualifier="LS">Inc.</suffix></name>

В следующем примере японские имя и фамилия показаны в трех формах: идеографической (Kanji), силлабической (Hiragana) и алфавитной (Romaji).

Пример 11 —

<name use="IDE">
<family>木村</family>
<given>通男</given>
</name>
<name use="SYL">
<family>きむら</family>
<given>みちお</given>
</name>
<name use="ABC">
<family>KIMURA</family>
<given>MICHIO</given>
</name>

B.2.19 Тип данных TrivialName (TN) (специализация типа данных EN)

Ограничение типа данных EN, представляющее собой простую строку именования предметов и мест.

Тип данных тривиального именования TN представляет собой вариант типа данных EN, состоящий только из одного компонента наименования, у которого нет ни типа компонента, ни какого-либо квалификатора. Поэтому тип данных TN и его единственный компонент именования эквивалентен простой строке символов. Эта эквивалентность выражается в форме понижающего приведения до типа данных ST и повышающего приведения до типа данных ST.

```
type TrivialName alias TN specializes EN {
  demotion   ST;
  promotion  TN (ST x);
};

invariant(TN x) where x.nonNull {
  x.head.nonNull;
  x.tail.isEmpty;
  x.formatted.equal(x.head);
};

invariant(ST x) {
  ((TN)x).head.equal(x);
};
```

Тривиальные именования обычно используются для мест и предметов, например, «Озеро Эри» или «Национальный аэропорт им. Рональда Рейгана».

Пример 12 —

<name>Озеро Эри</name>

<name>Национальный аэропорт им. Рональда Рейгана</name>

B.2.20 Тип данных PersonName (PN) (специализация типа данных EN)

Ограничение типа данных EN, используемое, когда именованной сущностью является физическое лицо. Представляет собой последовательность компонентов именования лица, например, имя или фамилия, префикс, суффикс и т. д. Компонент именования лица является ограничением компонента именования сущности, при котором используются только те квалификаторы компонента, которые применимы к именованиям лиц. Поскольку структура типа данных EN в основном определялась требованиями к именованиям лиц, это ограничение весьма минимально.

Так как большая часть функциональности типа данных EN относилась к именованиям лиц, то в целом тип данных PN представляет собой минимальное ограничение квалификаторов компонентов типа данных EN.

```
type PersonName alias PN specializes EN;

invariant(PN this) {
  forall(ENXP part)
      where this.contains(part) {
    part.qualifier.contains("LS").not;
    }
};
```

B.2.21 Тип данных OrganizationName (ON) (специализация типа данных EN)

Ограничение типа данных EN, используемое, когда именованной сущностью является организация. Представляет собой последовательность компонентов именования.

Название организации, например, «Health Level Seven, Inc.». Название организации состоит только из не типизированных частей наименования, префиксов, суффиксов и разделителей.

```
type OrganizationName alias ON specializes EN;

invariant(ON this) {
  forall(ENXP part)
      where this.contains(part) {
    part.type.implies("FAM").not;
    part.type.implies("GIV").not;
    }
};
```

B.2.21.1 Примеры

Ниже показан пример названия организации «Health Level Seven, Inc.» в форме простейшей строки.

Пример 13 —

<name>Health Level Seven, Inc.</name>

В следующем примере юридическая форма собственности вынесена в отдельный компонент именования.

Пример 14 —

<name>Health Level Seven, <suffix qualifier="LS">Inc.</suffix></name>

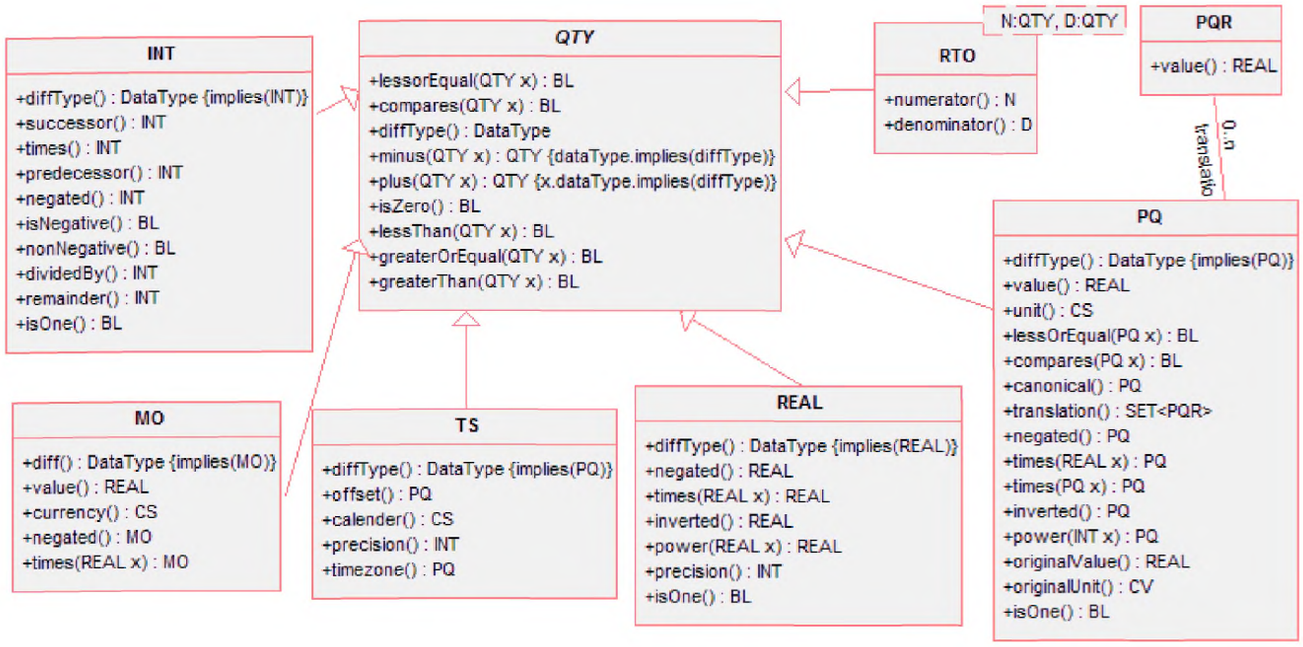


Рисунок В.9 — Типы данных количества

В.2.22 Абстрактный тип данных Quantity (QTY) (специализация типа данных ANY)

Тип данных Quantity (количество) представляет собой абстрактное обобщение всех типов данных, у которых (1) набор значений упорядочен (имеет отношение lessOrEqual) и (2) для всех полностью упорядоченных подмножеств значений типа данных определена операция вычитания. Абстрактный тип количества необходим для определения некоторых других типов, например, интервала и распределения вероятности.

```
abstract type Quantity alias QTY specializes ANY {
  BL    lessOrEqual(QTY x);
  BL    compares(QTY x);
  TYPE  diffType;
  QTY   minus(QTY x);
  QTY   plus(QTY x);
  BL    isZero;
  BL    lessThan(QTY x);
  BL    greaterOrEqual(QTY x);
  BL    greaterThan(QTY x);
};
```

В.22.2.1 Свойство упорядочения lessOrEqual: BL

Предикат, выражающий рефлексивное, симметричное и транзитивное отношение порядка между двумя значениями количества.

Отношение lessOrEqual определено на полностью упорядоченном подмножестве типа данных Quantity, то есть подмножестве, для всех элементов которого задан определенный порядок (например, целые и вещественные числа полностью упорядочены).

Напротив, в частично упорядоченном множестве некоторые, но не все пары элементов сопоставимы с помощью отношения порядка (например, структура дерева или множество физических величин представляют собой частично упорядоченные множества). Два значения данных x и y упорядоченного типа данных сопоставимы (x.compares(y)), если между ними существует отношение lessOrEqual (меньше или равно) в ту или иную сторону ($x \le y$ или $y \le x$).

Отношение частичного порядка генерирует полностью упорядоченные подмножества, объединение которых составляет полное множество (например, множество всех величин длины является полностью упорядоченным подмножеством во множестве всех физических величин).

Например, структура дерева частично упорядочена, если корень считать меньше или равным листу, но при этом отношения порядка между листьями может не быть. Физические величины также частично упорядочены, поскольку отношение порядка определено только для величин одной размерности (например, между двумя длинами, но не между длиной и временем). Полностью упорядоченное подмножество дерева представляет собой путь, который транзитивно соединяет лист с корнем. Физическая размерность времени представляет собой полностью упорядоченное подмножество физических величин.

```
      invariant (QTY x, y, z)
  where x.nonNull.and(y.nonNull).and(z.nonNull) {
    x.lessOrEqual(x);                              /* рефлексивное */
    x.lessOrEqual(y).implies(y.lessOrEqual(x)).not; /* асимметричное */
    x.lessOrEqual(y).and(y.lessOrEqual(z))
     .implies(x.lessOrEqual(z))                    /* транзитивное */
};
```

B.2.22.2 Свойство equal: BL (унаследовано от типа данных ANY)

Равенство является рефлексивным, симметричным и транзитивным отношением между двумя значениями данных. Равенство возможно только между подходящими значениями, пустые значения никогда не равны (даже если имеют одинаковую причину пустоты).

```
invariant(Quantity x, y, z)
    where x.nonNull.and(y.nonNull).and(z.nonNull) {
  x.equal(x);                                      /* reflexivity */
  x.equal(y).equal(y.equal(x));                    /* symmetry */
  x.equal(y).and(y.equal(z)).implies(x.equal(z))   /* transitivity */
  x.equal(y).implies(x.dataType.equal(y.dataType);
};
```

Способ установления равенства должен быть определен для каждого типа данных. Если иное не указано, то два значения данных равны, если они не различимы, то есть если у них нет различающихся семантических свойств. Это общее определение равенства может быть переопределено в типе данных с помощью указания собственного отношения равенства. Такое переопределение отношения равенства может быть использовано для исключения семантических свойств из проверки на равенство. Если в типе данных какие-то семантические свойства исключены из его определения равенства, это означает, что определенные свойства (или аспекты свойств), не ставшие частью проверки на равенство, не существенны для смысла значения.

Например, физическая величина имеет два семантических свойства: (1) вещественное число, (2) кодированная единица измерения. Однако при проверке на равенство необходимо учитывать тот факт, что, к примеру, 1 метр равен 100 сантиметрам. Таким образом, независимые равенства двух семантических свойств являются слишком строгим критерием равенства двух величин. Поэтому в определении типа данных физической величины необходимо переопределить отношение равенства.

B.2.22.3 Свойство compares: BL

Предикат, указывающий, что данное значение и операнд являются сравнимыми, и можно определить, какое из них больше другого.

Две величины являются сравнимыми, если они являются элементами общего полностью упорядоченного подмножества пространства значений их типов данных. Определение сравнения основано на свойстве lessOrEqual.

```
invariant (QTY x, y, z)
    where x.nonNull.and(y.nonNull) {
  x.compares(y).equal(x.lessOrEqual(y).or(y.lessOrEqual(x)));
};
```

B.2.22.4 Свойство diffType: TYPE

Свойство diffType определяет тип разности между двумя значениями конкретного типа данных QTY.

```
invariant(QTY x) {
  x.diffType.implies(QTY)
};
```

Тип разности представляет собой некоторый тип данных, который далее специализирует тип данных QTY.

B.2.22.5 Свойство minus: QTY

Величина, выражающая «дистанцию» данной величины от величины операнда, которая должна быть сравнимой. Тип данных величины разности связан с типом данных операнда, но не обязан быть тем же самым.

```
invariant(QTY x, y) {
  x.minus(y).implies(x.diffType);
};
```

Результат вычитания minus имеет тип данных, возвращаемых в свойстве diffType данной величины.

Разность определена в упорядоченном множестве, если семантически значимо утверждать, что Δ представляет собой разность значений *x* и *y*. Разность Δ должна быть значимой независимо от значений величин *x* и *y*. Такая независимость существует, если для всех значений *u* можно осмысленно вывести такое значение *v*, что Δ будет также разностью между *u* и *v*. Суждение о том, что значимо, не может быть определено формально[49)].

Свойство minus имеет тип данных, способный представить разность двух значений, между которыми существует отношение упорядоченности (то есть эти значения являются элементами общего полностью упорядоченного подмножества). Например, разность двух целых чисел имеет тип данных целого числа, а разность двух моментов времени является физической величиной, имеющей размерность времени. Тип данных разности имеет полностью упорядоченный домен значений.

Разность двух значений *x* и *y* должна быть определена для всех пар *x* и *y* в полностью упорядоченном подмножестве значений типа данных. Нуль представляет собой разность значения с самим собой.

```
invariant(QTY x, y)
    where x.compares(y) {
  x.minus(y).nonNull;
  x.minus(x).isZero;
};
```

Если значения x и y не сравнимы, то их разность должна иметь пустое значение.

```
invariant(QTY x, y)
    where x.compares(y).Not {
  x.minus(y).notApplicable;
};
```

B.2.22.6 Свойство plus: QTY

Сумма данного количества и значения операнда. Операнд должен иметь тип данных, который может выразить разность между двумя значениями этого типа данных физической величины.

```
invariant(QTY x, y)
    where x.compares(y) {
  x.plus(y.minus(x)).equal(y);
  y.dataType.implies(x.diffType);
};
```

Вопрос: как соотносятся друг с другом выражения {y.dataType.implies(x.diffType)} и {x.compares(y)}?

Если тип значения *y* не является допустимым для разности значений, имеющих тип значения *x*, то результатом операции должно быть пустое значение.

```
invariant(QTY x, y)
    where y.dataType.implies(x.diffType).not {
  x.plus(y).notApplicable;
};
```

[49)] Абстракция типа данных количества отвечает понятию шкал разности в противовес порядковым шкалам и шкалам отношений (Guttman и Stevens). Тип данных, в котором определены только требования к упорядочению, но не к разности значений, является порядковым. В настоящее время для порядковых типов данных специальный тип данных не определен. Обычно порядковыми величинами являются кодированные значения, взятые из систем кодирования, в которых определено отношения порядка. Но пока что семантика упорядочения не отражено в семантике типов данных, определенных в стандартах HL7.

B.2.22.7 Свойство isZero: BL (является нулем)

Нейтральный элемент операций разности и сложения, то есть, если величина равно нулю, то сложение с ней или ее вычитание из другой сравнимой величины должно быть равно этой другой величине.

```
invariant(QTY x) {
  x.minus(x).isZero;
};
```

B.2.22.8 Свойство отношения порядка lessThan: BL (меньше)

Предикат, указывающий асимметричное и транзитивное отношение упорядочения между данной величиной и другой величиной. Это отношение упорядочения то же самое, что и для свойства lessOrEqual, но не является рефлексивным.

```
invariant (QTY x, y, z)
    where x.nonNull.and(y.nonNull) {
  x.lessThan(y).equal(x.lessOrEqual(y)
          .and(x.equal(y).not));
};
```

B.2.22.9 Свойство отношения порядка greaterOrEqual: BL (больше или равно)

Предикат, указывающий асимметричное и транзитивное отношение упорядочения между данной величиной и другой величиной. Оно является обращением отношения lessOrEqual.

```
invariant (QTY x, y, z)
    where x.nonNull.and(y.nonNull) {
  x.greaterOrEqual(y).equal(y.lessOrEqual(x));
};
```

B.2.22.10 Свойство отношения порядка greaterThan : BL (больше)

Предикат, указывающий асимметричное и транзитивное отношение упорядочения между данной величиной и другой величиной. Оно является обращением отношения lessThan.

```
invariant (QTY x, y, z)
    where x.nonNull.and(y.nonNull) {
  x.greaterThan(y).equal(y.lessThan(x));
};
```

B.2.23 Целочисленный тип данных IntegerNumber (INT) (специализация типа данных QTY)

Целые числа (-1,0,1,2, 100, 3398129 и т. д.), являющиеся точными числами, полученными в результате подсчета и перечисления. Целые числа являются дискретными, множество целых чисел бесконечно, но счетно. На диапазон целых чисел не накладывается никакого произвольного ограничения. Для положительной и отрицательной бесконечности предусмотрены отдельные причины пустоты.

```
type IntegerNumber alias INT specializes QTY {
    INT    successor;
    INT    times(INT x);
    INT    predecessor;
    INT    negated;
    BL     isNegative;
    BL     nonNegative;
    INT    dividedBy(INT x);
    INT    remainder(INT x);
    BL     isOne;
  literal ST;
};
```

Так как тип данных IntegerNumber имеет всю семантик математического понятия целого числа, то определены базовые операции plus (сложение) и times (умножение). Эти операции определены здесь как характеризующие операции в смысле стандарта ИСО 11404 по той причине, что они необходимы в других частях настоящей спецификации, а именно для определения семантики литеральной формы.

Традиционное рекурсивное определение сложения и умножения восходит к Грассману и использует понятие «следующее» (successor)[50].

```
invariant(INT x, o, i)
    where x.nonNull.and(o.isZero) {
  x.lessThan(x.successor);
  x.plus(o).equal(x);
  x.plus(y.successor).equal(x.plus(y).successor);
  x.times(o).equal(o);
  x.times(y.successor).equal(x.times(y)).plus(x);
};
```

B.2.23.1 Свойство successor: INT (следующее)

Следующим называется такое значение типа INT, большее данного значения типа INT, для которого не существует никакого значения типа INT, находящегося между данным и следующим значением.

```
invariant(INT x, y)
    where x.successor(y) {
  x.lessThan(y).and.not(exists(INT z) {
    x.lessThan(z);
    z.lessThan(y);
    });
};
```

B.2.23.2 Свойство типа данных разности diffType: TYPE (унаследовано от типа данных QTY)

```
invariant(INT x) {
  x.diffType.implies(INT);
};
```

Тип данных разности (diffType) двух значений типа INT также является типом данных INT.

B.2.23.3 Свойство сложения plus: INT (унаследовано от типа данных QTY)

```
invariant(INT x, y, o)
    where x.nonNull.and(y.nonNull).and(o.isZero) {
  x.plus(o).equal(x);
  x.plus(y.successor).equal(x.plus(y).successor);
};
```

B.2.23.4 Свойство умножения times: INT

Результат умножения данного целого значения на операнд, эквивалентный повторениям сложения этого значения.

```
invariant(INT x, y, i, o)
    where x.compares(y).and(o.isZero).and(i.isOne) {
  x.times(o).equal(o);
```

[50] H. Grassman. *Lehrbuch der Arithmetik.* 1861. Исходным аксиомам, предложенным Грассманом, отдается предпочтение перед аксиомами Пеано, поскольку аксиомы Грассмана охватывают все целые числа, а не только натуральные. Также «довольно хорошо известно, что по собственному признанию Пеано, он позаимствовал свои аксиомы у Дедекинда и при их разработке существенно использовал труды Грассмана». (Hao Wang. The Axiomatization of Arithmetic. J. Symb. Logic; 1957:22(2); p. 145.)

```
  x.times(i).equal(x);
  x.times(y.successor).equal(x.times(y)).plus(x);
};
```

B.2.23.5 Свойство predecessor: INT (предыдущее)

Свойство, обратное свойству successor.

```
invariant(INT x, y)
    where x.successor(y) {
  x.successor.predecessor.equal(x);
};
```

B.2.23.6 Свойство negated: INT (отрицание)

Элемент, обратный значению типа INT, то есть такое другое значение типа INT, которое при сложении с данным значением дает нулевой результат (нейтральный элемент).

```
invariant(INT x)
    where x.nonNull {
  x.plus(x.negated).isZero;
};
```

B.2.23.7 Свойство nonNegative: BL (неотрицательное)

Предикат, указывающий, является ли нулевое значение типа INT (нейтральный элемент, меньшим данного значения типа INT ил равным ему.

```
invariant(INT x, o)
    where x.nonNull.and(o.isZero) {
  x.nonNegative.equal(o.lessOrEqual(x));
};
```

B.2.23.8 Свойство isNegative: BL (отрицательное)

Предикат, указывающий, что данное значение типа *INT* меньше нуля (не является неотрицательным).

```
invariant(INT x)
    where x.nonNull {
  x.isNegative.equal(x.nonNegative.not);
};
```

B.2.23.9 Свойство dividedBy: INT (деление)

Целочисленным делением данного целого значения (делимого — dividend) на другое целое значение (делитель — divisor) является целое значение, произведение которого на делитель равно делимому.

```
invariant(INT dividend, divisor, o, i)
    where divisor.isZero.not.and(o.isZero) {
  dividend.isZero.implies(dividend.dividedBy(divisor).equal(o));
  dividend.isZero.not.implies(dividend.dividedBy(divisor).equal(
    absolute(dividend).minus(absolute(divisor)).dividedBy(absolute(divisor))
                      .successor.times(sign(dividend)
                      .times(sign(divisor)))));
};
```

B.2.23.10 Свойство reminder: INT (остаток)

Остаток от целочисленного деления.

```
invariant(INT x, y)
    where x.nonNull.and(y.nonNull) {
  x.reminder(y).equal(x.minus(x.dividedBy(z).times(y)));
};
```

Данное определение остатка совпадает с определениями, приведенными в языках программирования C и Java.

B.2.23.11 Свойство isOne: BL (нейтральный элемент умножения)

Предикат, указывающий, что данное значение является единицей, то есть нейтральным элементом умножения. Этим свойством обладает ровно одно целое значение.

```
invariant(INT x, y)
    where x.nonNull.and(y.nonNull) {
  x.isOne.and(y.isOne).implies(x.equal(y));
  x.isOne.and(y.isZero).implies(x.equal(y).not);
};
```

B.2.23.12 Литеральная форма

Литеральной формой целого значения является простое десятичное число, то есть строка десятичных цифр.

```
INT.literal ST {
  INT digit : "0"        { $.isZero; }
            | "1"        { $.equal(0.successor); }
            | "2"        { $.equal(1.successor); }
            | "3"        { $.equal(2.successor); }
            | "4"        { $.equal(3.successor); }
            | "5"        { $.equal(4.successor); }
            | "6"        { $.equal(5.successor); }
            | "7"        { $.equal(6.successor); }
            | "8"        { $.equal(7.successor); }
            | "9"        { $.equal(8.successor); };

  INT uint  : digit      { $.equal($1); }
            | uint digit { $.equal($1.times(9.successor).plus($2)); };

  INT       : uint       { $.equal($1); }
            | "+" uint   { $.equal($2); }
            | "-" uint   { $.equal($2.negated); };
};
```

B.2.24 Вещественный тип данных RealNumber (REAL) (специализация тпа данных QTY)

Дробные числа. Обычно получаются при измерении и оценке физических величин, а также в результате вычислений, проводимых над другими вещественными числами. Обычно представляются в десятичной форме, где число значащих десятичных разрядов именуется как точность.

В настоящей спецификации термин «вещественное число» используется для обозначения дробных значений, не подразумевая под ними полное множество математических вещественных чисел, включающее в себя иррациональные числа, например, ρ, число Эйлера и т. д.[51)].

Примечание — В настоящей спецификации вещественный тип данных трактуется в максимально широком смысле. Однако из этого не следует, что любая спецификация реализуемой технологии или реализация, соответствующая стандарту, должна представлять полный диапазон вещественных чисел, что невозможно для любой конечной реализации. В настоящее время сценарии использования вещественных чисел в стандартах HL7 ограничиваются измеряемыми и оцениваемыми физическими величинами и денежными суммами. В этих сценариях может использоваться ограниченное пространство вещественных значений, и даже только очень ограниченное множество десятичных чисел (масштабируемых целых значений). Однако при этом объявляются такие представления пространства вещественных значений, как числа с плавающей точкой, рациональные числа, масштабируемые целые числа, строки десятичных цифр, и различные ограничения этих представлений не входят в область применения настоящей спецификации.

[51)] Термин «вещественный» применительно к типу данных дробных чисел восходит к языкам программирования Algol и Pascal, где он прочно укоренился.

В настоящей спецификации предлагаются две альтернативы числового типа данных. Выбор между ними осуществляется следующим образом: числовой атрибут считается вещественным, если точно не известно, что он является целым. Число является целым, если оно всегда является результатом подсчета, обычно представляя порядковый номер. В тех возможных сценариях, когда число является результатом оценки или усреднения, она не всегда имеет целое значение и, следовательно, необходимо использовать тип данных REAL.

```
type RealNumber alias REAL specializes QTY {
                REAL  negated;
                REAL  times(REAL x);
                REAL  inverted;
                BL    isOne;
                REAL  power(REAL x);
  literal  ST;
                INT   precision;
  demotion    INT;
  promotion   REAL (INT x);
  promotion   PQ;
  promotion   RTO;
};
```

Указанные здесь алгебраические операции определены как характеризующие операции в смысле стандарта ИСО 11404 по той причине, что они необходимы в других частях настоящей спецификации.

В отличие от целых значений семантика вещественных значений не конструируется методом индукции, а только интуитивно описывается на основе соответствующих аксиом об их алгебраических свойствах. Полнота аксиоматики намеренно оставлена в стороне, чтобы не делать никаких утверждений об иррациональных числах.

B.2.24.1 Свойство compares: BL (унаследовано от типа данных QTY)

Множество значений типа REAL является полностью упорядоченным.

```
invariant(REAL x, y)
    where x.nonNull.and(y.nonNull) {
  x.compares(y);
};
```

B.2.24.2 Свойство типа данных разности diffType TYPE (унаследовано от типа данных QTY)

```
invariant(REAL x) {
  x.diffType.implies(REAL);
};
```

Тип данных разности (diffType) двух значений типа REAL также является типом данных REAL.

B.2.24.3 Свойство сложения plus: REAL (унаследовано от типа данных QTY)

```
invariant(REAL x, y, z, o)
    where x.nonNull.and(y.nonNull).and(z.nonNull).and(o.isZero) {
  x.plus(o).equal(x);                          /* neutral element */
  x.plus(y).plus(z).equal(x.plus(y.plus(z)));  /* associative */
  x.plus(y).equal(y.plus(x));                  /* commutative */
  z.lessOrEqual(x).and(z.lessOrEqual(y)
                    .implies(z.lessOrEqual(x.plus(y));
  x.lessOrEqual(y).implies(x.plus(z)
                   .lessOrEqual(y.plus(z)));
};
```

B.2.24.4 Свойство negated (обратное значение к сложению): REAL

Элемент, обратный значению типа REAL, то есть такое другое значение типа REAL, которое при сложении с данным значением дает нулевой результат (нейтральный элемент сложения).

```
invariant(REAL x)
    where x.nonNull {
  x.plus(x.negated).isZero;
};
```

B.2.24.5 Свойство isOne: BL (нейтральный элемент умножения)

Предикат, указывающий, что данное значение является единицей, то есть нейтральным элементом умножения. Этим свойством обладает ровно одно целое значение.

```
invariant(REAL x, y)
    where x.nonNull.and(y.nonNull) {
  x.isOne.and(y.isOne).implies(x.equal(y));
  x.isOne.and(y.isZero).implies(x.equal(y).not);
};
```

B.2.24.6 Свойство умножения times: REAL

Операция над множеством значений REAL, формирующая абелеву группу и связанная со сложением правилом дистрибутивности.

```
invariant(REAL x, y, z, i, o)
    where x.nonNull.and(y.nonNull).and(z.nonNull)
        .and(i.isOne).and(o.isZero) {
  x.times(o).equal(o);
  x.times(i).equal(x);                                  /* neutral element */
  x.times(y).times(z).equal(x.times(y.times(z)));       /* associative */
  x.times(y).equal(y.times(x));                         /* commutative */
  x.times(y.plus(z)).equal(x.times(y).plus(x.times(z)); /* distributive */
  o.lessOrEqual(x).and(o.lessOrEqual(y).implies(o.lessOrEqual(x.times(y));
};
```

B.2.24.7 Свойство inverted (обратное действие для умножения): REAL

Значение типа REAL, которое при умножении на другое значение типа REAL, дает единицу (нейтральный элемент умножения). Нуль (нейтральный элемент сложения) не имеет обратного значения.

```
invariant(REAL x, i)
    where x.isZero.not.and(i.isOne) {
  x.times(x.inverted).equal(i);
};
```

B.2.24.8 Гомоморфизм типа данных INT в тип данных REAL: INT

Типы данных INT и REAL связаны гомоморфизмом, преобразующим каждое значение типа INT в значение типа REAL с сохранением алгебраических операций типа данных INT. Это означает, что целое значение может быть приведено к вещественному, а вещественное может быть понижено до целого числа с помощью округления дробной части.

```
invariant(INT n, m)
    where n.nonNull.and(m.nonNull) {
  ((REAL)n.plus(m)).equal(((REAL)n).plus((REAL)m));
  ((REAL)n.times(m)).equal(((REAL)n).times((REAL)m));
};
```

B.2.24.9 Свойство возведения в степень power: REAL

Основой возведения в степень является повторение умножения вещественного значения, расширенное до рациональной степени с помощью обратной операции извлечения корня.

Ниже перечислены только некоторые общие свойства возведения в степень.

```
invariant(REAL x, y, z, o, i)
    where x.nonNull.and(y.nonNull).and(z.nonNull)
        .and(o.isZero).and.(i.isOne) {
  forall(INT n)
      where n.nonNull {
    n.greaterThan(o).implies(
      x.power(n).equal(x.times(x.power(n.predecessor))));
    n.lessThan(o).implies(
      x.power(n).equal(x.power(n.negated).inverted);
    }
    x.power(o).equal(i);
    x.power(i).equal(x);
    x.power(y).power(z).equal(x.power(y.times(z)));
    x.power(y).times(x.power(z)).equal(x.power(y.plus(z)));
    x.power(y).inverted.equal(x.power(y.negated));
    x.power(y).power(y.inverted).equal(x);
};
```

B.2.24.10 Литеральная форма

Литеральными формами вещественного значения являются строка десятичных цифр с необязательным ведущим знаком «+» или «-» и необязательной десятичной точкой, и необязательная экспоненциальная нотация, в которой используется символ «e» (не чувствительный к регистру) между мантиссой и экспонентой. Число значащих цифр должно соответствовать свойству точности precision.

```
REAL.literal ST {
  REAL              : mantissa              { $.equal($1); }
                    | mantissa /[eE]/ INT   { $.equal($1.times(10.power($3)); };

  REAL mantissa     : /0*/ 0                { $.isZero;
                                              $.precision.equal(1); }
                    | /0*/ "." /0*/         { $.isZero;
                                              $.precision.equal($3.length.successor); }
                    | /0*/ "." /0*/ fractional { $.equal($4);
                                              $.precision.equal($4.precision); }
                    | integer               { $.equal($1); }
                    | integer "." fractional { $.equal($1.plus($2));
                                              $.precision.equal(
                                              $1.precision.plus($3.precision)); };

  REAL integer      : uintval               { $.equal($2); }
                    | "+" uintval           { $.equal($1.times($2)); }
                    | "-" uintval           { $.equal($1.times($2).negated); };

  REAL uintval      : /0*/ uint             { $.equal($2); };

  REAL uint         : digit                 { $.equal($1);
                                              $.precision.equal(1); }
                    | uint digit            { $.equal($1.times(10).plus($2));
                                              $.precision.equal(
                                              $1.precision.successor); };
```

```
REAL fractional  : digit                { $.equal($1.times(10.inverted));
                                          $.precision.equal(1); }
                 | digit fractional     { $.equal(
                                          $1.plus($2.times(10.inverted));
                                          $.precision.equal(
                                          $1.precision.successor); };

INT digit        : /[0-9]/              { $.equal($1); }
};
```

Примерами литералов вещественного значения «две тысячи» служат 2000, 2000., 2e3, 2.0e+3, +2.0e+3.

Обратите внимание, что литеральная форма не несет информацию о типе данных. Например, «2000» является допустимым представлением как вещественного, так и целого числа. Для отличия от целых значений концевая десятичная точка не используется. Спецификация реализуемой технологии, использующая эту литеральную форму, должна получать информацию о типе значения из других источников.

B.2.24.11 Свойство precision для десятичной формы: INT

Число значащих цифр в десятичном представлении.

Формальное определение точности основано на литеральной форме.

Атрибут precision характеризует только точность десятичного представления, а не точность вещественного значения.

Назначением свойства precision вещественного типа данных является достоверная передача всей информации, представляемой человеку в форме числа. Количество показываемых десятичных цифр несет информацию о степени определенности измеряемого значения (например, число значащих цифр и точность).

Примечание — Точность представления не зависит от степени неопределенности (степени точности) результата измерений. Если важно знать степень неопределенности результата измерений, то этот результат надо передавать как значение типа PPD.

Действуют следующие правила определения значащих цифр:

1) все ненулевые цифры являются значащими;

2) все нули справа от значащей цифры являются значащими;

3) если все цифры числа являются нулями, то первый нуль слева от десятичной точки являются значащими (и в соответствии с правилом 2 все следующие нули также являются значащими).

Примечание — Эти правила определения значащих цифр несколько отличаются от более привычных правил, преподаваемых в школе. А именно, здесь все концевые нули перед десятичной точкой последовательно считаются значащими. Иначе, например, в числе 2000 нельзя определить, какие нули являются значащими. Это отклонение от привычного правила предназначено для обеспечения однозначной коммуникации.

Таблица B.30 — Примеры точности литералов вещественных чисел

Литерал	Число значащих цифр
2000	4 значащие цифры
2e3	1 значащая цифра, само число могло бы быть записано как «2000», но точность составляет только один десятичный разряд
0.001	1 значащая цифра
1e-3	1 значащая цифра, само число могло бы быть записано как «0.001», но точность составляет только один десятичный разряд
0	1 значащая цифра.
0.0	2 значащие цифры
000.0	2 значащие цифры
0.00	3 значащие цифры
4.10	3 значащие цифры
4.09	3 значащие цифры
4.1	2 значащие цифры

Точность представления должна совпадать с неопределенностью значения. Однако точность представления и неопределенность значения представляют собой два независимых понятия. Детальное обсуждение неопределенности вещественных значений приведено в подразделе PPD<REAL>.

Например, представление «0.123» имеет 3 значащие цифры, но неопределенность значения может быть в любом показанном или не показанном десятичном разряде, например, неопределенность может быть 0.123 ± 0.0005, 0.123 ± 0.005 или 0.123 ± 0.00005 и т. д. Учтите, что спецификация реализуемой технологии должна согласовывать свою точность представления с неопределенностью значения. Но поскольку точность строки цифр кратна 0.5 от наименьшего значащего разряда, то неопределенность принимать любое значение между этими «пределами» и 0.123 ± 0.005 будет также адекватным представлением значения, находящегося в диапазоне от 0.118 до 0.128.

Примечание — в спецификации реализуемой технологии, основанной на представлении чисел в форме десятичных строк, точность не обязательно должна быть явным атрибутом. В этом случае подобная спецификация должна предусматривать правила однозначного определения значащих цифр. Представление числа должно обеспечивать не больше, но и не меньше значащих цифр, изначально имевшихся у этого числа. Соответствие можно проверить с помощью циклического преобразования кодирование — декодирование — кодирование.

B.2.25 Тип данных Ratio (RTO) (специализация типа данных QTY)

Величина, представленная как отношение величины. Указанной в числителе, к величине, указанной в знаменателе. Общие множители числителя и знаменателя автоматически не сокращаются. Тип данных RTO обеспечивает представление титров (например, «1:128») и других величин, измеряемых в лабораториях и действительно представляющих отношения. Такие отношения не являются просто «структурированными числами», в частности, результаты измерения артериального давления (например, «120/60») не являются отношениями. Во многих случаях вместо типа данных RTO можно использовать тип данных REAL.

Отношения отличаются от рациональных чисел тем, что в них общие множители числителя и знаменателя никогда не сокращаются. Отношение двух вещественных или целых чисел автоматически не понижается до вещественного числа.

Таблица B.31 — Сводка свойств типа данных Ratio

Имя	Тип	Описание
numerator	N	Числитель — значение, представляющее собой делимое в отношении. По умолчанию является целым числом 1 (один)
denominator	D	Знаменатель — значение, представляющее собой делитель в отношении. По умолчанию является целым числом 1 (один). Делитель не должен равняться нулю

Примечание — Этот тип данных определен не для общего представления рациональных чисел. Он используется только в том случае, когда общие множители числителя и знаменателя не предполагается сокращать. Такое происходит не часто. При передаче результатов измерений отношения встречаются почти исключительно в титрах.

```
type Ratio<QTY N, QTY D> alias RTO specializes QTY {
  N          numerator;
  D          denominator;
  demotion   REAL;
  demotion   PQ;
};
```

По умолчанию свойства numerator и denominator имеют целое значение 1 (один.) Свойство denominator не может иметь нулевое значение.

Примечание — Этот тип данных определен как параметризованный (см. B.1.9.5 «Параметризованные типы данных»), но обсуждается в контексте других типов данных, связанных с величинами. Причина определения типа данных отношения RTO как параметризованного состоит в том, что в этом случае можно точно ограничить типы данных числителя и знаменателя.

B.2.25.1 Свойство numerator (числитель): N

Величина, представляющая собой делимое в отношении. По умолчанию является целым числом 1 (один).

B.2.25.2 Свойство denominator (знаменатель): D

Величина, представляющая собой делитель в отношении. По умолчанию является целым числом 1 (один). Делитель не должен равняться нулю.

```
invariant(RTO x)
    where x.nonNull {
  x.denominator.isZero.not;
};
```

B.2.25.3 Литеральная форма

Литеральная форма существует для всех отношений, у которых и числитель, и знаменатель имеют литеральные формы. Литерал отношения представляет собой литерал числителя, за которым в качестве разделителя следует двоеточие, а за ним литерал знаменателя. Если двоеточие и знаменатель отсутствуют, то по умолчанию в качестве знаменателя предполагается целое число 1.

```
RTO.literal ST {
  RTO   : QTY            { $.numerator.equal($1);
                           $.denominator.equal((INT)1); };
        | QTY ":" QTY    { $.numerator.equal($1);
                           $.denominator.equal($3); };
};
```

Например, значение титра антител к вирусу краснухи 1:64 может быть представлено, используя литерал «1:64».

B.2.26 Тип данных PhysicalQuantity (PQ) (специализация типа данных QTY)

Размерностная величина, представляющая результат измерения.

Таблица 32 — Сводка свойств типа данных PhysicalQuantity

Имя	Тип	Описание
value	REAL	Количественное значение, измеренное в соответствующих единицах физической величины
unit	CS	Единица измерения, указанная согласно системе Унифицированных кодов единиц измерения UCUM (Unified Code for Units of Measure) [http:///unitsofmeasure.org]
translation	SET<PQR>	Альтернативное представление той же самой физической величины, выраженной в других единицах, или в другой системе кодирования единиц, возможно, с другим количеством
canonical	PQ	Физическая величина, представленная в канонических единицах. В любой конкретной системе единиц каждой физической размерности может быть присвоена одна каноническая единица. Определение канонических единиц не входит в область применения настоящей спецификации, утверждается лишь, что такие канонические единицы существуют (и могут быть произвольно выбраны) для каждой физической величины. Абстрактная физическая величина равна своей канонической форме
diffType	TYPE	Тип данных разности двух значений конкретного типа данных QTY
toPQ	REAL	

```
type PhysicalQuantity alias PQ specializes QTY {
        REAL        value;
        CS          unit;
        BL          equal(ANY x)
        BL          lessOrEqual(PQ x);
        BL          compares(PQ x);
        PQ          canonical;
        SET<PQR>    translation;

        PQ          negated;
        PQ          times(REAL x);
```

```
            PQ          times(PQ x);
            PQ          inverted;
            PQ          power(INT x);
            BL          isOne;

  literal   ST;
  demotion  REAL;

            REAL        originalValue;
            CV          originalUnit;
};
```

B.2.26.1 Свойство value: REAL

Количество величины, измеренное в терминах единиц.

B.2.26.2 Свойство unit: CS

Единица измерения, указанная согласно системе Унифицированных кодов единиц измерения UCUM (Unified Code for Units of Measure) [http:///unitsofmeasure.org].

Примечание — Для равенства физических величин не требуется, чтобы у них соответственно совпадали свойства value и unit. Эти свойства относятся только к способу представления физических величин. К примеру, 1 м равен 100 см. Хотя различны и единицы, и количества, эти физические величины равны. Таким образом, надо не рассчитывать на то, что физическая величина имеет конкретную единицу измерения, а обеспечивать автоматическое преобразование разных сопоставимых единиц.

B.2.26.3 Свойство translation : SET<PQR>

Альтернативное представление той же самой физической величины, выраженной в других единицах, или в другой системе кодирования единиц, возможно, с другим количеством.

Семантически физические величины являются результатами действий измерения. Хотя эти величины представляются как пары «значение-единица измерения», семантически физическая величина не ограничивается этим. Для определения, являются ли две физические величины равными, недостаточно провести независимое сравнение их значений и единиц измерения. К примеру, 1 м равен 100 см, но различны и единицы, и количества. Для установления равенства вводится понятие канонической формы.

B.2.26.4 Свойство canonical: PQ

Физическая величина, представленная в канонических единицах. В любой конкретной системе единиц каждой физической размерности может быть присвоена одна каноническая единица. Определение канонических единиц не входит в область применения настоящей спецификации, утверждается лишь, что такие канонические единицы существуют (и могут быть произвольно выбраны) для каждой физической величины. Абстрактная физическая величина равна своей канонической форме.

```
invariant(PQ x, y)
    where x.nonNull.and(y.nonNull) {
  x.canonical.equal(x);
};
```

Например, в системе, основанной на Международной системе единиц (СИ), можно определить каноническую форму как а) произведение только базовых единиц, б) без приставок, в) с использованием только умножения и возведения в степень (без операций деления), г) с определенным порядком указания базовых единиц (например, m, s, g, ...). В этом случае 1 мм рт. ст. можно представить как 133322 m-1 s-2. Как видно, правила образования канонической формы могут быть довольно сложными. Однако для семантической спецификации не имеет значения ни как конструируется каноническая форма, ни как выбирается конкретная каноническая форма. Важно лишь, что некоторая каноническая форма может быть определена.

B.2.26.5 Свойство equal: BL (унаследовано от типа данных ANY)

Две физические величины равны, если у их канонических форм равны значения и единицы измерения.

```
invariant(PQ x, y)
    where x.nonNull.and(y.nonNull) {
  x.equal(y).equal(x.canonical.value
        .equal(y.canonical.value).and(
          x.canonical.unit.equal(y.canonical.unit)));
};
```

B.2.26.6 Свойство compares: BL (унаследовано от типа данных QTY)

Две физические величины сравнимы (упорядочены и имеют разность), если у их канонических форм равны единицы измерения.

```
invariant(PQ x, y)
    where x.nonNull.and(y.nonNull) {
  x.compares(y).equal(x.canonical.unit.equal(y.canonical.unit));
};
```

B.2.26.7 Свойство diffType: TYPE (унаследовано от типа данных QTY)

```
invariant(PQ x) {
  x.diffType.implies(PQ);
};
```

Разностью двух физических величин является другая физическая величина с теми же самыми единицами измерения.

```
invariant(PQ x, y)
    where x.compares(y) {
  x.minus(y).canonical.unit.implies(x.canonical.unit);
};
```

B.2.26.8 Свойство isOne: BL (нейтральный элемент умножения)

Предикат, указывающий, что данное значение равно числу один, то есть нейтральному элементу умножения. Существует ровно одна физическая величина с таким свойством, называемая единицей.

```
invariant(PQ x, y)
    where x.nonNull.and(y.nonNull) {
  x.isOne.and(y.isOne).implies(x.equal(y));
  x.isOne.and(y.isZero).implies(x.equal(y).not);
      };
```

B.2.26.9 Свойство times: PQ (умножение)

Произведением двух физических величин является произведение их количеств, умноженное на произведение их единиц.

```
invariant(PQ x, y, z, i, o)
    where x.nonNull.and(y.nonNull).and(z.nonNull)
        .and(o.isZero).and(i.isOne) {
  x.times(o).equal(o);
  x.times(i).equal(x);                  /* нейтральный элемент */
  x.times(y).times(z).equal(
       x.times(y.times(z)));            /* ассоциативность */
  x.times(y).equal(y.times(x));         /* коммутативность */
  o.lessOrEqual(x).and(o.lessOrEqual(y).implies(o.lessOrEqual(x.times(y));
};
```

B.2.26.10 Свойство inverted: PQ (обратная величина)

Значение типа PQ, которое, будучи умноженным на другое значение типа PQ, дает единицу (нейтральный элемент умножения). Нуль (нейтральный элемент сложения) не имеет обратной величины. Отношение двух сравнимых величин сравнимо с единицей (единица измерения 1).

```
invariant(PQ this, that, one)
    where this.nonNull.and(that.nonNull).and(one.isOne) {
  this.times(this.inverted).equal(one);
  this.compares(that).implies(this.times(that.inverted).equal(one));
};
```

B.2.26.11 Свойство times: PQ (умножение на вещественное число)

При умножении на вещественное число образуется масштабированная величина. Эта величина сравнима с ее исходной величиной.

Если две величины Q1 и Q2 сравнимы друг с другом, то существует вещественное число r, для которого r1 = Q1 / Q2.

```
invariant(PQ x; REAL r)
    where x.nonNull.and(r.nonNull) {
  x.times(r).value.equal(x.value.times(r));
  x.times(r).compares(x);
};
```

B.2.26.12 Гомоморфизм значения типа REAL в величину типа PQ: REAL

Значение типа REAL может быть преобразовано в значение типа PQ с единицей 1 (один). Аналогично, физическая величина, сравнимая с единицей, может быть преобразована в вещественное значение.

```
invariant(PQ x, unity)
    where x.nonNull.and.unity.isOne.and(x.compares(unity)) {
  unity.times((REAL)x).equal(x);
};
```

B.2.26.13 Свойство power: PQ (возведение в степень)

Физическая величина может быть возведена в целую степень.

```
invariant (PQ x, i; INT n, o)
    where x.nonNull.and(i.isOne).and(n.nonNull.and(o.isZero) {
  x.power(o).equal(i);
  n.greaterThan(o).implies(
    x.power(n).equal(x.times(x.power(n.predecessor))));
  n.lessThan(o).implies(
    x.power(n).equal(x.power(n.negated).inverted);
};
```

B.2.26.14 Свойство plus: PQ (сложение)

Две сравнимые физические величины могут быть сложены.

```
invariant (PQ x, y)
    where x.compares(y) {
  x.canonical.plus(y.canonical).value.equal(
    x.canonical.value.plus(y.canonical.value));
};
```

B.2.26.15 Литеральная форма

Литеральной формой физической величины является литерал вещественного числа, за которым следуют необязательный пробельный символ и строка символов, представляющая допустимый код, взятый из системы Унифицированных кодов единиц измерения UCUM (Unified Code for Units of Measure) [http:///unitsofmeasure.org].

```
PQ.literal ST {
  PQ       :  REAL unit   { $.value.equal($1);
                            $.unit.equal($2); }
  CS  unit :  ST          { $.value.equal($1);
                            $.codeSystem.equal(2.16.840.1.113883.6.8); };
};
```

Например, литеральная форма 20 минут имеет вид «20 min».

B.2.27 Тип данных PhysicalQuantityRepresentation (PQR) (специализация типа данных CV)

Расширение типа данных кодированного значения (CV), представляющее физическую величину, у которой единица берется из какой-либо системы кодирования. Используется для альтернативного представления физической величины.

Таблица B.33 — Сводка свойств типа данны PhysicalQuantityRepresentation

Имя	Тип	Описание
value	REAL	Количественное значение, измеренное в единицах физичес-кой величины, указанных в конкретной системе кодирования
code	ST	Символ кода, определенный в системе кодирования. Например, «784.0» является символом кода головной боли «784.0», определенным в системе кодирования МКБ-9
codeSystem	UID	Указывает систему кодирования, в которой определен код.
codeSystemName	ST	Общее имя системы кодирования
codeSystemVersion	ST	Если применим, дескриптор версии, определенный специально для данной системы кодирования
displayName	ST	Имя или название кода, под которым система-отправитель показывает значение кода своим пользователям
originalText	ED	Текст или фраза, используемые в качестве основы для кодирования

```
type PhysicalQuantityRepresentation alias PQR specializes CV {
  REAL value;
};
```

B.2.27.1 Свойство value: REAL

Измеренное количество величины в терминах единиц, указанных в свойстве code.

B.2.27.2 Свойство code: ST (унаследовано от типа данных CV)

Символ кода, определенный в системе кодирования. Например, «784.0» является символом кода головной боли «784.0», определенным в системе кодирования МКБ-9.

Не исключительное значение типа данных PQR имеет непустое свойство code, значением которого является строка символов, определенная в системе кодирования, идентифицированной в свойстве codeSystem. И наоборот, значение типа PQR, у которого свойство code не имеет значения или имеет значение, не принадлежащее указанной системе кодирования, считается исключительным значением (пустым значением (NULL) с типом пустоты «other»).

```
invariant(PQR x)
    where x.nonNull {
  x.code.nonNull;
};
```

B.2.27.3 Свойство codeSystem: UID (унаследовано от типа данных CV)

Указывает систему кодирования, в которой определено значение свойства code.

Идентификатор системы кодирования должен иметь тип данных UID, позволяющий однозначно указать стандартные системы кодирования HL7, другие стандартные системы кодирования, а также местные системы кодирования. Комитет HL7 должен присваивать идентификатор типа UID каждой из своих таблиц кодов, а также внешним стандартным системам кодирования, которые используются в стандартах HL7. На местах должны использовать свои объектные идентификаторы ИСО (тип данных OID), с помощью которых можно сконструировать глобально уникальные идентификаторы местных систем кодирования.

Под ветвью комитета HL7, 2.16.840.1.113883, подветви 5 и 6 содержат соответственно идентификаторы стандартных систем кодирования HL7 и внешних систем кодирования. Эти ветви ведутся техническим комитетом HL7 Vocabulary Technical Committee.

Не исключительное значение типа PQR (то есть значение типа PQR с непустым свойством code) имеет непустое свойство codeSystem, указывающее систему понятий, в которой определено значение свойства code. Другими словами, если есть код, должна быть и система кодирования.

Примечание — Хотя для каждого непустого значения типа PQR определена конкретная система кодирования, в некоторых обстоятельствах представление значения типа PQR в соответствии со спецификацией

реализуемой технологии не нуждается в явном упоминании системы кодирования. Например, когда контекст подразумевает одну и только одну систему кодирования, то ее явное указание стало бы избыточным. Однако в таком случае свойство codeSystem принимает контекстно-зависимое значение по умолчанию и не является пустым.

```
invariant(PQR x)
    where x.code.nonNull {
  x.codeSystem.nonNull;
};
```

Причина пустоты «other» у исключительного значения типа PQR указывает, что понятие не может быть закодировано в указанной системе кодирования. Эта система кодирования, в которой нет такого исключительного понятия, должна быть указана в свойстве codeSystem.

Некоторые домены кодов квалифицированы таким образом, что они могут включать в себя некоторые части подходящей местной системы кодирования, не являющиеся парафразами стандартной системы кодирования (coded with extensibility, CWE — кодированные с расширением.) Если поле с квалификатором «CWE» действительно содержит такой местный код, то в свойстве системы кодирования должен быть указан идентификатор местной системы кодирования, из которой взят этот код. Однако в доменах с квалификатором «CWE» местный код является допустимым членом домена, поэтому использование местного кода не является ни ошибкой, ни исключительным значением (с причиной пустоты «other») в смысле настоящей спецификации.

```
invariant(PQR x)
    where x.other {
  x.code.other;
  x.codeSystem.nonNull;
};
```

B.2.27.4 Свойство codeSystemName: ST (унаследовано от типа данных CV)

Общее имя системы кодирования.

Имя системы кодирования не используется для вычислений. Оно может быть указано для облегчения интерпретации человеком значений свойств code и codeSystem. Предполагается, хотя и не является обязательным, что в спецификациях реализуемой технологии будет предусматриваться указание значения свойства codeSystemName в качестве аннотации к идентификатору UID, рассчитанной на восприятие человеком.

Системы, соответствующие стандарту HL7, не должны функционально полагаться на значение свойства codeSystemName. Это значение не должно модифицировать значение свойства codeSystem и не может существовать без значения свойства codeSystem.

```
invariant(PQR x) {
  x.codeSystemName.nonNull.implies(x.codeSystem.nonNull);
};
```

B.2.27.5 Свойство codeSystemVersion: ST (унаследовано от типа данных CV)

Если применим, дескриптор версии, определенный специально для данной системы кодирования.

Для каждой внешней системы кодирования в стандарте HL7 будет указано, как формируется строка со значением версии. Если для конкретной системы кодирования такое указание отсутствует, то для такой системы обозначение версии не имеет определенного значения.

Различные версии одной и той же системы кодирования должны быть совместимыми. Если система кодирования изменена несовместимым образом, то она представляет собой другую систему кодирования, а не другую версию, как бы издатель это не называл.

Например, издатель классификаций МКБ-9 и МКБ-10 назвал эти системы кодирования «9-м пересмотром» и «10-м пересмотром». Однако МКБ-10 представляет собой полное изменение кодов МКБ, и не является обратно совместимой. Поэтому в целях настоящей спецификации МКБ-9 и МКБ-10 рассматриваются как разные системы кодирования, а не как разные версии одной системы кодирования. Напротив, когда версия «1.0j» системы кодирования LOINC была обновлена до версии «1.0k», комитет HL7 рассматривал это как смену версий, поскольку новая версия была обратно совместимой с предыдущей.

```
invariant(PQR x) {
  x.codeSystemVersion.nonNull.implies(x.codeSystem.nonNull);
};
```

B.2.27.6 Свойство displayName: ST (унаследовано от типа данных CV)

Имя или название кода, под которым система-отправитель показывает значение кода своим пользователям.

Свойство displayName включено как для удобства интерпретации человеком значения кода, так и для документирования имени, используемого для изображения понятия пользователю. Оно не имеет функционального значения, не может существовать без кода и никогда не должно модифицировать смысл кода.

Примечания

1 В своих словарных доменах стандарты HL7 предусматривают атрибут «печатаемого имени» (print name). Значения этого атрибута могут использоваться в качестве значений свойства displayName.

2 Имена, предусмотренные для кодов, не могут менять смысл кодированного значения. Поэтому они не должны предоставляться пользователю прикладной системы-получателя, пока не будет уверенности в том, что такое имя адекватно отражает понятие, соответствующее кодированному значению. При коммуникациях нельзя просто полагаться на имя кодированного значения. Основная цель этого имени в обеспечении возможности отладки единиц данных в протоколах HL7 (например, сообщений).

```
invariant(PQR x) {
  x.displayName.nonNull.implies(x.code.nonNull);
};
```

B.2.27.7 Свойство originalText: ED (унаследовано от типа данных CV)

Текст или фраза, используемые в качестве основы для кодирования.

Исходный текст появляется в тех случаях, когда код был присвоен информации не ее источником, а специальным кодировщиком уже после ее появления. Тогда при создании дескриптора понятия исходный текст может существовать без кода.

Примечание — Хотя кодировка пост-фактум часто осуществляется по информации, представленной как свободный текст, например, в форме документов, сканированных изображений или диктовки, мультимедийные данные явным образом не разрешены в качестве исходного текста. Кроме того, свойство исходного текста originalText не может означать ссылку на весь исходный документ. Связи между различными артефактами медицинской информации (например, документом и кодированным результатом) не входят в область применения настоящей спецификации и рассматриваются в других стандартах HL7. Исходный текст должен представлять собой извлечение из оригинального источника, а не точную копию его содержания или указатель на этот источник. Поэтому исходный текст должен представляться в неформатированном виде.

Значение типа PQR могут иметь непустое свойство исходного текста, даже если свойство code имеет пустое значение. В этом случае свойство originalText является именем или описанием понятия, которое не было закодировано. Такие исключительные значения типа PQR могут также содержать преобразования translation. Такие преобразования обеспечивают непосредственное кодирование понятия, описанного в свойстве originalText.

Тип данных PQR может быть понижен до типа данных ST. В этом случае значение типа ST будет представлено только свойством originalText значения типа CD.

```
invariant(PQR x)
    where x.originalText.nonNull {
  ((ST)x).equal(x.originalText);
};
```

B.2.28 Тип данных MonetaryAmount (MO) (специализация типа данных QTY)

Тип данных MO является величиной, представляющей денежную сумму в некоторой валюте, то есть в денежных единицах, используемых в различных экономических регионах. В то время как денежная сумма представляет собой простой вид величины (деньги), курс обмена валют является переменным. В этом и состоит принципиальная разница между типами данных PQ и MO, по которой денежные единицы не являются физическими единицами.

Таблица B.34 — Сводка свойств типа данных MonetaryAmount

Имя	Тип	Описание
value	REAL	Денежная сумма в единицах некоторой денежной системы
currency	CS	Код денежной единицы в соответствии с ИСО 4217
diffType	TYPE	Тип данных разности между двумя значениями конкретного типа данных QTY

```
type MonetaryAmount alias MO specializes QTY {
        REAL  value;
        CS    currency;
        MO    negated;
        MO    times(REAL x);
  literal ST;
};
```

B.2.28.1 Свойство value: REAL

Денежная сумма в терминах денежной единицы currency.

Примечание — Значения типа MO обычно имеют точность до 0.01 (один цент, пенс, пайса и т. д.). Для больших сумм важно не хранить значения типа MO в форме вещественных чисел с плавающей точкой, поскольку это может привести к потере точности. Однако настоящая спецификация не определяет внутреннее представление значений типа REAL в форме чисел с фиксированной или плавающей точкой.

Свойство REAL.precision означает точность десятичного представления вещественного значения, а не точность самого значения. В определении типа данных REAL отсутствует понятие неопределенности или точности. Например, результат произведения «1.99 USD» (точность 3) на 7 равен «13.93 USD» (точность 4) и не должен округляться до «13.9» для сохранения исходной точности.

B.2.28.2 Свойство currency: CS

Код денежной единицы в соответствии со стандартом ИСО 4217.

Таблица B.35 — Домен значений свойства currency

Код	Имя	Определение
ARS	Argentine Peso	Аргентинское песо, денежная единица Аргентины
AUD	Australian Dollar	Австралийский доллар, денежная единица Австралии
BRL	Brazilian Real	Бразильский реал, денежная единица Бразилии
CAD	Canadian Dollar	Канадский доллар, денежная единица Канады
CHF	Swiss Franc	Швейцарский франк, денежная единица Швейцарии
CLF	Unidades de Formento	Единица развития, денежная единица Chile
CNY	Yuan Renminbi	Юань, денежная единица China
DEM	Deutsche Mark	Немецкая марка, денежная единица Германии
ESP	Spanish Peseta	Испанская песета, денежная единица Испании
EUR	Euro	Евро, денежная единица Европейского Союза
FIM	Markka	Марка, денежная единица Финляндии
FRF	French Franc	Французский франк, денежная единица Франции
GBP	Pound Sterling	Фунт стерлингов, денежная единица Великобритании
ILS	Shekel	Шекель, денежная единица Израиля
INR	Indian Rupee	Индийская рупия, денежная единица Индии
JPY	Yen	Йена, денежная единица Японии
KRW	Won	Вона, денежная единица Кореи (Южной)
MXN	Mexican Nuevo Peso	Мексиканский новый песо, денежная единица Мексики
NLG	Netherlands Guilder	Голландский гульден, денежная единица Нидерландов
NZD	New Zealand Dollar	Новозеландский доллар, денежная единица Новой Зеландия
PHP	Philippine Peso	Филлипинский песо, денежная единица Филлипин
RUR	Russian Ruble	Российский рубль, денежная единица Российской Федерации

Окончание таблицы B.35

Код	Имя	Определение
THB	Baht	Бат, денежная единица Тайланда
TRL	Lira	Лира, денежная единица Турции
TWD	Taiwan Dollar	Тайваньский доллар, денежная единица Тайваня
USD	US Dollar	Доллар США, денежная единица США
ZAR	Rand	Ранд, денежная единица Южной Африки

В таблице B.35 показано только представительное подмножество кодов, определенных в стандарте ИСО 4217. Все коды из стандарта ИСО 4127 являются допустимыми значениями этого атрибута.

B.2.28.3 Свойство equal: BL (унаследовано от типа данных ANY)

Два значения типа MO равны, если совпадают значения их атрибутов value и currency.

```
invariant(MO x, y)
    where x.nonNull.and(y.nonNull) {
  x.equal(y).equal(x.value.equal(y.value)
        .and(x.unit.equal(y.unit)));
};
```

B.2.28.4 Свойство compares: BL (унаследовано от типа данных QTY)

Два значения типа MO сравнимы (упорядочены и имеют разность), если равны их свойства currency.

Если их свойства currency не идентичны, то нельзя сравнивать значения их свойств values. Конверсия валют не входит в область применения настоящей спецификации. На практике обменные курсы валют чрезвычайно изменчивы не только в течение длительных и кратких периодов времени, но даже зависят от места и доступности обменных пунктов.

```
invariant(MO x, y)
    where x.nonNull.and(y.nonNull) {
  x.compares.equal(x.currency.equal(y.currency));
};
```

B.2.28.5 Тип данных diffType: TYPE (унаследован от типа данных QTY)

```
invariant(INT x) {
  x.diffType.implies(MO);
};
```

Разность между двумя значениями типа MO имеет тип данных MO.

B.2.28.6 Свойство plus: MO

Два значения типа MO можно сложить, если у них равны значения свойство currency.

```
invariant (MO x, y)
    where x.currency.equal(y.currency) {
  x.plus(y).currency.equal(x.currency);
  x.plus(y).value.equal(x.value.plus(y.value));
};
```

B.2.28.7 Свойство times: MO (умножение на значение типа REAL)

При умножении на значение типа REAL образуется масштабированная сумма. Она сравнима с исходной суммой.

```
invariant(MO x; REAL r)
    where x.nonNull.and(r.nonNull) {
  x.times(r).value.equal(x.value.times(r));
  x.times(r).currency.equal(x.currency);
};
```

B.2.28.8 Литеральная форма

Литеральная форма типа данных MO состоит из кода валюты, необязательного пробельного символа и литерала значения REAL, представляющего величину суммы.

```
MO.literal ST {
  MO          : currency value     { $.currency.equal($1); }
                                     $.value.equal($2);
  CS currency : ST                 { $.currency.value.equal($1);
                                     $.currency.codeSystem
                                     .equal(2.16.840.1.113883.6.9); }
  REAL value  : REAL               { $.value.equal($1); }
};
```

Например, строка «USD189.95» является литералом суммы 189.95 долларов США.

B.2.29 Тип данных Calendar (CAL) (специализация типа данных SET<CLCY>)

Календарь представляет собой понятие измерения различных циклов времени. Такими циклами являются годы, месяцы, дни, часы, минуты, секунды и недели. Некоторые из этих циклов синхронизированы, а некоторые — нет (например, недели и месяцы не синхронизированы).

После «сворачивания оси времени» в эти циклы календарь представляет момент времени как последовательность целых значений счетчиков циклов, например, год, месяц, день, час и т. д. Точкой отсчета календаря является некоторый согласованный момент времени, называемый «эрой».

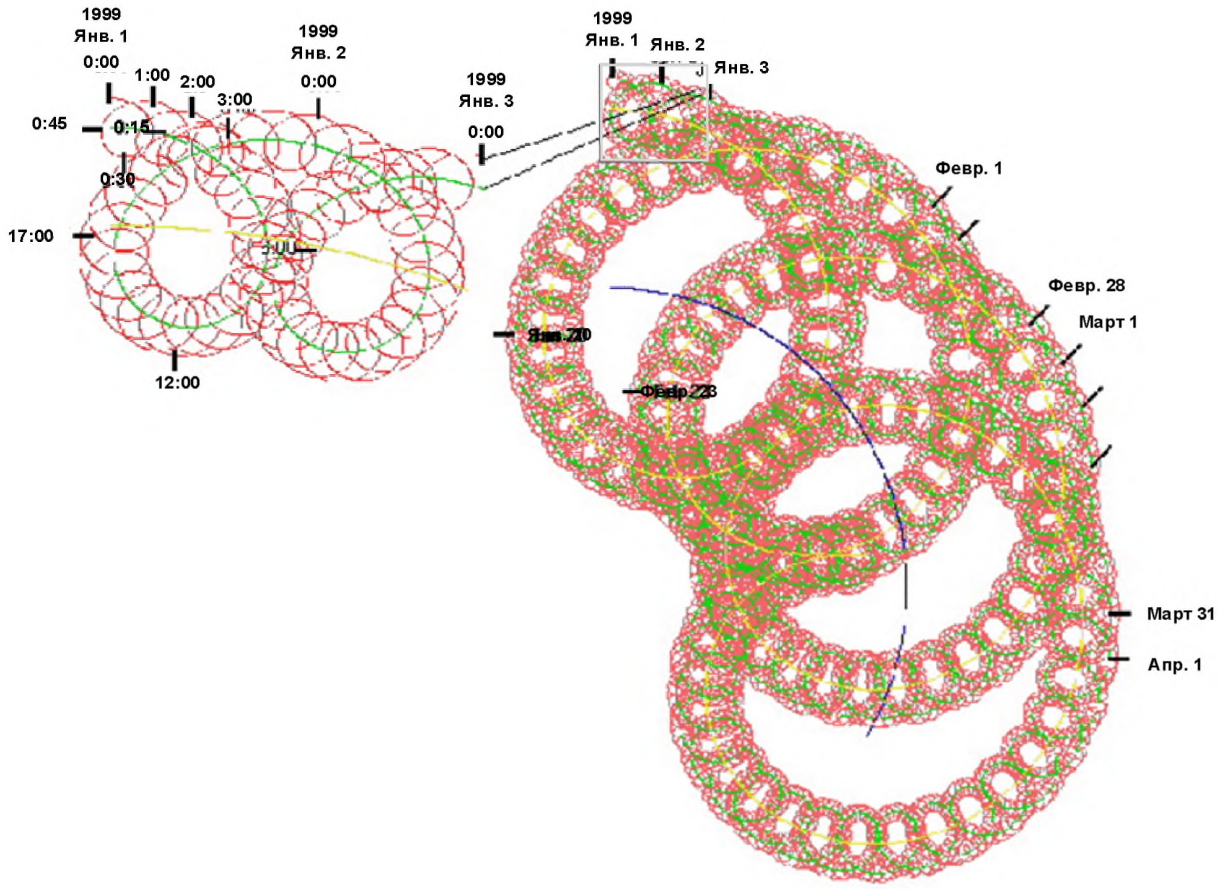


Рисунок B.10 — Календарные циклы

Календарь «сворачивает» ось времени в сложную спираль в соответствии с периодами календаря год (синий), месяц (желтый), день (зеленый), час (красный) и т. д. Разные циклы могут не совпадать по фазе, например, неделя (не показана) не совпадает по фазе с месяцем[52)].

Тип данных календаря Calendar имеет имя и код и определяется как множество календарных циклов. Свойство Calendar.head представляет наибольший календарный цикл, который в календарном выражении указан крайним справа. Свойство epoch является началом отсчета календаря, то есть моментом начала всех календарных циклов.

```
private type Calendar alias CAL specializes SET<CLCY> {
  CS   name;
  CLCY head;
  TS   epoch;
};

invariant(CAL c)
    where c.nonNull {
  c.name.nonNull;
  c.contains(c.head);
};
```

В таблице В.36 показано определение современного Григорианского календаря. В каждой строке этой таблицы показан календарный цикл. Календарные единицы зависят друг от друга и показаны в столбце «Имя». В столбце «Порядок» показан порядок следования свойств. Другие столбцы показаны в соответствии с формальным определением календарного цикла[53)].

Таблица В.36 — Домен календарных циклов

Имя	Код 1	Код 2	Порядок	Цифры	Начало	Условие
year (год)	Y	CY	1	4	0	
month of the year (месяц года)	M	MY	2	2	1	
month (continuous) (непрерывный месяц)		CM			0	
week (continuous) (непрерывная неделя)	W	CW			0	
week of the year (неделя года)		WY		2	1	
day of the month (день месяца)	D	DM	3	2	1	
day (continuous) (непрерывный день)		CD			0	
day of the year (день года)		DY		3	1	
day of the week (begins with Monday) (день недели, начиная с понедельника)	J	DW		1	1	
hour of the day (час дня)	H	HD	4	2	0	
hour (continuous) (непрерывный час)		CH			0	

[52)] Представьте себе специальные часы, измеряющие эти циклы, в которых все стрелки не имеют общей оси, а каждая стрелка прикреплена к концу другой стрелки, измеряющей следующий больший цикл.

[53)] В настоящее время свойства sum и value календарного цикла формально не определены. Вычисление цифр календаря включает в себя довольно сложный алгоритм, описание которого в настоящей спецификации было бы сложным для понимания и оценки правильности. К несчастью, не существует стандарта, содержащего формальное определение связей между календарными выражениями и временем, прошедшим от начала эры. Спецификации языка ACH.1, типов данных XML-схемы и языка SQL92 ссылаются на стандарт ISO 8601, однако этот стандарт описывает только синтаксис календарных выражений григорианского календаря, а не их семантику. В настоящем стандарте формально определены и синтаксис, и семантика, однако семантика свойств sum и value в нем не описана.

Окончание таблицы В.36

Имя	Код 1	Код 2	Порядок	Цифры	Начало	Условие
minute of the hour (минута часа)	N	NH	5	2	0	
minute (continuous) (непрерывная минута)		CN			0	
second of the minute (секунда минуты)	S	SN	6	2	0	
second (continuous) (непрерывная секунда)		CS			0	

В.2.30 Тип данных календарного цикла CalendarCycle (CLCY) (специализация типа данных ANY)

Календарный цикл определяет одну группу цифр в календарном выражении. Примерами календарных циклов служат год, месяц, день, час, минута, секунда и неделя.

Календарный цикл имеет имя и два кода, один состоит из одной буквы, а другой из двух букв. Свойство ndigits указывает число десятичных цифр, занимаемых в календарном выражении. Свойство start указывает, с какого значения начинается отсчет (например, с 0 или 1). Свойство next указывает следующий по порядку меньший цикл в календарном выражении. Свойство max(t) представляет максимальное число циклов за время t (зависит от времени t с учетом високосных лет и секунд). Свойство value(t) представляет целое число циклов, показанных в календарном выражении для времени t. Свойство sum(t, n) представляет собой результат сложения n календарных циклов с временем t.

```
private type CalendarCycle alias CALCY specializes ANY {
  CE      name;
  INT     ndigits;
  INT     start;
  CALCY   next;
  INT     max(TS);
  TS      sum(TS t, REAL r);
  INT     value(TS t);
};

invariant(CALCY c)
    where c.nonNull {
  c.name.nonNull;
  c.start.equal(0).or(c.start.equal(1));
  c.digits.greaterThan(0);
};
```

В.2.31 Тип данных момента времени PointInTime (TS) (специализация типа данных QTY)

Величина, указывающая момент на оси естественного времени. Момент времени чаще всего представляют как календарное выражение.

Однако семантически время не зависит от календарей и лучше всего может быть описано с помощью связи с прошедшим временем (измеряемому как физическая величина, имеющая размерность времени). Сумма значения типа TS с прошедшим временем дает другое значение типа TS. И наоборот, вычитание значения типа TS из другого значения типа TS дает прошедшее время.

Поскольку никто не знает, когда началось время, тип данных TS концептуализируется как количество времени, прошедшее с некоторого произвольного момента отсчета, называемого началом эры. Так как на оси времени нет абсолютной (нулевой) точки отсчета, то естественное время является разностной величиной, для которой определена только разность, но не отношение. (Например, нельзя сказать, что какой-то момент времени «вдвое больше» другого момента времени.)

Задав некоторую произвольную точку отсчета времени, любой момент времени можно представить в терминах времени, прошедшего от этой точки. Такая произвольная точка отсчета называется началом эры. В настоящей спецификации используется форма смещения от начала эры, но при этом не требуется, чтобы в каждой системе тип данных TS был реализован именно таким способом. Системам, которым не требуется вычислять дистанции между значениями типа TS, достаточно ограничиваться литеральными представлениями календарных выражений.

```
type PointInTime alias TS specializes QTY {
        PQ   offset;
        CS   calendar;
        INT  precision;
        PQ   timezone;
        BL   equal(ANY x);
  literal  ST;
};
```

B.2.31.1 Свойство offset: PQ

Время, прошедшее от любого постоянного начала эры и измеряемое как физическая величина, размерностью которой является время (например, сравнимо с одной секундой).

```
invariant(TS x)
    where x.nonNull {
  x.offset.compares(1 s);
};
```

Настоящая спецификация не требует определения канонического начала эры; семантика остается той же самой для любой эры, при условии, что ее начало постоянно.

Примечание — Свойство offset может трактоваться как чисто семантическое свойство, которое не обязано быть представлено иначе как в литерале календарного выражения. Однако в спецификации реализуемой технологии может быть выбрано собственное постоянное начало эры и значения типа TS могут быть в ней представлены как смещение времени относительно начала эры. Но в такой технологии, использующей другие календари, тем не менее существует потребность в передаче кода календарного цикла и точности календарного представления.

B.2.31.2 Свойство equal: BL (унаследовано от типа данных QTY)

Два значения типа данных TS равны в том и только том случае, если равны их свойства offset (отсчитываемые от одного и того же начала эрыrelative to the same epoch).

```
invariant(TS x, y)
    where x.nonNull.and(y.nonNull) {
  x.equal(y).equal(x.offset.equal(y.offset));
};
```

B.2.31.3 Свойство calendar: CS

Код, указывающий календарь, используемый для литерального представления данного значения типа TS[54].

Таблица B.37 — Домен значений свойства calendar

Код	Имя	Определение
GREG	Gregorian	Григорианский календарь используется в большинстве стран, находящихся под влиянием христианской веры примерно с 1582 года. Он заменил юлианский календарь

Свойство calendar определено в основном для целей достоверного определения. что пользователь должен видеть или вводить в системе, оперирующей данным значением типа TS value. Оно также служит рекомендацией, какой календарь следует использовать системе, отображающей литеральную форму значения типа TS. Однако его значение является лишь рекомендацией и система, отображающая литеральную форму значений типа TS, может выбрать любой календарь и литеральную форму, необходимую ее пользователям, а не обязательно тот календарь, что указан в свойстве calendar. Таким образом, свойство calendar не является константой при передаче данных из одной системы в другую и поэтому не является часть проверки на равенство.

[54] В настоящее время никакие другие календари, кроме григорианского, не определены. Однако понятие календаря как произвольного соглашения об абсолютном времени является важным для правильного определения семантики времени и типов данных, относящихся к времени. Кроме того, для облегчения использования стандартов HL7 другими культурами может потребоваться поддержка иных календарей.

Для целей определения отношения между календарным выражением и значением в форме начала эры/смещения введены два типа данных, а именно, CAL и CLCY, имеющих область видимости private. Они не могут использоваться вне данной спецификации.

B.2.31.4 Свойство precision: INT (точность календарной литеральной формы)

Число значащих цифр в представлении календарного выражения.

Свойство precision формально определено на основе определения свойства literal (см. подраздел B.2.31.9).

Свойство precision характеризует только точность десятичного представления, а не точность самого значения типа TS.

Это свойство предназначено для достоверной оценки информации, представленной человеку в календарном выражении. Показываемое ему число цифр содержит информацию о неопределенности (а именно, степени точности) измеренного значения типа TS.

Примечание — Точность представления не зависит от степени неопределенности (степени точности) результата измерений. Если важно знать степень неопределенности результата измерений, то этот результат надо передавать как значение типа PPD.

Значение свойства precision зависит от значения свойства calendar. Данное значение свойства precision, относящееся к одному календарю, не обязательно будет иметь тот же смысл в календаре с другими периодами.

Например, строка «20000403» имеет 8 значащих цифр в представлении, но значение типа TS, имеющего это представление, может быть неопределенным в любой показанной или не показанной цифре, например, не иметь точного дня, недели или часа. Точность внешнего представления должна коррелировать с точностью самого значения. Однако точность строки цифр зависит от календаря и может быть определена только в терминах календарных периодов, а точность значения может не укладываться в эти периоды (например, строка «2000040317» является адекватным представлением значения, находящегося в границах «2000040305» и «2000040405».)

Примечание — В спецификации реализуемой технологии, основанной на литеральном представлении календарных выражений, точность не обязательно должна быть явным атрибутом. В этом случае подобная спецификация должна предусматривать правила однозначного определения значащих цифр. Представление значения типа TS должно обеспечивать не больше, но и не меньше значащих цифр, изначально имевшихся у этого значения. Соответствие можно проверить с помощью циклического преобразования кодирование — декодирование — кодирование.

B.2.31.5 Свойство timezone: PQ (часовая зона)

Разность между местным временем в данной часовой зоне и Всемирным координируемым временем (Universal Coordinated Time, UTC), которое ранее именовалось средним временем по Гринвичу (Greenwich Mean Time, GMT). Часовая зона имеет тип данных PQ с размерностью времени (то есть сравнима с одной секундой). Нулевая часовая зона означает UTC. Значение часовой зоны не позволяет сделать вывод о географической долготе или о согласованном названии часовой зоны.

Например, значение 200005121800-0500 может быть Восточным стандартным временем (Eastern standard time, EST) в Индианаполисе (Индиана) или Центральным летним временем (Central daylight savings time, CDT) в Декатуре (Иллинойс). В других странах, имеющих иную долготу, часовые зоны могут иметь другие именования.

```
invariant(TS x, y)
    where x.nonNull.and(y.nonNull) {
  x.timezone.compares(1 s);
};
```

Если свойство timezone имеет пустое значение (причина пустоты «не известно»), то подразумевается «местное время». Однако местное время всегда относится к какому-то месту, и без знания этого места часовая зона не известна. Поэтому местное время не может быть переведено в значение UTC. Чтобы избежать значительной потери точности при сравнении значений типа TS, свойство timezone должно быть задано для всех значений типа TS. Разность двух значений местного времени, если не известно, к каким местам они относятся, может иметь ошибку ± 12 часов.

В контексте административных данных некоторые значения времени не должны иметь часовую зону. Например, для административной даты рождения указание часовой зоны является некорректным, поскольку при преобразовании в другую часовую зону эта дата может измениться. У таких административных данных свойство timezone должно быть пустым с причиной пустоты «не применимо».

B.2.31.6 Свойство diffType: TYPE (унаследовано от QTY)

```
invariant(TS x) {
  x.diffType.implies(PQ);
};
```

Свойство diffType означает тип данных разности между двумя значениями типа TS, а именно, тип PQ с размерностью времени.

B.2.31.7 Свойство plus: TS (унаследовано от типа данных QTY)

Свойство plus определяет сложение значения типа TS со значением прошедшего времени (имеющего тип данных PQ с размерностью времени), имеющее тип данных TS.

```
invariant(TS x, PQ t)
    where x.nonNull.and(t.compares(1 s)) {
  x.plus(t).offset.equal(x.offset.plus(t));
};
```

B.2.31.8 Свойство minus: QTY (унаследовано от типа данных QTY)

Свойство minus представляет собой разность двух значений типа TS, являющуюся прошедшим временем.

```
invariant(TS x)
    where x.nonNull {
  x.minus(y).offset.equal(
    x.offset.plus(y.offset.negated));
};
```

B.2.31.9 Литеральная форма

Литералы типа данных TS представляют собой простые календарные выражения, составленные на основе таблицы определения календаря. По умолчанию должен использоваться западный (григорианский) каледарь (таблица .36–).

Согласно настоящей спецификации, литералы календарных выражений григорианского календаря, используемого по умолчанию, соответствуют ограничению стандарта ИСО 8601, определенному в разделе 32 (всемирное время) стандарта ИСО 8824 (ASN.1) и в описании типа данных TS, приведенному в стандарте HL7 Версии 2.

Литеральные выражения являются последовательностями целых чисел, упорядоченных в соответствии со столбцом «Порядок» таблицы B.36. Периоды с меньшими порядковыми номерами записываются слева от периодов с большими порядковыми номерами. Периоды, которым порядковые номера не присвоены, не должны включаться в календарное выражение типа данных TS.

В столбце «Цифры» таблицы B.36 указано точное число цифр в числе, задающем конкретный период.

Таким образом, в соответствии с таблицей B.36 календарные выражения западного календаря начинаются с четырех цифр года (отсчет начинается с нуля), за которыми следуют две цифры месяца (отсчет начинается с 1), за ними две цифры дня месяца (начиная с 1), две цифры часа дня (начиная с нуля) и т. д. Например, строка «200004010315» является допустимым выражением для 3 ч 15 м 1 апреля 2000 г.

Опуская правые компоненты, можно получить календарные выражения разной точности.

Например, строка «20000401» является календарным выражением с точностью только до дня месяца.

Наименьший определенный период календаря (например, секунда) может быть записан как число типа REAL, в котором указаны цифры целой части секунд, за ними десятичная точка и любое число цифр дробной части.

Например, строка «20000401031520.34» означает 20.34 с 3 ч 15 м 1 апреля 2000 г.

Если когда-нибудь будут использоваться другие календари, то перед западным (григорианским) календарным выражением может быть указан префикс «GREG:», отличающий его от других календарей. Каждый календарь должен быть иметь собственный префикс. Но если префикс не указан, то по умолчанию используется западный календарь.

В современном григорианском календаре (и во всех календарях, у которых время дня основано на Всемирном координируемом времени UTC) календарное выражение может содержать суффикс часовой зоны. Он начинается со знака плюса (+) или минуса (-), за которым указаны цифры циклов часов и минут. Часовая зона UTC обозначается как смещение «+00» ог «-00»; суффикс «Z», предложенный для этой зоны в стандартах ИСО 8601 и ИСО 8824, не разрешен.

```
TS.literal ST {
  TS   : cal timestamp($1)                  { $.equal($2); }
       | timestamp(GREG)                    { $.equal($1); };

  TS timestamp(Calendar C)
       : cycles(C.head, C.epoch) zone(C)    { $.equal($1.minus($2)); }
```

```
                                        $.timezone.equal($2); }
        | cycles(C.head, C.epoch)       { $.equal($1);
                                          $.timezone.unknown; };
  Calendar cal
        : /[a-zA-Z_][a-zA-Z0-9_]*:/     { $.equal($1); };

  TS cycles(CalendarCycle c, TS t)
        : cycle(c, t) cycles(c.next, $1)  { $.equal($2); }
        | cycle(c, t) "." REAL.fractional { $.equal(c.sum($1, $3));
                                          $.precision.equal(
t.precision.plus($3.precision)); }
        | cycle(c, t)                   { $.equal($1); };

  TS cycle(CalendarCycle c, TS t)
        : /[0-9]{c.ndigits}/            { $.equal(c.sum(t, $1));
                                          $.precision.equal(
  t.precision.plus(c.ndigits)); };

  PQ zone(Calendar C)
        : "+" cycles(C.zonehead, C.epoch)  { $.equal($2.minus(C.epoch)); }
        | "-" cycles(C.zonehead, C.epoch)  { $.equal(C.epoch.minus($2)); };
};
```

B.3 Общие коллекции

Диаграмма классов типов данных общих коллекций показана на рисунке B.11.

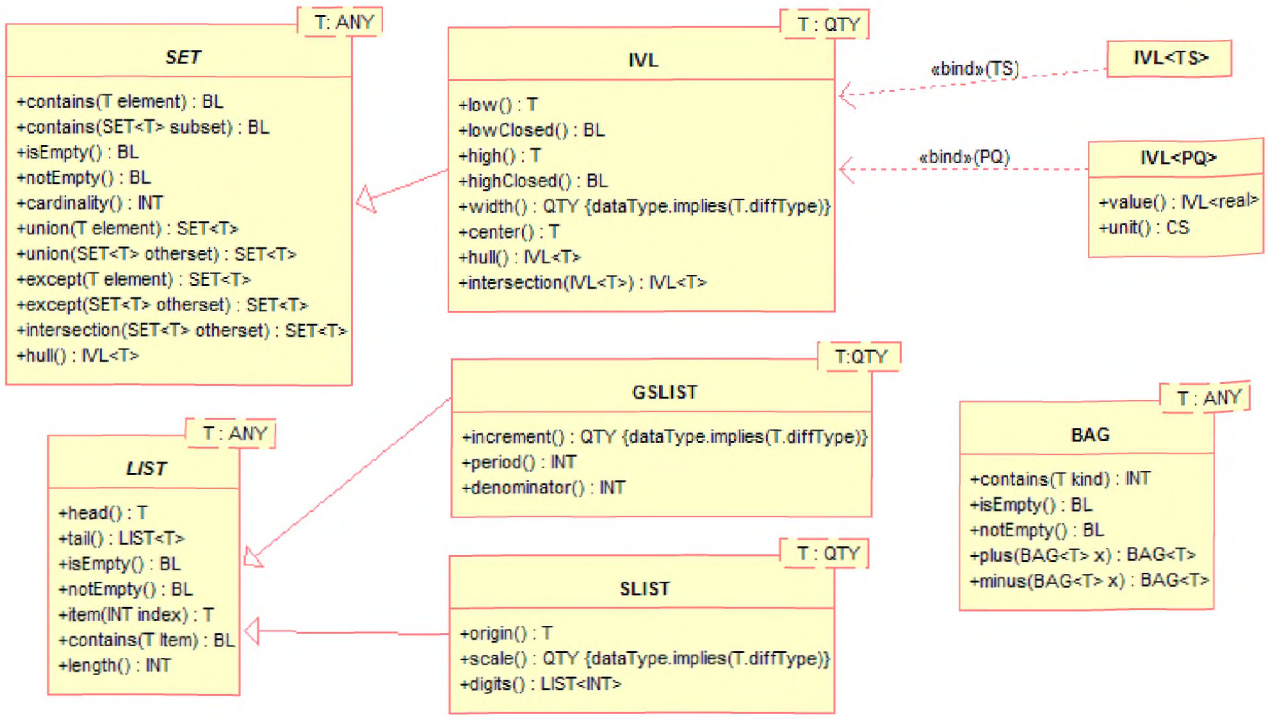


Рисунок B.11 — Типы данных общих коллекций

В этом разделе описаны типы данных, представляющие собой «коллекции» других данных, а именно множество SET, список LIST, «мешок» BAG и интервал IVL[55]. Эти типы коллекций определены как параметризованные типы данных. Понятие параметризованного типа данных описано в подразделе 0 «

Параметризованные типы данных».

B.3.1 Тип данных множества (SET) (специализация типа данных ANY)

Значение, представляющее собой неупорядоченную совокупность различных однотипных значений.

```
template<ANY T>
type Set<T> alias SET<T> specializes ANY {
            BL      contains(T element);
            BL      isEmpty;
            BL      notEmpty;
            BL      contains(SET<T> subset);
            INT     cardinality;
            SET<T>  union(SET<T> otherset);
            SET<T>  union(T element);
            SET<T>  except(T element);
            SET<T>  except(SET<T> otherset);
            SET<T>  intersection(SET<T> otherset);
 literal    ST;
 promotion  SET<T>  (T x);
            IVL<T>  hull;
};
```

B.3.1.1 Свойство contains: BL

Отношение множества к своим элементам; принимает значение «true», если данное значение является элементом множества.

Это примитивное свойство множества, с помощью которого определяются все другие свойства.

Множество может содержать только непустые различные элементы. Исключительные (пустые) значения не могут быть элементами множества.

```
invariant(SET<T> s, T n)
    where s.nonNull.and(n.isNull) {
  s.contains(n).not;
};
```

B.3.1.2 Свойство contains: BL

Отношение множества к своим подмножествам, при котором каждый элемент подмножества является также элементом этого множества.

```
invariant(SET<T> superset, subset)
    where superset.nonNull.and(subset.nonNull)
      superset.contains(subset).equal(
    forall(T element) where subset.contains(element) {
      superset.contains(element);
        });
};
```

Отсюда вытекает, что пустое множество является подмножеством каждого множества, в том числе самого себя.

---

[55] В некоторых языках программирования под типами «коллекций» понимаются контейнеры индивидуально перечисляемых типов данных, поэтому интервал (нижняя граница — верхняя граница) в них не считался бы коллекцией. Однако такая узкая интерпретация «коллекции» существенно зависит от представления/реализации. С позиции семантики типа данных не существенно, «действительно ли элемент коллекции содержится в коллекции». Поэтому элементы коллекции не обязательно являются индивидуально перечисляемыми.

B.3.1.3 Свойство notEmpty: BL

Предикат, указывающий, что данное множество содержит элементы.

```
invariant(SET<T> set)
    where set.nonNull {
  set.notEmpty.equal(exists(T element) {
    set.contains(element);
    });
};
```

B.3.1.4 Свойство isEmpty: BL

Предикат, указывающий, что данное множество не содержит элементов (отрицание свойства notEmpty). Пустое множество является допустимым значением множества, а не исключительным.

```
invariant(SET<T> set)
    where set.nonNull {
  set.isEmpty.equal(notEmpty.not);
};
```

B.3.1.5 Свойство cardinality : INT

Свойство cardinality (мощность) указывает число различных элементов множества.

```
invariant(SET<T> set)
    where set.nonNull {
  exists(T element) where set.contains(element) {
    set.cardinality.equal(set.except(element)
                .cardinality.successor);
      };
};
```

Это определение мощности не является достаточным, поскольку оно не применимо к несчетным бесконечным множествам (значения типов REAL, PQ и т. д.) и не закончено для бесконечных множеств. Кроме того, определение целочисленного типа данных, приведенное в настоящей спецификации, является неполным для таких случаев, поскольку не учитывает бесконечности. Наконец, значение мощности служит примером, когда надо различать мощность ℵ0 (алеф 0) счетных бесконечных множеств (например, значений типа INT) от мощности ℵ1 (алеф 1) несчетных множеств (например, значений типов REAL, PQ).

B.3.1.6 Свойство union: SET<T>

Свойство, описывающее объединение двух множеств (компонентов множества), при котором каждый элемент объединения является также элементом как минимум одного компонента объединения.

```
invariant(SET<T> x, y, z)
    where x.nonNull.and(y.nonNull).and(z.nonNull) {
  x.union(y).equal(z).equal(forall(T e) {
    z.contains(e).equal(x.contains(e).or(y.contains(e)));
    });
};
```

B.3.1.7 Свойство union: SET<T>

Свойство, описывающее добавление элемента к множеству.

```
invariant(SET<T> set, singletonset, T element)
    where set.nonNull.and(element.nonNull)
                .and(singletonset.cardinality.isOne)
                .and(singletonset.contains(element)) {
  set.union(element).equal(set.union(singleton));
};
```

B.3.1.8 Свойство except: SET<T>

Свойство, описывающее разность между данным множеством и вычитаемым, содержащую все элементы данного множества, которые не являются элементами вычитаемого множества.

```
invariant(SET<T> x, y, z)
    where x.nonNull.and(y.nonNull).and(z.nonNull) {
  x.except(y).equal(z).equal(forall(T e) {
    z.contains(e).equal(x.contains(e).and(y.contains(e).not));
    });
};
```

B.3.1.9 Свойство except: SET<T>

Свойство, описывающее удаление элемента из данного множества и представляющее собой множество, состоящее из всех элементов данного множества, кроме удаляемого. Если удаляемый элемент не входит в данное множество, то результирующим множеством является данное множество.

```
invariant(SET<T> x, z; T d)
    where z.nonNull.and(z.nonNull).and(d.nonNull) {
  x.except(d).equal(z).equal(forall(T e) {
    z.contains(e).equal(x.contains(e).and(d.equal(e).not));
    });
};
```

B.3.1.10 Свойство intersection: SET<T>

Свойство, описывающее пересечение двух множеств, содержащее те и только те элементы, которые входят в каждое из этих множеств.

```
invariant(SET<T> x, y, z)
    where x.nonNull.and(y.nonNull).and(z.nonNull) {
  x.intersection(y).equal(z).equal(forall(T e) {
    z.contains(e).equal(x.contains(e).and(y.contains(e)));
    });
};
```

B.3.1.11 Литеральная форма

Если элементы типа T имеют литеральную форму, то множество элементов типа T имеет литеральную форму, в которой элементы множества перечислены внутри фигурных скобок, отделяемые друг от друга точками с запятой.

```
SET<T>.literal ST {
  SET<T>        : "{" elements "}"     { $.equal($2); };
  SET<T> elements : elements ";" T{ $.except($2).equal($1); }
                  | T                  { $.contains($1);
                                         $.except($1).isEmpty; };
};
```

Примечание — Эта литеральная форма практична только для относительно малых перечисляемых множеств, но это не означает, что все множества являются относительно малым перечислением элементов.

Таблица B.38 — Примеры литералов множеств

Литерал	Значение
{1; 3; 5; 7; 19}	Множество целых или вещественных чисел
{3; 1; 5; 19; 7}	То же самое множество целых или вещественных чисел
{1.2 м; 2.67 м; 17.8 м}	Дискретное множество физических величин
{яблоко; апельсин; банан}	Множество строк символов

Примечание — Если спецификация реализуемой технологии, основанной на символьном представлении данных, имеет более естественную форму литералов таких коллекций, то она должна использовать ее для литералов множеств.

B.3.1.12 Преобразование значения элемента во множество: SET<T>

Значение типа T может быть преобразовано в тривиальное множество, содержащее это значение как единственный элемент.

```
invariant(T x) {
  ((SET<T>)x).contains(x);
  ((SET<T>)x).except(x).isEmpty;
};
```

B.3.1.13 Выпуклая оболочка полностью упорядоченных множеств: IVL<T>

Множества величин могут быть полностью упорядоченными, если между любыми двумя элементами множества определено отношение порядка. Учтите, что понятие «упорядоченного множества» не тождественно списку (тип данных LIST). Например, множество {3; 2; 4; 88; 1} является упорядоченным. Позиции элементов множества не имеют значения, но можно провести сравнение элементов и записать их в порядке возрастания (1; 2; 4; 88).

Для полностью упорядоченных множеств можно построить выпуклую оболочку. Выпуклой оболочкой полностью упорядоченного множества S является наименьший интервал, содержащий множество S. Важность этого понятия будет обсуждаться позже.

```
type Set<QTY> alias SET<QTY> {
  BL          totallyOrdered;
  IVL<T>  hull;
};

invariant(SET<QTY> s)
    where s.nonNull {
  s.totallyOrdered.equal(forall(QTY x, y)
      where s.contains(x).and(s.contains(y)) {
    x.compares(y); });
};

invariant(SET<QTY> s)
    where s.totallyOrdered {
  s.hull.contains(s);
  forall(T e)
      where s.contains(e) {
    s.hull.low.lessOrEqual(e);
    e.lessOrEqual(s.hull.high);
    };
};
```

Обратите внимание, что выпуклая оболочка определена в том и только том случае, если действительное множество является полностью упорядоченным. Значения типа данных его элементов не обязаны быть полностью упорядоченным. Например, значения типа данных PQ упорядочены только частично (поскольку сравнивать можно только величины одного типа), но множество типа SET<PQ> тем не менее может быть полностью упорядоченным (если содержит только сравнимые величины. Например, выпуклой оболочкой множества {4 s; 20 s; 55 s} является интервал [4 s; 55 s]; выпуклая оболочка множества {яблоко; апельсин; банан} не определена, поскольку это множество не является упорядоченным; выпуклая оболочка множества {2 m; 4 m; 8 s} также не определена, поскольку и оно не является полностью упорядоченным (секунды не сравнимы с метрами).

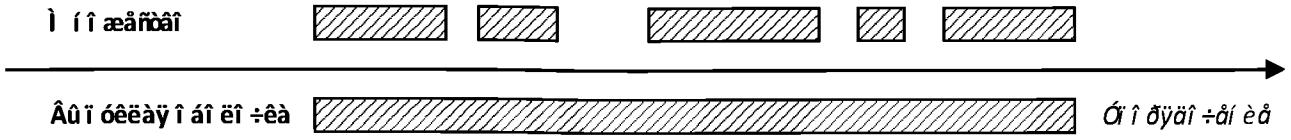


Рисунок В.12 — Выпуклая оболочка полностью упорядоченного множества

В.3.2 Тип данных списка LIST (специализация типа данных ANY)

Значение, содержащее другие дискретные (но не обязательно различные) значения в определенной последовательности.

```
template<ANY T>
type Sequence<T> alias LIST<T> specializes ANY {
            T          head;
            LIST<T>    tail;
            BL         isEmpty;
            BL         notEmpty;
            T          item(INT index);
            BL         contains(T item);
            INT        length;
  literal   ST;
  promotion LIST<T>    (T x);
};
```

Список может не содержать ни одного элемента.

В.3.2.1 Свойство head: T

Первый элемент списка. Свойство head является определяющим для семантики списка.

В.3.2.2 Свойство tail: LIST<T>

Список элементов, следующих за первым элементом списка. Свойство tail является определяющим для семантики списка.

В.3.2.3 Свойство isEmpty: BL

Предикат, имеющий значение «true», если список пуст, то есть не содержит элементов.

Обратите внимание на отличие пустого списка от пустого значения: пустой список является допустимым списком, а не исключительным значением.

```
invariant(LIST<T> x)
    where x.isEmpty {
  x.head.isNull;
  x.tail.isNull;
};
```

Обратите внимание, что пустота элемента head и списка tail является необходимым, но не достаточным условием пустоты списка, поскольку список может содержать в качестве элементов пустые значения, это условие может означать что данный список имеет единственный элемент, имеющий пустое значение.

В.3.2.4 Свойство notEmpty: BL

Предикат, имеющий значение «true», если список не является пустым. Является отрицанием свойства isEmpty.

```
invariant(LIST<T> x)
    where x.nonNull {
  x.notEmpty.equal(x.isEmpty.not);
};
```

B.3.2.5 Свойство item: T

Значением этого свойства является элемент, находящийся в заданной позиции списка (имеющий заданный индекс). Нулевой индекс задает первый элемент списка (значение свойства head).

```
invariant(LIST<T> list; INT index)
    where list.nonNull.and(index.nonNegative) {
  list.isEmpty.implies(list.item(index).isNull);
  list.notEmpty.and(index.isZero)
    .implies(list.item(index).equal(list.head));
  list.notEmpty.and(index.nonZero)
    .implies(list.item(index).equal(
      list.tail.item(index.predecessor)));
};
```

B.3.2.6 Свойство contains: BL

Предикат, имеющий значение «true», если данный список содержит заданное значение.

```
invariant(LIST<T> list; T item)
    where list.nonNull {
  list.isEmpty.implies(list.contains(item).not);
  list.nonEmpty.and(item.nonNull).implies(list.contains(item).equal(
    list.head.equal(item).or(list.tail.contains(item))));
    list.notEmpty.and(item.isNull).implies(list.contains(item).equal(
      list.head.isNull.or(list.tail.contains(item)));
};
```

B.3.2.7 Свойство length: INT

Число элементов списка. Пустые элементы включаются в подсчет наравне с обычными элементами списка.

```
invariant(LIST<T> list)
    where x.nonNull {
  list.isEmpty.equal(list.length.isZero);
  list.notEmpty.equal(list.length.equal(
    list.tail.length.successor));
};
```

B.3.2.8 Свойство equal: BL (унаследовано от типа данных ANY)

Два списка равны в том и только том случае, если либо они оба пусты, либо у них одинаковы свойства head и свойства tail.

```
invariant(LIST<T> x, y)
    where x.nonNull.and(y.nonNull) {
  x.isEmpty.and(y.isEmpty).implies(x.equal(y));
  x.notEmpty.and(y.notEmpty).and(x.head.nonNull)
    .implies(x.equal(y).equal(
      x.head.equal(y.head).and(x.tail.equal(y.tail))));
  x.notEmpty.and(y.notEmpty).and(x.head.isNull)
        .implies(x.equal(y).equal(
          y.heas.isNull.and(x.tail.equal(y.tail))));
};
```

B.3.2.9 Литеральная форма

Если элементы типа T имеют литеральную форму, то список элементов типа T имеет литеральную форму, в которой элементы списка пронумерованы и перечисляются внутри круглых скобок, разделяемые точками с запятой.

```
LIST<T>.literal ST {
  LIST<T> : "(" elements ")"          { $.equal($2); }
          | "(" ")"                   { $.isEmpty; };
  LIST<T> elements
          : T ";" elements            { $.head.equal($1);
                                        $.tail.equal($3); }
          | T                         { $.head.equal($1);
                                        $.tail.isEmpty; };
};
```

Таблица В.39 — Примеры литералов списков

Литерал	Значение
(1; 3; 5; 7; 19)	Список целых или вещественных чисел
(3; 1; 5; 19; 7)	Другой список целых или вещественных чисел
(1.2 m; 2.67 m; 17.8 m)	Дискретный список физических величин
(яблоко; апельсин; банан)	Список строк символов

Примечание — Если спецификация реализуемой технологии, основанной на символьном представлении данных, имеет более естественную форму литералов таких коллекций, то она должна использовать ее для литералов списков.

В.3.2.10 Преобразование значения элемента в список: LIST<T>

Значение типа T может быть преобразовано в тривиальный список, содержащий это значение как единственный элемент.

```
invariant(T x) {
  ((LIST<T>)x).head.equal(x);
  ((LIST<T>)x).tail.isEmpty;
};
```

В.3.3 Тип данных GeneratedSequence (GLIST) (специализация типа данных LIST)

Периодическая или монотонная последовательность значений, генерируемая с помощью небольшого числа параметров вместо перечисления. Используется для задания регулярной последовательности считывания биосигналов.

```
type GeneratedSequence<QTY T> alias GLIST specializes LIST<T> {
  T     head;
  QTY   increment;
  INT   period;
  INT   denominator;
};
```

Таблица В.40 — Сводка свойств типа данных GeneratedSequence

Имя	Тип	Описание
head	T	Первый элемент данной последовательности. Свойство head является определяющим для семантики последовательности
increment	QTY	Разность между значением и предшествующим ему отличающимся значением. Например, при генерации последовательности (1; 4; 7; 10; 13; ...) свойство increment равно 3; аналогично, при генерации последовательности (1; 1; 4; 4; 7; 7; 10; 10; 13; 13; ...) свойство increment также равно 3

Окончание таблицы B.40

Имя	Тип	Описание
period	INT	Если это свойство не пусто, то оно определяет чередование последовательности, то есть через заданное в нем число различных элементов значение элемента последовательности возвращается к начальному значению. Например, последовательность (1; 2; 3; 1; 2; 3; 1; 2; 3; ...) имеет период 3; последовательность (1; 1; 2; 2; 3; 3; 1; 1; 2; 2; 3; 3; ...) также имеет период 3
denominator	INT	Целое число, на которое делится индекс элемента последовательности. Задает число повторений того же самого значения элемента последовательности перед тем, как перейти к следующему значению. Например, при генерации последовательности (1; 1; 1; 2; 2; 2; 3; 3; 3; ...) свойство denominator равно 3

Вычисление элемента последовательности с заданным индексом i (x_i) осуществляется следующим образом: индекс *i* делится нацело на значение свойства denominator (*d*) и определяется остаток от деления результата на значение свойства period (*p*). Полученное значение умножается на значение свойства increment (Δx) и прибавляется к значению свойства head (x_0):

$$x_i = x_0 + \Delta x \times (i/d) \bmod p$$

```
invariant(GLIST<T> list, INT index)
    where list.nonNull.and(index.nonNull) {
  list.period.nonNull.implies(list.item(index).equal(
    list.head.plus(item.dividedBy(list.increment.denominator)
      .remainder(list.period)).times(increment)));
  list.period.isNull.implies(list.item(index).equal(
    list.head.plus(item.dividedBy(list.increment.denominator))
      .times(increment)));
};
```

B.3.3.1 Свойство head: T (унаследовано от типа данных LIST)

Первый элемент генерируемой последовательности.

Свойство increment: QTY

Разность между значением и предшествующим ему отличающимся значением. Например, при генерации последовательности (1; 4; 7; 10; 13; ...) свойство increment равно 3; аналогично, при генерации последовательности (1; 1; 4; 4; 7; 7; 10; 10; 13; 13; ...) свойство increment также равно 3.

```
invariant(GLIST<T> x) {
  x.increment.dataType.implies(T.diffType);
};
```

B.3.3.2 Свойство period: INT

Если это свойство не пусто, то оно определяет чередование последовательности, то есть через заданное в нем число различных элементов значение элемента последовательности возвращается к начальному значению. Например, последовательность (1; 2; 3; 1; 2; 3; 1; 2; 3; ...) имеет период 3; последовательность (1; 1; 2; 2; 3; 3; 1; 1; 2; 2; 3; 3; ...) также имеет период 3.

Задание свойства period позволяет периодически повторять группу значений. «Сигнал» такого периодического генератора всегда является «пилой» наподобие изображения линейной функции на осциллографе[56)].

B.3.3.3 Свойство denominator: INT

Целое число, на которое делится индекс элемента последовательности. Задает число повторений того же самого значения элемента последовательности перед тем, как перейти к следующему значению. Например, при генерации последовательности (1; 1; 1; 2; 2; 2; 3; 3; 3; ...) свойство denominator равно 3.

[56)] Обратите внимание на отличие типа данных GTS. Тип данных GTS представляет собой генератор для типа SET<TS>, а не LIST<TS>. Последовательность дискретных значений из непрерывного домена не имеет особого смысла кроме как для приложений оцифровки биосигналов. Однако тип данных SET<TS> можно представлять себе как последовательность значений интервалов IVL<TS>, которая тем не менее отличается от типа данных LIST<TS>.

Свойство denominator используется для генерирования нескольких последовательностей, используемых для периодического сканирования многомерного пространства. Например, (абстрактный) экран телевизора использует два таких генератора для строк и столбцов изображения. Скажем, если на экране отображается развертка 200 строк с 320 растровыми столбцами, то у генератора столбцов свойство denominator равно 1, а у генератора строк это свойство равно 320.

Таблица В.41 — Примеры генерируемых последовательностей

Свойство head	Свойство increment	Свойство denominator	Свойство period	Описание последовательности
0	1	1	∞	Последовательность идентификаторов, в которой каждый элемент равен своему индексу
198706051900	2 hour	1	∞	Последовательность значений времени, начинающаяся с 19:00 5 июня 1987 г. и каждый раз возрастающая на два часа: 21:00, 23:00, 1:00 (6 июня), 3:00, 5:00 и т. д.
0 V	1 mV	1	100	Считывание линейно возрастающего напряжения на входе цифрового осциллографа в пределах от 0 до 100 мВ за 100 шагов по 1 мВ. Частота опроса по этим данным не определяется, поскольку интервал времени между последовательными считываниями не известен
2002072920300	100 us	1	∞	Временной ряд, значения которого начинаются в 20:30 29 июня 2002 г. и возрастающие на 100 мкс на каждом шаге. Если скомбинировать его с предыдущим генератором в качестве моментов считывания сигнала, то напряжение будет меняться на 1 мВ каждые 100 мкс. Поскольку период составляет 100 шагов, то его длительность равна 10 мс, что эквивалентно частоте сигнала 100 Гц
0 В	1 mV	100	100	Сочетание этого генератора с двумя предыдущими может описывать трехмерное пространство считываний с двумя осями напряжения и осью времени. Каждое приращение напряжения данного генератора также составляет 100 мв с периодичностью 100 шагов, однако изменение напряжения происходит только через 100 считываний, поэтому первый генератор напряжения успевает сделать полный цикл за одно приращение напряжения в данном генераторе. Эти два генератора напряжения можно представить себе как «строки» и «столбцы» считываемого «кадра». Если в качестве таймера используется предыдущий генератор, то частота кадров 100 × 100 мВ составит 1 Гц

В.3.4 Тип данных SampledSequence (SLIST) (специализация типа данных LIST)

Последовательность считываемых величин, получаемая из списка целых значений с помощью масштабирования и сдвига. Используется для описания считываемых биосигналов.

```
type SampledSequence<QTY T> alias SLIST specializes LIST<T> {
  T          origin;
  QTY        scale;
  LIST<INT>  digits;
};
```

Таблица В.42 — Сводка свойств типа данных SampledSequence

Имя	Тип	Описание
origin	T	Начало шкалы значений элементов списка, то есть значение величины, которой соответствует нулевое значение элемента списка
scale	QTY	Величина, определяющая масштаб значений, указанных в списке
digits	LIST<INT>	Последовательность целых значений считываний. Обычно это исходные значения, получаемые от аналого-цифрового преобразователя

Значение физической величины, соответствующее определенному индексу (i) последовательности, вычисляется с помощью умножения значения элемента списка digits (d_i) с этим же индексом на масштаб scale (s) и прибавления к результату значения начала шкалы origin (x_o):

$$x_i = x_o + s \times d_i$$

```
invariant(SLIST<T> list, INT index)
    where list.nonNull.and(index.nonNegative) {
  list.item(index).equal(
    list.scale.times(digits.item(index))
      .plus(list.origin));
};
```

В.3.4.1 Свойство origin: T

Начало шкалы значений элементов списка, то есть значение величины, которой соответствует нулевое значение элемента списка.

В.3.4.2 Свойство scale: QTY

Величина масштаба, на которую умножаются значения элементов списка.

```
invariant(SLIST<T> x) {
  x.scale.dataType.implies(T.diffType);
};
```

В.3.4.3 Свойство digits : LIST<INT>

Последовательность целых значений считываний. Обычно это исходные значения, получаемые от аналого-цифрового преобразователя.

В.3.5 Тип данных BAG (специализация типа данных ANY)

Неупорядоченная коллекция значений, в которой каждое значение может присутствовать более одного раза (мультимножество).

```
template<ANY T>
type Bag<T> alias BAG<T> specializes ANY {
            INT      contains(T kind);
            BL       isEmpty;
            BL       notEmpty;
            BAG<T>   plus(BAG<T> x);
            BAG<T>   minus(BAG<T> x);
  promotion BAG<T>   (T x);
};
```

Примечание — значение типа BAG может быть представлено двумя способами: как перечисление элементов, включая повторяющиеся, или как «сжатое мультимножество», содержание которого представляется парами «значение элемента»-»число значений». Гистограмма, показывающая абсолютные количества, может служить примером сжатой формы представления значения типа BAG. Поэтому тип данных BAG полезен для передачи необработанных статистических данных.

В.3.5.1 Свойство contains: INT

Число элементов с данным значением, присутствующих в данном мультимножестве.

Свойство contains является примитивным семантическим свойством типа данных BAG, исходя из которого определяются все остальные свойства.

```
invariant(BAG<T> bag; T item)
    where bag.nonNull.and(item.nonNull) {
  bag.contains(item).nonNegative;
  bag.isEmpty.equal(bag.contains(item).isZero);
};
```

B.3.5.2 Свойство notEmpty: BL

Предикат, указывающий, что данное мультимножество содержит элементы.

```
invariant(BAG<T> bag)
    where bag.nonNull {
  bag.notEmpty.equal(exists(T item) {
    bag.contains(item);
    });
};
```

B.3.5.3 Свойство isEmpty: BL

Предикат, указывающий, что данное мультимножество не содержит элементов (отрицание предиката notEmpty). Пустое мультимножество является допустимым, а не исключительным значением.

```
invariant(BAG<T> bag)
    where bag.nonNull {
  bag.isEmpty.equal(notEmpty.not);
};
```

B.3.5.4 Свойство plus: BAG<T>

Мультимножество, содержащее все элементы операндов-мультимножеств, то есть количества элементов с каждым значением складываются.

```
invariant(BAG<T> x, y, z)
    where x.nonNull.and(y.nonNull) {
  x.plus(y).equal(z).equal(
    forall(T e)
        where e.nonNull {
      z.contains(e).equal(x.contains(e)
              .plus(y.contains(e)));
      });
};
```

B.3.5.5 Свойство minus: BAG<T>

Мультимножество, содержащее все элементы данного мультимножества (уменьшаемого) за вычетом элементов другого мультимножества BAG (вычитаемого). Мультимножества не могут иметь дефицит. Если вычитаемое содержит больше экземпляров одного и того же значения, то разность содержит нуль экземпляров этого значения.

```
invariant(BAG<T> x, y, z)
    where x.nonNull.and(y.nonNull) {
  x.minus(y).equal(z).equal(
    forall(T e)
        where e.nonNull {
          exists(INT n)
            where n.equal(x.contains(e).minus(y.contains(e)) {
      n.nonNegative.equal(z.contains(e));
```

```
      n.isNegative.equal(z.contains(e).isZero);
      };
    });
};
```

B.3.5.6 Преобразование значения элемента в мультимножество: BAG<T>

Значение типа T может быть преобразовано в тривиальное мультимножество, содержащее это значение как единственный элемент.

```
invariant(T x) {
  ((BAG<T>)x).contains(x).equal(1);
  forall(T y) {
    ((BAG<T>)x).contains(y)
                .implies(x.equal(y)) };
};
```

B.3.6 Тип данных IVL (специализация типа данных SET)

Этот тип данных описывает интервал, то есть множество последовательных значений упорядоченного базового типа данных.

Любой упорядоченный тип данных может быть базовым для типа данных IVL; является ли он дискретным или непрерывным, значения не имеет. Если базовый тип данных упорядочен только частично, то все элементы типа IVL должны принадлежать полностью упорядоченному подмножеству этого типа данных.

Например, тип данных PQ (физическая величина) считается упорядоченным. Однако он упорядочен только частично; полная упорядоченность определена только для сопоставимых физических величин (имеющих одну и ту же размерность). В то время как между 2 и 4 метрами существуют интервалы, нельзя задать интервал между 2 метрами и 4 секундами.

Интервалы являются множествами (тип данных SET) и обладают всеми свойствами множеств. Однако объединение и разность интервалов может не быть интервалом, поскольку значения элементов, составляющих результат этих операций, могут не быть последовательными. Пересечения интервалов всегда являются интервалами.

```
template<QTY T>
type Interval<T> alias IVL<T> specializes SET<T> {
                T       low;
                BL      lowClosed;
                T       high;
                BL      highClosed;
                QTY     width;
                T       center;
                IVL<T>  hull;
                IVL<T>  hull(IVL<T> x);
  literal       ST;
  promotion     IVL<T>  (T x);
  demotion      T;
};
```

B.3.6.1 Свойство low: T

Нижняя граница интервала.

```
invariant(IVL<T> x; T e)
    where x.nonNull.and(x.contains(e)) {
  x.low.lessOrEqual(e);
};
```

B.3.6.2 Свойство high: T

Верхняя граница интервала.

```
invariant(IVL<T> x; T e)
    where x.nonNull.and(x.contains(e)) {
  e.lessOrEqual(x.high);
};
```

B.3.6.3 Свойство width: QTY

Ширина интервала, то есть разность граничных значений high и low. Свойство width выделяется в целях симметричной обработки всех случаев, когда интервал задан не полностью. В любом представлении значения типа IVL достаточно указать только два из трех свойств high, low и width, а третье из них выводится.

Если известны две границы интервала, то ширина интервала width может быть вычислена как разность между значениями свойств high и low. Если известна одна граница и ширина width, то известна и другая граница. Если ни одна граница не известна, то значение свойства width тем не менее может быть известно. Например, известно, что некоторая деятельность занимает около 30 минут, но не известно, когда она начинается.

Учтите, что тип данных свойства width не всегда совпадает с типом данных границ интервала. Для величин, измеряемых по относительной шкале (типы данных REAL, PQ, MO), он совпадает. Для величин, измеряемых по разностной шкале (например, тип данных TS), свойство width имеет тип данных разности (например, тип данных PQ с размерностью времени в случае интервала с границами типа TS). Для дискретных элементов (тип данных INT) ширина width определяется как число элементов в интервале, деленное на 2. Поэтому она может иметь тип данных REAL.

```
invariant(IVL<T> x) {
  x.low.lessOrEqual(x.high);
  x.width.equal(x.high.minus(x.low));
};

invariant(IVL<T> x) {
  x.width.dataType.implies(T.diffType);
};
```

B.3.6.4 Свойство center: T

Это свойство задает центр интервала, то есть арифметическое среднее его границ (сумма значений свойств low и high, деленная на 2). Свойство center выделяется как семантическое для преобразования интервалов в значения точек и обратно.

Учтите, что свойство center существует не у каждого интервала. Например, интервалы, не ограниченные с одной стороны, не имеют центра. То же относится и к интервалам дискретных значений, содержащим четное число элементов. Если одна или обе границы интервала не известны, свойство center тем не менее может быть задано. В действительности это свойство в основном предназначено для использования, когда границы интервала не известны.

```
invariant(IVL<T> x)
    where x.low.nonNull.and(x.high.nonNull) {
  x.center.equal(x.low.plus(x.width.times(0.5))));
};

invariant(IVL<T> x)
    where x.low.isNull.or(x.high.isNull) {
  x.center.notApplicable;
};
```

B.3.6.5 Свойство lowClosed: BL

Указывает, включается ли нижняя граница low в значение типа IVL (закрытая граница) или исключается из него (открытая граница).

```
invariant(IVL<T> x)
    where x.nonNull {
  x.low.nonNull.implies(x.lowClosed.equal(x.contains(x.low)));
  x.low.isNull.implies(x.lowClosed.not);
};
```

B.3.6.6 Свойство highClosed: BL

Указывает, включается ли верхняя граница high в значение типа IVL (закрытая граница) или исключается из него (открытая граница).

```
invariant(IVL<T> x)
    where x.nonNull {
  x.high.nonNull.implies(x.highClosed.equal(x.contains(x.high)));
  x.high.isNull.implies(x.highClosed.not);
};
```

B.3.6.7 Литеральная форма

Литеральные формы типа данных IVL определены таким образом, чтобы они по возможности были интуитивно очевидными для человека. Определены следующие пять различных форм[57]:

- форма интервала с квадратными скобками, например, «[3.5; 5.5[«;
- форма с дефисом, например, «3.5-5.5»;
- форма с оператором сравнения, используя символы операторов отношения, например, «<5.5»;
- форма с центром и шириной, например, «4.5[2.0[«;
- форма только с шириной и квадратными скобками, например, «[2.0[«.

```
IVL<T>.literal ST {
  IVL<T> range   : interval              { $.equal($1); }
                 | dash                  { $.equal($1); }
                 | comparator            { $.equal($1); }
                 | center_width          { $.equal($1); }
                 | width                 { $.equal($1); };

  IVL<T> interval
                 : open T ";" T close;   { $.low.equal($2);
                                           $.high.equal($4);
                                           $.lowClosed.equal($1);
                                           $.highClosed.equal($5); };
  BL open        : "["                   { $.equal(true); }
                 | "]"                   { $.equal(false); };
  BL close       : "]"                   { $.equal(true); }
                 | "["                   { $.equal(false); };
  IVL<T> width
                 : open T.diffType close { $.width.equal($2);
                                           $.lowClosed.equal($1);
                                           $.highClosed.equal($3); };
  IVL<T> center_width
                 : T width               { $.center.equal($1);
                                           $.width.equal($2.width);
                                           $.lowClosed.equal($2.lowClosed);
```

[57] Наличие столь большого числа вариантов заслуживает разъяснений. В принципе достаточно было бы иметь форму интервала и форму только с шириной. Однако форма интервала воспринимается чужеродной во многих приложениях медицинской информатики. Одной из важных целей литеральных форм является искоренение несоответствия за счет облегчения соответствия, не входящего в противоречие со значением понятий.

Далее, различные литеральные формы имеют свою силу и свою слабость. Сила формы интервала и формы с центром и шириной в том, что они более точны и позволяют указать открытость и закрытость границ. Но их слабость в том, что для обозначения бесконечных границ требуются специальные символы, что не требуется для формы с оператором сравнения. Форма с центром и шириной вообще не позволяет задавать интервалы с бесконечными границами. Но форма с оператором сравнения может представлять только интервалы с одной границей (то есть когда другая граница бесконечна или не известна). Форма с дефисом слабее всех, но при этом наиболее интуитивна для интервалов с двумя границами.

```
                                    $.highClosed.equal($2.highClosed); };
  IVL<T> dash     : T "-" T;        { $.low.equal($2);
                                    $.high.equal($4);
                                    $.lowClosed.equal(true);
                                    $.highClosed.equal(true); };
  IVL<TS> comparator
                  : "<"  T          { $.high.equal(T);
                                    $.high.closed(false);
                                    $.low.negativelyInfinite; }
                  | ">"  T          { $.low.equal(T);
                                    $.low.closed(false);
                                    $.high.positivelyInfinite; }
                  | "<=" T          { $.high.equal(T);
                                    $.high.closed(true);
                                    $.low.negativelyInfinite; }
                  | ">=" T          { $.low.equal(T);
                                    $.low.closed(true);
                                    $.high.positivelyInfinite; };
      };
```

B.3.6.8 Преобразование значений элементов в интервалы: IVL<T>

Величина типа T может быть преобразована в тривиальный интервал типа IVL, у которого свойства low и high равны и границы закрыты.

```
invariant(T x) {
  ((IVL<T>)x).low.equal(x);
  ((IVL<T>)x).high.equal(x);
  ((IVL<T>)x).highClosed;
  ((IVL<T>)x).lowClosed;
};
```

B.3.6.9 Приведение интервала к представительному значению элемента типа T

Значение типа IVL может быть приведено к одной величине типа T, представляющей весь интервал. Если обе границы финитны, этой величиной является значение свойства center. Если одна из границ бесконечна, этой величиной является другая граница. Если обе границы бесконечны, приведение к одной точке не применимо.

```
invariant(IVL<T> x)
    where x.nonNull {
  x.low.nonNull.and(x.high.nonNull).implies(((T)x).equal(x.center));
  x.high.nonNull.and(x.low.isNull).implies(((T)x).equal(x.high));
  x.low.nonNull.and(x.high.isNull).implies(((T)x).equal(x.low));
  x.low.isNull.and(x.high.isNull).implies(((T)x).notApplicable);
};
```

B.3.6.10 Свойство hull: IVL<T> (унаследовано от типа данных SET)

Это свойство задает выпуклую, или «интервальную» оболочку двух значений типа IVL, представляющую наименьший интервал, содержащий оба операнда.

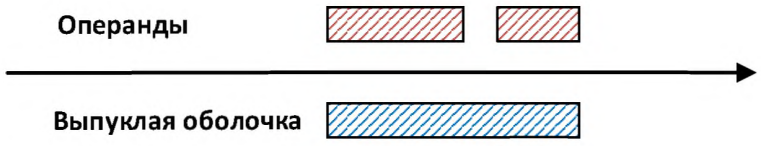


Рисунок B.13 — Выпуклая оболочка двух интервалов

```
invariant(IVL<T> h, IVL<T> i, j)
    where h.equal(i.hull(j)) {
  i.low.lessOrEqual(j.low).implies(h.low.equal(i.low));
  j.low.lessOrEqual(i.low).implies(h.low.equal(j.low));
  i.high.lessOrEqual(j.high).implies(h.high.equal(j.high));
  j.high.lessOrEqual(i.high).implies(h.high.equal(i.high));
};
```

B.3.7 Тип данных физических величин IVL<PQ> (специализация типа данных IVL)

Множество последовательных значений физических величин.

Тип данных интервала физических величин строится по общему типу интервала. Но с учетом того, что единицы измерения могут быть взяты от границ интервала, в определение типа данных IVL<PQ> добавлены новая семантика и отдельная литеральная форма. Дополнительным представлением интервала физических величин является интервал вещественных чисел с одной единицей измерения.

```
type Interval<PQ> alias IVL<PQ> {
  IVL<REAL> value;
  CS  unit;
};
```

Единица измерения применяется как к нижней, так и верхней границе интервала.

```
invariant(IVL<PQ> x)
    where x.nonNull {
  x.value.nonNull;
  x.low.value.equal(x.value.low);
  x.low.unit.equal(x.unit);
  x.lowClosed.equal(x.value.lowClosed);
  x.high.value.equal(x.value.high);
  x.high.unit.equal(x.unit);
  x.highClosed.equal(x.value.highClosed);
};
```

Специальная литеральная форма представляет собой просто интервал вещественных чисел, пробел и единицу измерения.

```
IVL<PQ>.literal ST {
  IVL<PQ>  : IVL<REAL> " " unit   { $.value($1);
                                    $.unit.equal($3); }
           | IVL<REAL>            { $.equal($1); };
  CS unit  : ST                   { $.value.equal($1);
 $.codeSystem(2.16.840.1.113883.3.2); };
};
```

Например, строки «[0;5] mmol/L « или «<20 mg/dL» являются допустимыми литеральными формами интервалов физических величин. Допускается также общая литеральная форма, например, «[50 nm; 2 m]».

B.3.8 Тип данных интервала времени IVL<TS> (специализация типа данных IVL)

Множество последовательных значений штампов даты и времени.

Общий тип данных интервала позволяет также определить интервалы моментов времени. Однако для этих интервалов имеются определенные особенности литеральных представлений и преобразования, описанные в настоящем подразделе.

```
type Interval<TS> alias IVL<TS> {
  literal   ST;
  promotion  IVL<TS> (TS x);
};
```

В.3.8.1 Преобразование моментов времени в интервалы: IVL<TS> (унаследовано от типа данных IVL)

Значение типа TS может быть преобразовано в интервал типа IVL<TS>, у которого нижняя граница представляет собой само это значение, а свойство ширины width выводится из точности значения типа TS и длительности наименее значащего календарного периода, указанного в этом значении. Верхняя граница интервала открыта. Например, литерал «200009» значения типа TS преобразуется в интервал типа IVL<TS> с нижней границей 200009 и шириной 30 дней, что эквивалентно интервалу «[200009;200010[«.

В.3.8.1 Литеральная форма

Литеральная форма интервала моментов времени является исключительной:

- форма с дефисом не разрешена для интервалов моментов времени;
- вместо нее используется «форма оболочки».

Чтобы избежать синтаксических конфликтов с часовой зоной и несколько отличающихся профилей, предусмотренных в стандарте ИСО 8601 и используемых в некоторых платформах реализуемых технологий, форма с дефисом для значений типа IVL<TS> не разрешена. Предпочтительной является форма интервала с квадратными скобками.

Пример — Литералом интервала от 20:00 до 21:40 12 мая 1987 г. служит строка «[198705122000;198705122130]».

Примечание — Точность границы интервала не существенна для самого интервала. Можно было бы ошибочно предположить, что интервал «[19870901;19870930]» охватывает весь сентябрь 1987 года вплоть до завершения суток 30 сентября. Однако это не так! Правильный способ задания полного календарного цикла (например, час, день, месяц, год и т. д.) состоит в использовании нотации с открытой верхней границей. Например, весь сентябрь 1987 г. может быть задан литералом «[198709;198710[«.[58)]

«Форма оболочки» литерала определяется как выпуклая оболочка (см. В.3.6.10 «Свойство hull: IVL<T> (унаследовано от типа данных SET)») интервалов, полученных в результате преобразования двух значений моментов времени в интервалы.

```
IVL<TS> hull : TS ".." TS    { $.equal(((IVL<TS>)$1)
                                 .hull((IVL<TS>)$3));
};
```

Например, строка «19870901..19870930» является допустимым литералом в «форме оболочки». Она эквивалентна форме интервала «[19870901;19871001[«.[59)]

Форма оболочки позволяет сокращение, при котором в литерале верхней границы не требуется повторять левые цифры, одинаковые с литералом нижней границы. Два литерала моментов времени выравниваются вправо и недостающие левые цифры копируются из литерала нижней границы в литерал верхней. Эта простая строковая операция здесь формально не определяется.

Например, интервал с 12 мая 1987 г. по 23 мая 1987 г. может быть записан как «19870512..23». Однако интервал с 12 мая 1987 г. по 2 июня 1987 г. должен быть записан как «19870512..0602», а не «20000512..02».

В.4 Параметризованные расширения типов данных

Параметризованные расширения типов данных представляют собой параметризованные типы данных, у которых в качестве параметра используется один тип данных и которые расширяют (специализируют) этот параметр. На формальном языке определения типов данных параметризованные расширения типов данных могут быть описаны следующим шаблоном: «**template**<ANY T> **type***GenericTypeExtensionName***specializes** T { ... };». Эти расширения наследуют большинство свойств своего базового типа данных и добавляют к ним некоторые специфичные свойства. Поскольку параметризованное расширение типа данных является специализацией своего базового типа, то значение расширения типа данных может быть использовано в качестве значения своего базового типа данных.

Примечание — Значения расширенных типов могут быть подставлены вместо значений свого базового типа. Однако в спецификации реализуемой технологии могут быть наложены некоторые ограничения на то, какие расширения являются приемлемыми. В частности, расширения не должны определяться для тех компонентов, которые содержат значения свойств типов данных. Таким образом, хотя любые значения данных могут быть ан-

---

[58)] Может возникнуть впечатление, что это утверждение противоречит правилу преобразования значения типа TS в интервал типа IVL<TS>. Однако никакого противоречия нет. Точность границы значения не имеет, но точность отдельного момента времени (в отличие от границы интервала) существенна, когда момент времени преобразуется в интервал.

[59)] Форма оболочки может показаться излишней для представления простого интервала. Однако она становится важной для нотации периодических интервалов, поскольку сокращает запись и (хотя это и может быть спорным) делает запись более сложных структур периодов времени более интуитивной.

нотированы вне спецификации типа данных, спецификация реализуемой технологии может не предусматривать способ аннотирования свойства типа данных. В настоящее время комитет HL7 не позволяет использовать параметризованные расширения типов данных за исключением тех ситуаций, когда такие расширения явно разрешены (в настоящей спецификации или в других спецификациях комитета HL7) для тех вариантов использования, где важна такая сложная функциональность. В этих случаях экземпляры указанных параметризованных расширений типов данных должны быть специально и явно отражены в моделях HL7 RIM, MIM, RMIM и HMD (если применимы) в результате утверждения техническим комитетом[60)].

B.4.1 Тип данных компонента истории HXIT (специализация типа данных T)

Параметризованный тип данных, добавляющий интервал времени к любому значению любого типа данных. Этот интервал указывает время, в тесение которого информация, представленная данным значением, действительна или была действительна.

Если базовый тип данных T не обладает свойством validTime (время действительности), то тип данных HXIT добавляет эту функциональность к базовому типу. Если же базовый тип данных T обладает свойством времени действительности (в настоящее время это свойство имеется только у типа данных EN), то это свойство отображается на свойство времени действительности, предусмотренное в типе данных HXIT[61)].

```
template<ANY T>
type HistoryItem<T> alias HXIT<T> specializes T {
  IVL<TS> validTime;
};
```

B.4.1.1 Свойство validTime: IVL<TS>

Интервал времени, в течение которого данная информация была действительной, либо является действительной, либо ожидается действительной. Любой конец интервала может быть открытым или закрытым, бесконечным или неопределенным.

B.4.2 Тип данных истории HIST (специализация типа данных SET<HXIT>)

Множество значений данных, имеющих свойство времени действительности и тем самым соответствующим типу данных HXIT. Информация об истории не ограничена прошлым, в ней могут присутствовать ожидаемые будущие значения.

Тип данных HIST предназначен для хранения действительных исторических (и будущих) значений элемента, а не для журнализации значений, которые конкретная система присваивала этому элементу.

```
template<ANY T>
type History<T> alias HIST<T> specializes SET<HXIT<T>> {
        HXIT<T>  earliest;
        HIST     exceptEarliest;
        HXIT<T>  latest;
        HIST     exceptLatest;
 demotion        HXIT<T>;
};
```

Семантика этого типа данных не содержит принципиального запрета на пересечение интервалов времени действительности. Но если у двух элементов истории границы IVL.low и IVL.high интервалов действительности совпадают, то нет возможности утверждать, который их них должен считаться более ранним, а какой — более поздним.

[60)] Настоящая спецификация накладывает на себя определенные ограничения, обеспечивающие существующим системам удобный переход на нее. Однако в формальной спецификации параметризованные расширения сохранены как подставляемые вместо своих базовых типов. В будущем это самоограничение может быть отменено. В новых разработках рекомендуется предусмотреть некоторую обобщенную поддержку этих параметризованных расширений типов данных.

[61)] Учтите, что типы данных являются спецификациями абстрактных свойств значений. Настоящая спецификация не указывает, как эти значения должны быть представлены в спецификации реализуемой технологии или реализованы в прикладной программе. В частности, она не указывает, как представленные компоненты должны быть поименованы или в каком порядке они должны следовать. Кроме того, семантическая иерархия генерализации может отличаться от иерархии классов, выбранной для реализации (если технология реализации обеспечивает наследование). Не забывайте о различии между типом (интерфейсом) и реализацией (конкретной структурой данных, классом). Спецификация реализуемой технологии должна содержать отображение свойств любого определенного в ней типа данных на семантические свойства, определенные в настоящем документе.

```
invariant(HIST x)
    where x.nonNull {
  x.notEmpty;
  ((T)x).equal(x.latest);
};
```

B.4.2.1 Свойство earliest: HXIT<T>

Элемент множества интервалов действительности, у которого граница IVL.low (время начала действительности) меньше или равна (то есть более ранняя), чем у любого другого элемента истории из этого множества.

```
invariant(HIST x; HXIT<T> e)
    where x.contains(e) {
  x.earliest.validTime.low.lessOrEqual(e.validTime.low);
};
```

B.4.2.2 Свойство latest: HXIT<T>

Элемент множества интервалов действительности, у которого граница IVL.high (время начала действительности) меньше или равна (то есть более поздняя), чем у любого другого элемента истории из этого множества.

```
invariant(HIST x; HXIT<T> e)
    where x.contains(e) {
  x.latest.validTime.high.greaterOrEqual(e.validTime.high);
};
```

B.4.2.3 Свойство exceptEarlies: HIST<T>

Производная история, из которой исключен самый ранний элемент.

```
invariant(HIST x)
    where x.nonNull {
  x.exceptEarliest.equal(x.except(x.earliest));
};
```

B.4.2.4 Свойство exceptLatest: HIST<T>

Производная история, из которой исключен самый поздний элемент.

```
invariant(HIST x)
    where x.nonNull {
  x.exceptLatest.equal(x.except(x.latest));
};
```

B.4.2.5 Приведение истории к одному компоненту: HXIT<T>

Преобразование, при котором значение типа HIST преобразуется в значение типа HXIT. Это преобразование выбирает самый поздний элемент истории.

Такое преобразование полезно в тех случаях, когда поставщик информации передает не одно значение, а историю изменения значений, а потребитель информации, ожидающий получении не истории, а единственного значения, выбирает из нее более позднее.

Учтите, что согласно определению тип данных HXIT семантически специализирует тип данных T. Это означает, что потребитель информации, ожидающий значение типа T, но получивший значение типа HXIT, не увидит для себя никакой разницы (подстановка специализации).

B.4.3 Тип данных вероятностного неопределенного значения UVP (специализация типа данных T)

Параметризованное расширение типа данных, используемое для указания вероятности, которую, по мнению поставщика информации, имеет данное значение.

Способ вычисления вероятности не входит в область применения настоящей спецификации.

Вероятности субъективны и (как и любое значение данных) должны интерпретироваться в индивидуальном контексте, например, если появилась новая информация, то значение вероятности может измениться. Таким образом, в любом сообщении (в документе или в ином представлении информации) информация, и в частности вероятности, отражает то, что поставщик информации полагал адекватным цели сообщения (документа) в момент создания сообщения (документа).

Например, в начале бейсбольного сезона 2000 года (май), в Лас Вегасе ставки на то, что команда Нью-Йорк Янкиз выиграет Мировую серию, могли приниматься букмекерами в отношении 1 к 10 (то есть с вероятностью 0.100). Во время написания настоящей спецификации команды Нью-Йорк Янкиз и Нью-Йорк Метс выиграли в своих подгруппах, и должна была начаться Мировая серия. Очевидно, что в этот момент вероятность, что Нью-Йорк Янкиз выиграют Мировую серию, стала ощутимо больше, скажем, 6 к 10 (вероятность 0.600). Все в мире зависти от контекста, в частности, от его даты.

Поскольку вероятности являются субъективными мерами веры, то о них надо говорить не как о «правильных» или «неправильных», а как о «точных» или «неточных». Примечательно, что на проведение экспериментов по частоте появления некоторого результата, позволяющих оценить его вероятность, нельзя рассчитывать. Действительно, какие бы утверждения о людях и событиях не делались, невозможно подтвердить их вероятность экспериментами «частоты» соответствующих событий.

В условиях предыдущего примера букмекеры Лас Вегаса не могут настаивать, чтобы Янкиз и Метс провели 1000 пробных игр до начала Мировой серии. Даже если бы это было возможно, эти игры не носили бы столь ожесточенный характер, как реальная Мировая серия, и вероятность бы не была достаточно точной. Поэтому букмекеры должны оценивать вероятность на основании предыдущих игр, статистики игроков, сведений о травмах и т. д.

```
template<ANY T>
type UncertainValueProbabilistic<T> alias UVP<T> specializes T {
  REAL probability;
};
```

Формальные ограничения на тип данных T не накладываются. Теоритически дискретные вероятности могут быть приписаны только дискретным значениям данных. Поэтому параметризованный тип данных UVP не должен использоваться для базовых типов данных REAL, PQ или MO.

B.4.3.1 Свойство probability: REAL

Вероятность, приписанная значению и представляющая собой десятичное значение между 0 (невероятно) и 1 (определенно), включительно.

```
invariant(UVP<T> x)
    where x.nonNull.and(x.probability.nonNull) {
  ((IVL<REAL>)[0;1]).contains(x.probability);
};
```

Значение вероятности «по умолчанию», которое можно было бы использовать, если вероятность на задана, не существует. Поэтому невозможно провести семантическое различие между значением типа UVP, вероятность которого не указана, и значением типа T. Использования типа данных UVP не означает «неопределенность», а использование простого типа данных T не означает «определенность». Действительно, вероятность значения типа UVP может составлять 0.999 или 1, то есть это значение имеет высокую степень определенности, в то время как значение типа T может быть очень смутной догадкой.

B.4.4 Тип данных непараметрического распределения вероятности (NPPD) (специализация типа данных SET<UVP>)

Множество значений типа UVP с вероятностями (известное как гистограмма). Все элементы множества рассматриваются как альтернативные и ранжируются по своей вероятности, выражающей степень веры (или частоту) каждого из этих значений.

Тип данных NPPD в основном предназначен для обеспечения статистических отчетов по измерениям, полученным от многих субъектов и консолидированных в гистограмме. Подобные отчеты встречаются в эпидемиологии, ветеринарии, лабораторном деле, а также в контроле цены и конструировании деловых процессов.

Семантически информация об объявленном значении существует по контрасту с дополнительным множеством не объявленных возможных значений. Таким образом, семантически значение типа NPPD содержит все возможные значения, каждому из которых приписана определенная вероятность.

Наиболее простым способом визуализации является гистограмма наподобие показанной на рисунке .14–.

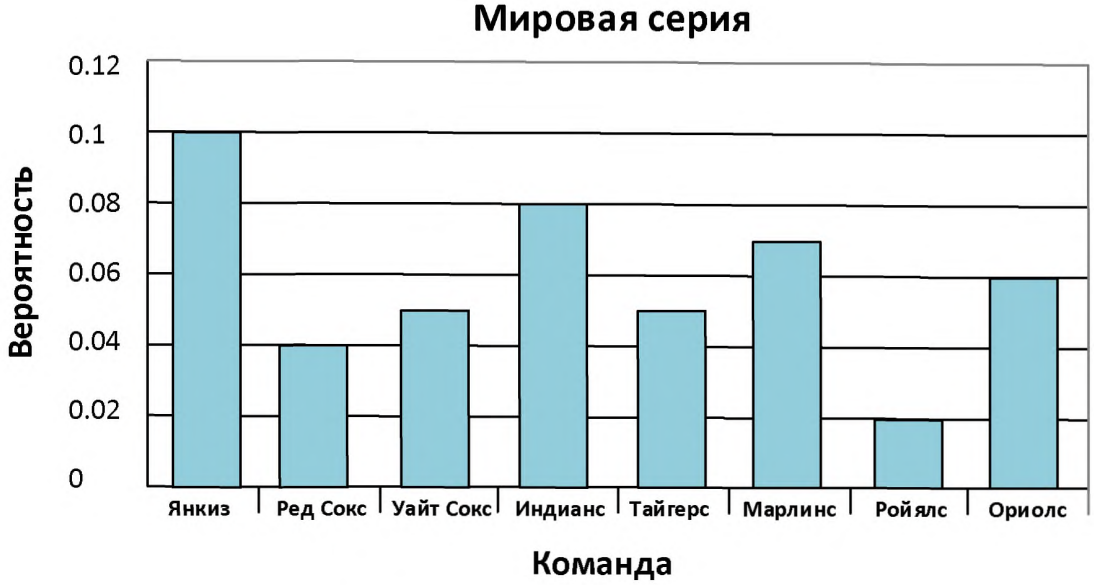


Рисунок В.14 — Пример гистограммы

Этот пример иллюстрирует вероятность выигрыша Мировой серии избранными бейсбольными командами основной лиги (до старта сезона). Команды взаимно исключительны, и если бы на диаграмме были показаны все команды, то сумма вероятностей была бы равна 1 (то есть определенно какая-нибудь команда выиграет).

Примечание — Хотя семантически в типе данных NPPD вероятности присвоены всем возможным значениям, не все значения должны быть представлены явно. Вероятности тех значений, которые не были упомянуты, будут равны равномерному распределению остатку вероятности. Например, если множество значений имеет элементы {A; B; C; D}, но в описании типа данных NPPD указаны только значения {(B; 0.5); (C; 0.25)}, то остаток вероятности составит 1 - 0.75 = 0.25 и будет распределен равномерно по элементам дополняющего множества: {(A; 0.125); (D; 0.125)}. Семантически тип данных NPPD представляет собой объединение объявленного распределения вероятности и необъявленного дополнения с равномерным распределением остатка вероятности.

```
template<ANY T>
type NonParametricProbabilityDistribution<T> alias NPPD<T>
    specializes SET<UVP<T>> {
  SET<UVP<T>> mostLikely(INT n);
};
```

Как и в случае типа данных UVP, на тип данных T не накладываются формальные ограничения несмотря на то, что не все типы данных разумно использовать. Обычно тип данных NPPD используется для неупорядоченных базовых типов T, если только «небольшому» множеству его значений явно приписаны вероятности, или если распределение вероятности не может (или не должно) описываться параметрическим методами. В других случаях предпочтительнее использовать тип данных PPD.

В.4.4.1 Свойство mostLikely: UVP (наиболее вероятное значение)

```
invariant(NPPD<T> x)
    where x.nonNull {
  x.notEmpty;
  x.contains(x.mostLikely(n));
  x.mostLikely(n).
  forall(UVP<T> d, e; SET<UVP<T>> m; INT n)
      where x.contains(d).and(m.equal(x.mostLikely(n)))
              .and(m.contains(e)) {
    e.greaterOrEqual(d).or(m.contains(d));
    };
};
```

B.5 Спецификации моментов и интервалов времени

Обзор типов данных спецификации моментов и интервалов времени показан на рисунке B.15.

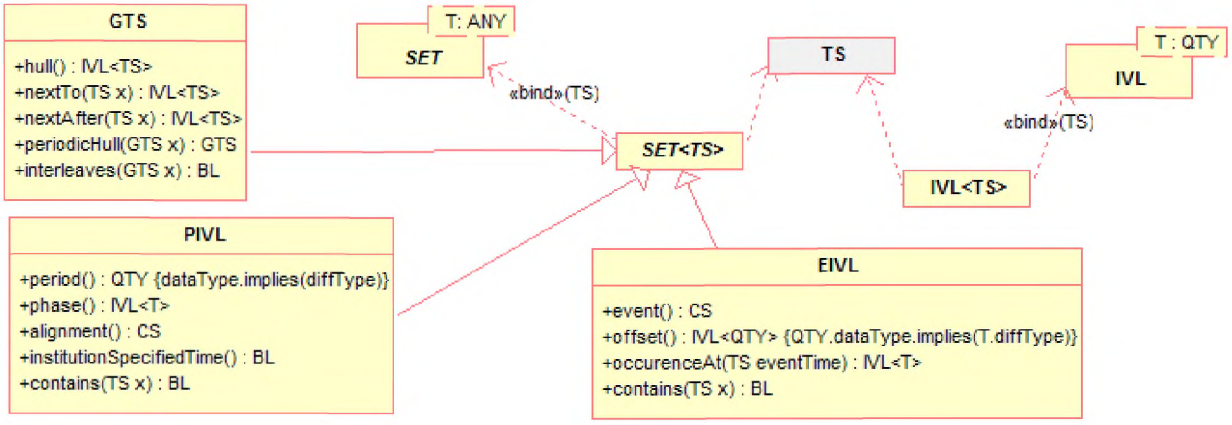


Рисунок B.15 — Обзор типов данных спецификации моментов и интервалов времени

Комплекс спецификаций моментов и интервалов времени используется для указания сложных сроков событий и действий, которые, к примеру, случаются в системах управления заказами и ведения расписаний. Он также поддерживает циклические шаблоны действительности, которые могут применяться к определенным видам информации, например, к телефонным номерам (вечерним, дневным), часам работы, или к адресам так называемых «перелетных птиц» (людям, которые предпочитают зимой быть поближе к экватору, а летом подальше от него).)

Спецификации моментов и интервалов времени включают в себя момент времени (тип данных TS), интервал времени (тип данных IVL<TS>), а также дополнительные типы данных, специально сконструированные для повторяющихся расписаний. К ним относятся типы данных PIVL, EIVL и, наконец, собственно тип данных GTS. Все эти типы данных описывают распределение моментов времени повторения состояний или событий.

B.5.1 Тип данных периодического интервала времени PIVL (специализация типа данных SET)

Определение. Периодически повторяющийся интервал времени. Тип данных PIVL имеет два свойства: phase (фаза) и period (период). Свойство phase задает «прототип интервала», повторяющийся с периодичностью, указанной в свойстве period.

Таблица B.43 — Сводка свойств типа данных PIVL

Имя	Тип	Описание
phase	IVL<T>	Прототип повторяющегося интервала, задающий длительность каждого повторения и привязку последовательности компонентов типа данных PIVL к определенному моменту времени
period	T.diff	Длительность, указанная как обратная величина частоты повторения компонентов типа данных PIVL
alignment	CS	Указывает, должны ли повторения привязываться к календарным циклам, и если да, то каким образом (например, чтобы отличать «каждые 30 дней» от «5-го числа каждого месяца). Не привязанное значение типа PIVL описывает повторения независимо от календаря. Привязанное значение типа PIVL синхронизируется с календарем
institution SpecifiedTime	BL	Указывает, оставляется ли точное время события на усмотрение стороны, которая ведет расписание (например, чтобы отличать «каждые 8 часов» от «3 раза в день»)

Например, расписание «каждые 8 ч в течение 2 мин» можно представить как значение типа PIVL, у которого интервал IVL.width равен двум минутам, а период повторения period равен 8 ч.

Свойство phase также обозначает привязку все серии периодически повторяющихся интервалов к определенному моменту времени. Повторения, описываемые типом данных PIVL, не имеют ни начала, ни конца. Они бесконечны как в будущем, так и в прошлом.

```
template<TS T>
protected type PeriodicInterval<T> alias PIVL<T>
          specializes  SET<T> {
          T.diff       period;
          IVL<T>       phase;
          CS           alignment;
          BL           institutionSpecifiedTime;
          BL           contains(TS);
  literal ST;
};
```

Тип данных PIVL полностью определен, если полностью определены его свойства period и phase. Интервал может быть определен только частично, если указаны либо только его ширина IVL.width, либо только одна из его границ.

Например, расписание «каждые 8 ч в течение 2 мин» задает только период period и ширину IVL.width фазы phase, но не границу фазы phase. Напротив, расписание «каждые 8 ч, начиная с 4 ч» задает только период period и нижнюю границу IVL.low фазы phase, но не ее верхнюю границу IVL.high. Расписание «каждые 8 ч в течение 2 мин, начиная с 4 ч» полностью определено, поскольку заданы как период period, так и оба значения IVL.low и IVL. width для фазы phase.

Тип данных PIVL представляет собой параметризованное расширение типа данных, у которого параметр типа T ограничен типом данных TS и его расширениями. Поскольку тип данных PPD<TS>> представляет собой расширение типа данных TS, то он может использоваться для формирования значений типа PIVL<PPD<TS>>.

Нередко моменты времени и повторяющиеся расписания определены только приблизительно. Например, расписание «три раза в день в течение 10 мин» обычно не означает, что свойство period в точности равно 8 ч, а интервал в точности 10 мин. На практике дистанция между событиями такого расписания может варьироваться от 3 до 12 ч, а ширина IVL.width интервала может быт меньше 5 мин или больше 15 мин. Для указания степени свободы или «критичности ко времени» подобного расписания может использоваться тип данных PIVL<PPD<TS>>.

B.5.1.1 Свойство phase : IVL<T>

Прототип повторяющегося интервала, задающий длительность каждого повторения и привязку последовательности компонентов типа данных PIVL к определенному моменту времени.

Свойство phase также обозначает привязку все серии периодически повторяющихся интервалов к определенному моменту времени. Повторения, описываемые типом данных PIVL, не имеют ни начала, ни конца. Они бесконечны как в будущем, так и в прошлом. Значение компонента IVL.width фазы phase должно быть меньше значения свойства period или равно ему.

```
invariant (PIVL<T> x)
    where x.nonNull {
  x.phase.nonNull.implies(x.phase.width.lessOrEqual(x.period));
};
```

B.5.1.2 Свойство period: T.diff

Длительность, указанная как обратная величина частоты повторения компонентов типа данных PIVL.

Свойство period имеет тип данных QTY с размерностью времени (T.diff). Для неопределенных значений типа PIVL свойство period представляет собой распределение вероятности прошедшего времени.

```
invariant(PIVL<T> x)
    where x.nonNull {
  x.period.nonNull;
};

invariant(PIVL x) {
  x.period.dataType.implies(PQ);
};
```

B.5.1.3 Свойство alignment: CS

Указывает, должны ли повторения привязываться к календарным циклам, и если да, то каким образом (например, чтобы отличать «каждые 30 дней» от «5-го числа каждого месяца). Не привязанное значение типа PIVL описывает повторения независимо от календаря. Привязанное значение типа PIVL синхронизируется с календарем.

Например, расписание «5-го числа каждого месяца» представляет собой значение типа PIVL, привязанное к календарю. Значение свойства period варьируется от 28 до 31 дней в зависимости от календарного месяца. Напротив, расписание «каждые 30 дней» описывает не зависящее о календаря свойство period, которое каждый месяц приходится на разный день.

При привязке значения типа PIVL к календарю должен быть указан календарный цикл, к которому осуществляется привязка. Затем сплошной поток времени подразделяется на календарные циклы. Такое подразделение называется календарной «решеткой», генерируемой привязываемым календарным циклом. В каждом подразделе границы очередного интервала отстоят на одно и то же время от начальной момента каждого подраздела. Другими словами, промежуток времени от следующей нижней линии решетки до начала интервала постоянен.

Например, в расписании «5-е число каждого месяца» привязка осуществляется к месяцу года (MY). Сплошной поток времени подразделяется на месяцы года. Промежуток времени от начала каждого месяца до начала очередного интервала составляет четыре дня (поскольку отсчет дней месяца (DM) начинается с 1.) Таким образом, промежутки времени между последовательными интервалами будут немного меняться, так что начало интервала каждый раз будет приходиться на 5-е число.

B.5.1.4 Свойство institutionSpecifiedTime: BL

Это свойство указывает, оставляется ли точное время события на усмотрение стороны, которая ведет расписание (например, чтобы отличать «каждые 8 ч» от «3 раза в день»).

Например, при расписании «три раза в день» среднее время между повторениями составляет 8 ч, но в случае, если свойство institutionSpecifiedTime имеет значение «true», точное время повторения должно следовать некоторому правилу, установленному лицом или организацией («стороной»), например, три раза в день могут означать повторения в 7:00, 12:00 и 19:00.

B.5.1.5 Литеральная форма

Общая литеральная форма

B.5.1.5.1 Общая литеральная форма периодических интервалов времени имеет следующий вид:

(phase : IVL<T>(/ (period : QTY ([@ (alignment( ] [IST]

```
PIVL<T>.literal ST {
  PIVL<T>      : S2                 { $.equal($1); }
               | S2 "IST"           { $.phase.equal($1.phase);
                                      $.period.equal($1.period);
$.institutionSpecified.equal(true); };
  PIVL<T> S2   : S1                 { $.equal($1); }
               | S1 "@" "(" ST ")"  { $.phase.equal($1.phase);
                                      $.period.equal($1.period);
                                      $.alignment.equal($4); };
  PIVL<T> S1 :
    IVL<T> "/" "(" QTY ")"          { $.phase.equal($1);
                                      $.period.equal($3); }
               | "/" "(" QTY ")"    { $.period.equal($2); };
};
```

Например, строка «[200004181100;200004181110]/(7 d)@DW» задает каждый вторник с 11:00 до 11:10 AM. С другой стороны, строка «[200004181100;200004181110]/(1 mo)@DM» задает каждое 18-е число месяца с 11:00 до 11:10.

Коды календарных циклов, определенные для григорианского календаря, приведены в таблице B.36. В ней указаны однобуквенные и двухбуквенные коды. В качестве значений свойства alignment предпочтительны двухбуквенные коды.

B.5.1.5.2 Форма календарного шаблона

Эта форма используется для указания привязки повторений к календарю а более интуитивной форме «календарных шаблонов». Синтаксис календарного шаблона (полуформально) определяется следующим образом:

(привязка([(цифры календарного цикла([.. (цифры календарного цикла ( ]] / (число : INT([IST]

Календарный шаблон представляет собой дату календаря, у которой опущены старшие значащие цифры (например, год и месяц). Для интерпретации цифр идентификатору периода приписывается префикс, идентифицирующий календарный цикл крайних левых цифр. Этот идентификатор календарного цикла «привязывает» правые цифры даты.

Коды календарных циклов, определенные для григорианского календаря, приведены в таблице B.36. В ней указаны однобуквенные и двухбуквенные коды. В качестве обозначения привязки календарного шаблона предпочтительны однобуквенные коды.

Например, строка «M0219» означает весь день 19 февраля каждого года. Свойство phase этого периодического интервала соответствует 19 февраля каждого года (например, «[19690219;19690220[«), свойство period имеет значение одного года, а свойство alignment означает месяц года (M). Привязанный календарный цикл совпадает с привязкой (в этом примере с месяцем года).

В календарном шаблоне могут быть также опущены правые цифры. Это означает интервал от наименьшего до наибольшего значения опущенных цифр. Например, строка «M0219» означает весь день 19 февраля; строка «M021918» — весь час 19 февраля от 18:00 до 19:00.

В отсутствие формального определения календарного шаблона, используются следующие правила его разбора (на примере строки «M021918..21»):

1) считать идентификатор привязываемого цикла (например, «M»);

2) положить привязку равной этому календарному циклу (например, месяцу года);

3) взять текущий момент времени и форматировать литерал точно до более высокого календарного цикла, следующего за привязываемым циклом (например, взять год «2000» и сконструировать строку «2000021918»); этот литерал служит «стеблем»;

4) считать сконструированный литерал (например, «2000021918») в значение типа TS и преобразовать его в интервал в соответствии со свойством IVL_TS.promotionTS (например, «[2000021918;2000021919[«); это «нижний интервал»;

5) если после разобранной части строки следует оператор оболочки «..», считать следующие цифры шаблона (например, «21»);

6) выровнять вправо литерал «стебля» и только что считанные цифры шаблона:

“2000021918”

“ 21”;

7) затем скопировать все цифры литерала стебля, отсутствующие в только что считанных цифрах шаблона (например, получить «2000021921»);

8) считать этот сконструированный литерал (например, «2000021918») в значение типа TS и преобразовать его в интервал в соответствии со свойством IVL_TS.promotionTS (например, «[2000021921;2000021922[«); это «верхний интервал»;

9) положить свойство phase равным выпуклой оболочке нижнего и верхнего интервала (например, «[2000021918;2000021922[«);

10) если оператор оболочки отсутствует, то положить свойство phase равным нижнему интервалу.

B.5.1.5.3 Чередование

Календарный шаблон, после которого указана косая черта и целое число n, означающее, что данный шаблон применяется каждый n-й раз.

Например, строка «D19/2» задает 19-е число каждого второго месяца.

Выражение календарного шаблона применяется при первом появлении шаблона. В этот момент отсутствующие левые цифры шаблона дополняются, используя самую раннюю дату, соответствующую шаблону (и следуя предшествующему шаблону в комбинации спецификаций времени).

Например, строка «D19/2» задает 19-е число каждого второго месяца. Если эта строка применяется 14 марта 2000 г., то свойству phase присваивается значение «[20000319;20000320[/(2 mo)@DM» и двухмесячный цикл начинается 19 марта, затем 19 мая и т. д. Если эта строка применяется 20 марта, то цикл начнется 19 апреля, затем 19 июня и т. д.

Если после идентификатора календарного цикла не указаны цифры, то шаблон совпадает с любой датой. Целое число, указанное после косой черты, означает продолжительность цикла. В этих случаях свойство phase обладает только свойством ширины width, указывающим длительность привязанного календарного цикла (в данном примере один день).

Например, строка «CD/2» означает каждый второй день, строка «H/8» означает каждый восьмой час с длительностью интервала один час.

B.5.1.5.4 Время, задаваемое стороной

После литерала типа данных PIVL или календарного шаблона могут быть указаны три буквы «IST» (Institution Specified Time — время, задаваемое стороной), обозначающие, что внутри большего календарного цикла (например, для цикла «час дня» большим циклом будет «день») повторяющиеся события планируются во время, задаваемое стороной. Этот суффикс используется при задании такого цикла, как «три раза в день», при котором между двумя последовательными событиями может пройти и четыре часа (между завтраком и обедом), и 10 ч (охватывающие ночь).

Таблица В.44 — Примеры литеральных выражений типа данных PIVL

Общая литеральная форма	Форма календарного шаблона	Описание
[198709;198710[/(1 a)@MY	M09	Весь месяц сентябрь каждого года (обратите внимание, что указание 1987 г. в общей литеральной форме не имеет особого значения, поскольку периодические интервалы повторяются каждый год как в прошлом, так и в будущем)
[19870915;19870916[/(1 a)@DM	M0915	Весь день 15 сентября каждого года
[1987091516;1987091517[/(1 a)@DM	M091516	Весь час с 16 ч 15 сентября каждого года
[198709151630;198709151710]/(1 a)@DM	M09151630..1710	Интервал 16:30—17:10, повторяющийся каждый год 15 сентября
[1987091516;[/(1 a)@DM		Интервал, начинающийся с 16 ч, конец интервала явно не указан. Повторяется каждый год 15 сентября
[198709151630;198709151631[/(1 a)@DM	M09151630	Интервал в целую минуту, начиная с 16:30. Повторяется каждый год 15 сентября
[1987091516;1987091517[/(1mo)@DM	D1516..17	15-й день каждого месяца с 16 до 17 ч
[1987091516;1987091517[/(1 mo)		15 сентября 1987 года с 16 до 17 ч, и затем через каждые 730.5 ч (в этом примере мало практического смысла, он приведен для сравнения непривязанной формы с привязанной, указанной в предыдущей строке)
[1987091516;1987091517[/(1 mo)@HD		15 сентября 1987 г. с 16 до 17 ч и затем через каждые 30.4375 дней, но привязанный к часу дня
[1 mo]/(2 mo)@MY	M/2	Каждый второй месяц года; не определено, будет ли это последовательность (январь, март, ...) или (февраль, апрель, ...)
[198701;197502[/(2 mo)@MY	M01..12/2	Каждый второй месяц года, январь, март, ...
[198702;197503[/(2 mo)@MY	M02..12/2	Каждый второй месяц года, февраль, апрель, ...
[19870401;19870930[/(1 a)@DM	M04..09	С 1 апреля по 30 сентября (включительно)
19870401-0930/(1 a)@DM	M0401..0930	С 1 апреля по 30 сентября (в общей литеральной форме фаза phase задана через дефис)
[20001202;20001203[/(1 wk)@DW	J6	Каждую субботу
[20001202;20001203[/(2 wk)@DW	J6/2	Каждую вторую субботу
[20001202;20001203[/(3 wk)@DW	J6/3	Каждую третью субботу
[1 d]/(2 d)@DW	J/2	Каждый второй день недели, не определено, будет ли это последовательность (понедельник, среда, пятница...) или (вторник, четверг, суббота, ...)
[20001204;20001205[/(2 d)@DW	J2..6/2	Каждый второй день недели (вторник, четверг, суббота, вторник, четверг, суббота, ...).
[20001204;20001205[/(2 d)	D/2	Каждый второй день (вторник, четверг, суббота, понедельник, среда, пятница, воскресенье, вторник, ...)
[19870601;19870606[/(1 wk)@DW	J1..5	Каждую неделю с понедельника по пятницу

Окончание таблицы В.44

Общая литеральная форма	Форма календарного шаблона	Описание
[19870601;19870608[/(2 wk)	W/2	Каждую вторую неделю (непрерывно)
[19870101;19870105[/(2 wk)@WY	WY/2	Каждую вторую неделю года (характерный пример влияния привязки к календарю: продолжительность фазы составляет только 4 дня и тем не менее представляет целую неделю в календарной привязке «неделя года)
[19870406;19870413[/(1 a)@WY	WY15	15-я календарная неделя каждого года
[19870105;19870112[/(1 mo)@WM	WM2	Вторая неделя каждого месяца
[19870508;19870509[/(1 a)@DY	DY128	128-й день каждого года
[10 min]/(2 d)		Каждый второй день в течение 10 мин (известна только ширина повторяющегося интервала)
[1 h]/(8 h)	H/8	Каждый восьмой час (в течение 60 мин)
[1 h]/(8 h) IST	H/8 IST	Три раза в день во время, заданное стороной (каждый раз в течение 60 мин)
/(8 h) IST		Три раза в день во время, заданное стороной. О повторяющемся интервале ничего не известно, то есть литерал описывает только период (частоту), а фаза остается не определенной

В.5.1.6 Периодические интервалы как множества

Существенным свойством множества является то, что оно содержит элементы. Для значений типов PIVL, не привязанных к календарю, свойство contains (содержит) определяется следующим образом: элемент TS t содержится во множестве типа PIVL в том и только том случае, если существует целое число i, для которого сумма *t* с произведением period на i является элементом фазы phase.

```
invariant (PIVL<TS> x, TS t)
    where x.nonNull.and(x.alignment.isNull) {
  x.contains(t).equal(exists(INT i) {
    x.phase.contains(t.plus(x.period.times(i)));
    });
};
```

Для значений типов PIVL, привязанных к календарю, свойство contains определяется, используя свойство календарного цикла sum(t, n), которое добавляет n таких календарных циклов к моменту времени t.

```
invariant (PIVL<TS> x, TS t, CalendarCycle c)
    where x.nonNull.and(c.equal(x.alignment)) {
  x.contains(t).equal(exists(INT i) {
    x.phase.contains(c.sum(t, i));
    });
};
```

В.5.2 Тип данных периодического интервала времени с привязкой к событию EIVL (специализация типа данных SET)

Задает периодический интервал времени, повторения которого основаны на событиях повседневной деятельности или других важных событиях, которые связаны с временем, но не имеют точно заданного времени.

Например, расписание «через час после завтрака « указывает, что интервал начинается через час после завершения завтрака. Предполагается, что завтрак происходит до обеда, но при этом никакое конкретное время завтрака не определено.

```
template<TS T>
protected type EventRelatedPeriodicInterval<T> alias EIVL<T>
        specializes  SET<T>{
        CS           event;
        IVL<PQ>      offset;
        IVL<T>       occurrenceAt(TS eventTime);
        BL           contains(TS);
 literal ST;
};
```

B.5.2.1 Свойство event : CS

Код повседневной (периодической) деятельности, на основе которой определяется периодический интервал, связанный с событием.

В домен значений этого атрибута могут включаться такие события, для которых верно все нижеперечисленное:

- событие обычно происходит на регулярной основе;
- событие означает деятельность в течение некоторого времени;
- событие не полностью определяется временем.

Таблица B.45 — Домен значений свойства event

Код	Имя	Определение
AC	AC	Перед едой (лат. ante cibus)
ACD	ACT	Перед обедом (лат. ante cibus diurnus)
ACM	ACM	Перед завтраком (лат. ante cibus matutinus)
ACV	ACV	Перед ужином (лат. ante cibus vespertinus)
HS	HS	Часы сна
IC	IC	Между приемами пищи (лат. inter cibus)
ICD	ICD	Между обедом и ужином
ICM	ICM	Между завтраком и обедом
ICV	ICV	Между ужином и часами сна
PC	PC	После еды (post cibus)
PCD	PCD	После обеда (post cibus diurnus)
PCM	PCM	После завтрака (post cibus matutinus)
PCV	PCV	После ужина (post cibus vespertinus)

B.5.2.2 Свойство offset: IVL<PQ>

Интервал прошедшего времени (длительность, а не абсолютные моменты времени), задающий смещения начала, ширины и конца значения типа EIVL относительно времени фактического совершения такого события.

Например, для расписания «за один час до завтрака в течение 10 минут» значение IVL.low имеет свойство offset, равное 1 час, а значение IVL.width имеет свойство offset, равное 10 минут.

B.5.2.3 Литеральная форма

Литеральная форма типа данных EIVL начинается с кода события, за которым следует необязательный интервал разности во времени.

```
EIVL<TS>.literal ST {
  EIVL<TS> : event                { $.event.equal($1); }
           | event offset         { $.event.equal($1);
                                    $.offset.equal($2); };
  CS event  : ST                  { $.code.equal($1);
$.codeSystem.equal(2.16.840.1.113883.5.1019); }
```

```
  IVL<PQ> offset
            : "+" IVL<PQ>          { $.equal($2); }
            | "-" IVL<PQ>          { $.low.equal($2.high.negate);
                                     $.high.equal($2.low.negate);
                                     $.width.equal($2.width);
                                     $.lowClosed($2.highClosed);
                                     $.highClosed($2.lowClosed); };
};
```

Например, расписание "через один час после еды" имеет литерал "PC+[1h;1h]". Расписание «за один час перед сном в течение 10 минут» имеет литерал «HS-[50min;1h]».

B.5.2.4 Разрешение связи с событием

Тип данных EIVL описывает множество моментов времени, поэтому можно проверить, является ли конкретный момент или интервал времени элементом этого множества. Вопрос, содержит ли значение типа EIVL данный интервал времени, разрешается с помощью отношения χ time, обозначаемого как EVENT(event, time). Свойство occurrenceAt(*t*) задает повторение интервала, имеющее место, если событие происходит в момент времени t.

```
invariant(EIVL<T> x, T eventTime, IVL<T> v)
    where v.equal(x.occurrenceAt(eventTime)) {
  v.low.equal(eventTime.plus(x.offset.low));
  v.high.equal(eventTime.plus(x.offset.high));
  v.lowClosed.equal(x.offset.lowClosed);
  v.highClosed.equal(x.offset.highClosed);
};
```

Таким образом, значение типа EIVL содержит момент времени TS t, если существует время события e с повторением интервала v, содержащим t.

```
invariant(EIVL<T> x, T y) {
  x.contains(y).equal(exists(T e, IVL<T> v)
    where EVENT(x.event, y).and(v.resolvedAt(y)) {
      v.contains(y);
      });
};
```

B.5.3 Тип данных общей спецификации времени GTS (специализация типа данных SET<TS>)

Множество моментов времени, указывающее времена событий и действий, а также шаблоны циклов, которые могут применяться к некоторым видам информации, например, к телефонным номерам (утро, вечер), к адресам (для так называемых «перелетных птиц», которые живут зимой на юге, а летом на севере), а также к часам работы.

Тип GTS имеет следующие особенности:

- он является общим типом множества моментов времени SET<TS>. Поэтому тип данных GTS может использоваться для определения того, попадает ли данное значение типа TS в расписание, описанное значением типа GTS;
- тип GTS представляет собой сочетание нескольких интервалов типа PIVL. Эта особенность позволяет определить, как именно простые и комплексные шаблоны повторений могут быть описаны с помощью типа данных GTS;
- тип данных GTS может служить генератором списков типа LIST<IVL<TS>>. Эта особенность позволяет использовать тип данных GTS для генерирования всех повторяющихся интервалов события или действия либо всех периодов действительности факта;
- тип данных GTS может использоваться как синтаксис календарных выражений. Эта особенность характерна для литеральной формы этого типа данных.

В любом случае тип данных GTS определяется как множество моментов времени SET<TS>. С помощью свойств SET.union, SET.intersect и SET.difference из простых множеств типа SET<TS> можно сконструировать более сложные. В конечном счете строительными блоками всех значений типа GTS являются типы данных IVL, PIVL и EIVL. Конструкция значения типа GTS может быть определена в литеральной форме. Для генерации комбинации более простых множеств моментов времени из данного значения типа GTS не определена никакая специальная

структура типа данных. Хотя любая реализация и могла бы содержать такое структурированное представление, оно не является необходимым, чтобы передавать значения типа GTS в литеральной форме[62)].

```
type GeneralTimingSpecification alias GTS specializes SET<TS> {
            IVL<TS>   hull;
            IVL<TS>   nextTo(TS x)
            IVL<TS>   nextAfter(TS x)
            GTS       periodicHull(GTS x);
            BL        interleaves(GTS x);
 demotion   LIST<IVL<TS>>;
 literal    ST;
};
```

B.5.3.1 Выпуклая оболочка

Выпуклая оболочка представляет собой наименьший интервал, содержащий все интервалы расписания. Как указано в описании свойства SET.hull, для всех полностью упорядоченных множество определена выпуклая оболочка. Поскольку тип данных GTS является специализацией типа данных SET<TS>, значения которого полностью упорядочены, то для всех значений типа GTS определена выпуклая оболочка.

Выпуклая облочка значения типа GTS может быть менее формально названа «внешним связным интервалом». Таким образом, выпуклая оболочка значения типа GTS описывают абсолютные начало и конец повторяющегося расписания. Для бесконечных повторений (например, описываемых типом данных PIVL) выпуклая оболочка имеет бесконечные границы.

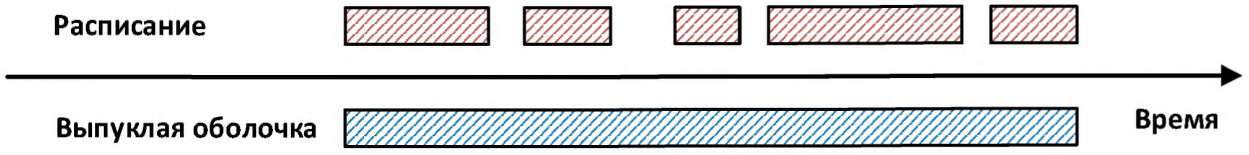


Рисунок B.16 — Выпуклая оболочка расписания

B.5.3.2 Тип данных GTS как последовательность интервалов событий

Значение типа GTS является генератором интервалов, в течение которых происходят события или осуществляются действия либо действительно некоторое состояние.

Свойство nextTo отображает каждый момент времени t на наибольшее непрерывное подмножество («интервал события») v значения S типа GTS, которое представляет собой ближайший интервал, начинающийся позже момента t или содержащий момент t.

```
invariant(GTS S, TS t, IVL<TS> v) {
  v.equal(S.nextTo(t)).equal(
    S.contains(o).and(
      forall(IVL<TS> u) where x.contains(u) {
        u.contains(v).implies(u.equal(v));
        })
      .and(v.contains(t).or(forall(TS i) where t.lessOrEqual(i)
          .and(i.lessThan(v.low)) {
        S.conatins(i).not; })));
};
```

Свойство nextAfter отображает каждый момент времени t на наибольшее непрерывное подмножество («интервал события») v значения S типа GTS, которое представляет собой ближайший интервал, начинающийся позже момента t.

---

62) Тип данных GTS служит примером типа данных, который имеет только алгебраическое определение, а не описывается структурой данных, с помощью которой можно было бы реализовать поведение такого типа данных. Алгебраическое определение выглядит чрезвычайно простым, поэтому можно подозревать его неполноту. Поскольку в настоящее время для представления значений типа GTS используется только литеральная форма, то все определение структуры данных оставлено на будущее.

```
invariant(GTS S, TS t) {
  S.contains(t).not.implies(S.nextAfter(t).equal(S.nextTo(t)));
  S.contains(t).implies(S.nextAfter(t).equal(
          S.except(nextTo(t)).nextTo(t)));
};
```


```
invariant(GTS S, TS t) {
  S.contains(t).not.implies(S.nextAfter(t).equal(S.nextTo(t)));
  S.contains(t).implies(S.nextAfter(t).equal(
          S.except(nextTo(t)).nextTo(t)));
};
```


Значение типа GTS может быть преобразовано в список типа LIST<IVL<TS>>.

```
invariant(GTS x)
    where x.isEmpty {
  ((LIST<IVL<TS>>)x).isEmpty; };

invariant(GTS x, IVL<TS> first)
    where x.notEmpty.and(x.hull.low.nonNull)
                .and(first.equal(x.nextTo(x.hull.low)) {
  ((LIST<IVL<TS>>)x).head.equal(first);
  ((LIST<IVL<TS>>)x).tail.equal(
   (LIST<IVL<TS>>)x.except(first));
};
```

B.5.3.3 Чередущиеся расписания и периодические оболочки

Два значения A и B типа GTS чередуются, если интервалы их событий чередуются на оси времени. Это понятие иллюстрируется рисунком B.17.

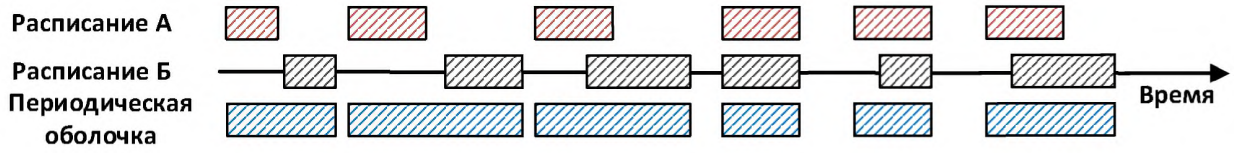


Рисунок B.17 — Чередущиеся расписания и периодическая оболочка

Значения A и B типа GTS считаются чередующимися, если интервалы событий обоих групп могут быть попарно упорядочены. При этом соответствующие пары интервалов $a \subseteq A$ и $b \subseteq B$ должны удовлетворять следующему условию: интервал a начинается до начала интервала b (или в то же самое время), и интервал b завершается после завершения интервала a (или в то же самое время).

Отношение чередования имеет место, если два расписания имеют одинаковую среднюю частоту и при этом второе расписание никогда не «перекрывает» первое расписание. Другими словами, никакой интервал события второго расписания не может начаться раньше соответствующего ему интервала события первого расписания.

Для двух чередующихся значений типа GTS можно определить периодическую оболочку, представляющую собой совокупность выпуклых оболочек соответствующих пар интервалов событий.

Периодическая оболочка важна для конструирования двух расписаний с помощью сочетания значений типа GTS. Например, для конструирования ежегодного периодического интервала от Дня памяти (последний понедельник мая) до Дня труда (первый понедельник сентября) можно сконструировать расписание M для Дня памяти и расписание L для Дня труда, а затем скомбинировать эти два расписания, используя периодическую оболочку расписаний M и L.

```
invariant(GTS A, B)
    where x.nonNull.and(y.nonNull) {
  A.interleaves(B).equal(
    forall(IVL<TS> a, b, c; TS t)
```

```
      where a.equal(A.nextTo(t))
              .and(b.equal(B.nextTo(a.low)))
              .and(c.equal(A.nextTo(b.high))) {
        b.equal(B.nextTo(a.high));
        a.low.lessOrEqual(b.low);
        c.equal(A.nextTo(b.high));
        c.equal(a).or(c.equal(A.nextAfter(a.high)));
        });
};
```

Периодическая оболочка двух чередующихся значений A и B типа GTS определяется как совокупность выпуклых оболочек соответствующих интервалов пар интервалов событий.

```
invariant(GTS A, B, C)
    where A.interleaves(B) {
  A.periodicHull(B).equal(C).equal(
    forall(IVL<TS> a, b; TS t)
      where a.equal(A.nextTo(t))
              .and(b.equal(B.nextTo(a.low))) {
        C.contains(c).equal(c.equal(a.hull(b)));
        });
};
```

Отношение чередования является рефлексивным, несимметричным и нетранзитивным. Операция периодической оболочки не коммутативна и не ассоциативна The periodic hull operation is non-commutative and non-associative[63].

B.5.3.4 Литеральная форма

Литерал типа данных GTS позволяет указывать комбинации значений типов данных IVL<TS> и PIVL, используя операции объединения и пересечения множеств[64].

Объединения описываются в форме списка значений, разделенных точками с запятыми. Пересечения задаются в форме списка значений, разделенных пробельными символами. Операция пересечения имеет более высокий приоритет по отношению к операции объединения. Разность множеств задается с помощью символа обратной косой черты, она имеет промежуточный приоритет, то есть слабее пересечения, но сильнее объединения. При необходимости для переопределения порядка применения операций могут использоваться скобки.

```
GTS.literal ST {
  GTS symbol        : union                  { $.equal($1); }
                    | exclusion              { $.equal($1); };
  SET<TS> union     : symbol ";" intersection { $.equal($1.union($3)); }
                    | intersection           { $.equal($1); };
  SET<TS> exclusion
                    : symbol "\" intersection { $.equal($1.except($3)); }
                    | intersection           { $.equal($1); };
  SET<TS> intersection
```

---

[63] Свойство чередования может показаться чрезмерно ограниченным. Однако эти ограничения вполне разумны в тех случаях, когда определяются свойства чередования и периодической оболочки. Для безопасного и предсказуемого комбинирования двух расписаний необходимо знать, который из операндов задает начальные моменты времени, а какой — конечные моменты периодической оболочки интервалов событий.

[64] Эта спецификация литерала опять-таки выглядит удивительно простой, поэтому можно подозревать ее неполной. Однако литерал типа данных GTS основан на литералах типов данных TS, IVL, PIVL и EIVL, а также подразумевает литералы расширений типа данных TS, в частности, параметризованного типа данных PPD<TS>. Спецификация самого литерала GTS должна только связывать между собой другие литеральные формы, что само по себе представляет достаточно простой задачей.

```
                : hull intersection        { $.equal($1.intersection($2)); }
                | hull                     { $.equal($1); };
  SET<TS> hull
                : hull ".." factor         { $.equal($1.periodicHull($3)); }
                | factor                   { $.equal($1); };
  SET<TS> factor
                : IVL<TS>                  { $.equal($1); }
                | PIVL<TS>                 { $.equal($1); }
                | EIVL<TS>                 { $.equal($1); }
                | «(« GTS «)»              { $.equal($1); };
};
```

В таблице В.46 приведены практические примеры комплексных литералов типа данных GTS. Более простые примеры приведены в описании литеральных форм типов данных IVL, PIVL и EIVL.

Таблица В.46 — Примеры литеральных выражений типа данных GTS

Литеральное выражение	Значение
M09 D15 H16 N30 S34.12	15 сентября, 16:30:34.12 как пересечение нескольких периодических интервалов времени (календарных шаблонов)
M0915163034.12	15 сентября, 16:30:34.12 как один простой периодический интервал времени (календарный шаблон)
M01; M03; M07	Январь, март и июнь (объединение трех периодических интервалов времени)
M04..09 M/2	Каждый второй месяц с апреля по сентябрь (апрель, июнь, август)
J1; J2; J4	Понедельник, вторник, четверг
W/2 J2	Каждый второй вторник Tuesday (пересечение каждой второй недели и каждого вторника)
1999 WY15	15-я календарная неделя 1999 г. (код периода старшей календарной единицы не обязателен)
WM2 J6	Суббота второй недели месяца
M05 WM2 J6	Суббота второй недели мая
M05 DM08..14 J7	День матери (второе воскресенье мая)
J1..5 H0800..1600	С 8:00 до 16:00 с понедельника по пятницу
J1..4 H0800..1600; J5 H0800..1200	С 8:00 до 16:00 с понедельника по четверг и с 8:00 до полудня в пятницу
[10 d] H/8	Три раза в день в течение 10 дней (каждый раз в течение 60-минутного интервала)
H0800..1600 \J3	С 8:00 до 16:00 каждый день, кроме среды
(M0825..31 J1)..M0831	Последняя календарная неделя августа
JHNUSMEM..JHNUSLBR	Сезон праздников США с Дня памяти до Дня труда

В.5.3.4.1 Символьное сокращение выражений типа данных GTS

В таблице В.47 приведены символьные сокращения значений типа GTS, которые могут использоваться в литералах вместо их эквивалентов. Сокращения определены для общих периодов дня (AM — утро, PM — пополудни), для периодов недели (рабочий день, выходной) и для праздников. Вычисления дат некоторых праздников, прежде всего Пасхи, представляют определенную сложность, превышающую возможности их представления в форме литерала типа данных GTS. Предполагается, что даты таких праздников берутся из некоторой таблицы или генерируются некоторым модулем вне области применения настоящей спецификации.

Эти сокращения являются именованными значениями типа GTS и могут в свою очередь использоваться в литерале типа GTS. Например, можно использовать строку «JHCHRXME H08..12» для указание, что рабочими

часами в Канун рождества ограничены интервалом с 8 до 12 ч. А строка «JHNUSMEM..JHNUSLBR» может использоваться для обозначения типичного плавательного сезона на Среднем Западе, продолжающимся от Дня памяти до Дня труда.

Таблица B.47 — Домен сокращений выражений типа данных GTS

Код	Определение	Формальное определение
AM	Каждое утро во время, заданное стороной	H00..11 IST
PM	Каждая середина дня во время, заданное стороной	H12..23 IST
BID	Два раза в день во время, заданное стороной	/(12 h) IST
TID	Три раза в день во время, заданное стороной	/(8 h) IST
QID	Четыре раза в день во время, заданное стороной	/(6 h) IST
JB	Обычные рабочие дни (с понедельника по пятницу, исключая праздники)	J1..5 \JH
JE	Обычные выходные дни (суббота и воскресенье, не считая праздники)	J6..7
JH	Праздники	
GTSAbbreviationHolidaysChristianRoman	Сокращения для римских христианских праздников	
JHCHRXME		M1224
JHCHRXMS		M1225
JHCHRNEW		M0101
JHCHREAS		
JHCHRGFR		
JHCHRPEN		
JHNUS	Национальные праздники США (праздничные дни, установленные для федеральных служащих Федеральным законом США 5 U.S.C. 6103)	
JHNUSMLK	День Мартина Лютера Кинга (третий понедельник января)	M0115..21 J1
JHNUSPRE	День рождения Вашингтона (День Президента), третий понедельник февраля	M0215..21 J1
JHNUSMEM	День памяти (последний понедельник мая)	M0525..31 J1
JHNUSMEM5	Пятница недели перед Днем памяти	M0522..28 J5
JHNUSMEM6	Суббота недели перед Днем памяти	M0523..29 J6
JHNUSIND	День независимости (4-е июля)	M0704
JHNUSIND5	Пятница, непосредственно предшествующая 4 июля [статья (b) Федерального закона 5 U.S.C. 6103]	M0703 J5
JHNUSIND1	Понедельник, непосредственно следующий за 4 июля [статья (b) Федерального закона 5 U.S.C. 6103]	M0705 J1
JHNUSLBR	День труда (первый понедельник сентября)	M0901..07 J1
JHNUSCLM	День Колумба (второй понедельник октября)	M1008..14 J1
JHNUSVET	День ветеранов (11 ноября)	M1111
JHNUSTKS	День благодарения (четвертый четверг ноября)	M1122..28 J4
JHNUSTKS5	Пятница после Дня благодарения	M1123..29 J5

Примечания

1 Эта таблица не полна. Она не включает нехристианские религиозные праздники (установленные григорианской (западной) традицией) или национальные праздники, установленные за пределами США. Эти ограничения могут быть устранены с помощью постепенного пополнения таблицы.

2 Праздники имеют местную специфику. Какие именно религиозные праздники отмечаются под сокращением JH, зависит от местной и других традиций. Для обеспечения глобальной интероперабельности вместо именованных сокращений праздников безопаснее использовать специально сконструированные литералы GTS. Однако некоторые праздники, зависящие от фаз луны (например, Пасха), или декретированные праздники не могут быть простым образом представлены литералом типа данных GTS.

B.6 Справочные типы данных

Эти типы данных в настоящее время отнесены к числу справочных, пока не будут разрешены известные вопросы, связанные с их конструированием.

B.6.1 Тип данных параметрического распределения вероятности PPD (специализация типа данных T)

Параметризованное расширение типа данных, описывающее неопределенность количественных значений с помощью функции распределения и ее параметров. Помимо специфичных параметров распределения всегда определяются среднее значение (математическое ожидание) и стандартное отклонение, что обеспечивает минимальный уровень интероперабельности в случае, когда приложения-получатели не могут обеспечить обработку определенного распределения вероятности.

Таблица B.48 — Сводка свойств типа данных PPD<T>

Имя	Тип	Описание
standardDeviation	QTY	Основная мера вариабельности/неопределенности значения (квадратный корень из суммы квадратов всех разностей значений данных и математического ожидания). Свойство standardDeviation используется для нормализации данных в целях вычисления функции распределения. По свойству standardDeviation приложения, которые не могут обеспечить обработку функции распределения, могут получить определенную информацию об уровне доверия к данным
distributionType	CE	Код, указывающий тип распределения вероятности. Возможные значения показаны в таблице B.49. Пустое значение кода (с причиной пустоты unknown — не известно) означает, что функция распределения вероятности не известна. В этом случае значение свойства standardDeviation означает эмпирическую догадку

```
template<QTY T>
type ParametricProbabilityDistribution<T> alias PPD<T> specializes T {
  QTY      standardDeviation;
  CS       distributionType;
  IVL<T>   confidenceInterval(REAL p);
  REAL     probability(IVL<T> x);
  PPD<T>   times(REAL x);
};
```

Например, наиболее распространенным вступительным экзаменом в высшие учебные заведения США является SAT, состоящий из двух разделов: анализ текста и математика. Каждая из них имеет минимальную оценку 400 (ни одного правильного ответа на вопросы) и максимальную 800. В 1998 году, согласно данным организации College Board, в этом тесте приняли участие 1,172,779 абитуриентов. Средняя оценка по математическому разделу составила 512, а стандартное отклонение — 112. Нормальное распределение со значениями параметров (512, 112), очень хорошо согласуется с реальным распределением оценок, полученных участниками тестирования. В большинстве случаев нет необходимости указывать свыше миллиона результатов, когда вместо них можно использовать только 2 параметра!

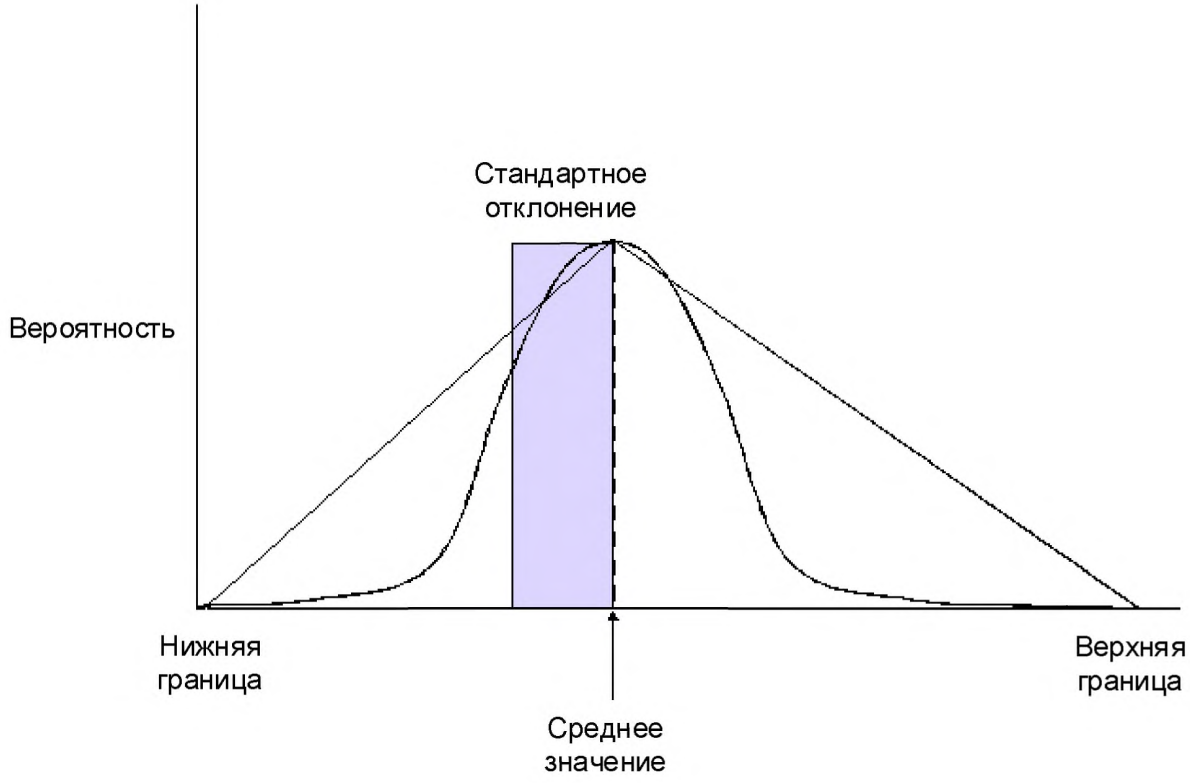


Рисунок В.18 — Пример параметрического распределения вероятности

Учтите, что нормальное распределение является только одним из нескольких распределений, используемых в стандартах HL7.

Так как тип данных PPD является специализацией типа T, то простое значение T является средним значением (математическим ожиданием) распределения вероятности. Если приложения не могут обеспечить обработку распределения вероятности, то они могут воспользоваться простым значением T и пренебречь его неопределенностью. Это простое значение T также используется для стандартизации данных при вычислении распределения.

Распределения вероятности определены для целых или вещественных чисел и нормализованы по отношению к некоторой точке отсчета (обычно нуль) и эталонной единице (например, стандартное отклонение standardDeviation = 1). Когда в настоящей спецификации в качестве базовых типов используются другие величины, то для масштабирования распределения вероятности используются среднее значении и стандартное отклонение standardDeviation. Например, если величина типа PPD<PQ> представляет собой длину со средним значением 20 футов и стандартным отклонением standardDeviation, равным 2 дюймам, то нормализованная функция распределения $f(x)$, отображающая вещественное число x на плотность вероятности, должна быть преобразована в функцию $f'(x')$, отображающую длину x' на плотность вероятности по формуле $f'(x') = f((x' - \mu) / \sigma)$.

По возможности тип данных PPD соответствует требованиям, изложенным в документе ISO Guide to the Expression of Uncertainty in Measurement (GUM) (руководство по представлению неопределенности измерений) и отраженным в публикации организации NIST «1297 Guidelines for Evaluating and Expressing the Uncertainty of NIST Measurement Results». Тип данных PPD описывает только то, как представляется неопределенность, а не как она вычисляется. Понятие «стандартной неопределенности», введенное в документе ISO GUM, соответствует свойству standardDeviation.

B.6.1.1 Свойство standardDeviation : QTY

Основная мера вариабельности/неопределенности значения (квадратный корень из суммы квадратов всех разностей значений данных и математического ожидания). Свойство standardDeviation используется для нормализации данных в целях вычисления функции распределения. По свойству standardDeviation приложения, которые не могут обеспечить обработку функции распределения, могут получить определенную информацию об уровне доверия к данным.

Свойство standardDeviation является специализацией типа данных QTY (произведенного от типа данных T.diffType), представляющего разности значений типа T. Если в качестве T выступают типы данных REAL или INT, то свойство T.diffType также имеет значение REAL или INT соответственно. Но если тип данных T равен TS, то тип данных T.diffType является типом данных физической величины PQ с размерностью времени.

```
invariant(PPD x) {
  x.standardDeviation.dataType.implies(T.diffType);
};
```

B.6.1.2 Свойство distributionType: CE

Код, указывающий тип распределения вероятности. Возможные значения показаны в таблице B.49. Пустое значение кода (с причиной пустоты unknown — не известно) означает, что функция распределения вероятности не известна. В этом случае значение свойства standardDeviation означает эмпирическую догадку.

Определенные распределения вероятности перечислены в таблице B.49. Многие типы распределений определены в терминах специальных параметров (например, α и β в γ-распределении, число степеней свободы в t-распределении и т. д.) Однако при этом для всех типов распределений определены среднее значение и стандартное отклонение.

Таблица B.49 — Домен значений свойства distributionType

Код	Имя	Определение
(NULL)		Пустое значение кода указывает, что среднее значение оценивалось без существенного учета распределения вероятности. В этом случае значение стандартного отклонения не имеет строго обоснованной интерпретации, но можно исходить из того, что принято для нормального распределения, а именно, что интервал «среднее значение ± одно стандартное отклонение» имеет доверительный уровень, примерна равный двум третям
U	Равномерное	Равномерное распределение присваивает одинаковую плотность вероятности значениям, принадлежащим определенному интервалу, и нулевую плотность всем остальным значениям. Ширина интервала составляет $2\ \sigma\ \sqrt{3}$. Таким образом, при равномерном распределении присваивается плотность вероятности $f(x) = (2\ \sigma\ \sqrt{3})^{-1}$ значениям в интервале $\mu - \sigma\ \sqrt{3} \geq x \leq \mu + \sigma\ \sqrt{3}$ и $f(x) = 0$ остальным значениям
N	Нормальное (Гауссово)	Это хорошо известное распределение вероятности с плотностью в форме колокола. В силу центральной предельной теоремы теории вероятностей нормальное распределение имеют неограниченные случайные значения, являющиеся результатом многих случайных процессов. Нормальное распределение является достаточно точным даже для значений, ограниченных с одной стороны (например, больших 0), если в терминах стандартных отклонений среднее значение находится «достаточно далеко» от границы шкалы
LN	Логарифмически-нормальное распределение	Логарифмически-нормальное распределение используется для преобразования экспоненциально растущей случайной переменной в нормально распределенную случайную переменную $U = \log X$. Для логарифмически-нормального распределения могут быть заданы среднее значение μ и стандартное отклонение σ. Эти значения являются параметрами исходного распределения, а не преобразованными параметрами нормального распределения, которые традиционно имеют те же самые буквенные обозначения. Эти логарифмически-нормальные параметры μ_{log} и σ_{log} связаны со средним значением μ и стандартным отклонением σ значений данных по формулам $\sigma_{log}^2 = \log(\sigma^2/\mu^2 + 1)$ и $\mu_{log} = \log \mu - \sigma_{log}^2/2$
G	γ (гамма)	Гамма-распределение (γ) используется для данных с наклонным графиком и правой границей, при которых максимум кривой распределения находится возле начала координат. Оно имеет два параметра α и β. Связи со средним значением μ и отклонением σ^2 задаются формулами $\mu = \alpha\ \beta$ и $\sigma^2 = \alpha\ \beta^2$
E	Экспоненциальное	Используется для данных, описывающих вымирание. Экспоненциальное распределение является частным случаем гамма-распределения, у которого $\alpha = 1$, следовательно, для вычисления среднего значения μ и отклонения σ^2 справедливы формулы $\mu = \beta$ и $\sigma^2 = \beta^2$
X2	χ	Используется для описания суммы случайных переменных, возникающих при оценке (а не измерении) отклонения от образца. Распределение χ^2 имеет единственный параметр υ, так называемое число степеней свободы (равное числу независимых слагаемых). Это распределение является частным случаем гамма-распределения, у которого $\alpha = \upsilon/2$ and $\beta = 2$, следовательно, для вычисления среднего значения μ и отклонения σ^2 справедливы формулы $\mu = \upsilon$ и $\sigma^2 = 2\ \upsilon$

Окончание таблицы В.49

Код	Имя	Определение
T	t (распределение Стьюдента)	Используется для описания отношения нормальной случайной переменной к квадратному корню от случайной переменной, имеющей распределение вероятности χ^2. Распределение Стьюдента имеет один параметр υ (число степеней свободы). Его связь со средним значением μ и отклонением σ^2 определяется формулами $\mu = 0$ и $\sigma^2 = \upsilon / (\upsilon - 2)$
F	F	Используется для описания отношения двух случайных переменных, имеющих распределение вероятности χ^2. F-распределение имеет два параметра υ_1 и υ_2, которые являются числами степеней свободы соответственно числителя и знаменателя. Связь этих параметров со средним значением μ и отклонением σ^2 определяется формулами $\mu = \upsilon_2 / (\upsilon_2 - 2)$ и $\sigma^2 = (2\,\upsilon_2\,(\upsilon_2 + \upsilon_1 - 2)) / (\upsilon_1\,(\upsilon_2 - 2)^2\,(\upsilon_2 - 4))$
B	β (beta)	Бета-распределение используется для данных, ограниченных с обоих сторон, график которых может быть, а может и не быть наклонным (что встречается при оценке вероятности). Для задания кривой используются два параметра, α и β. Связь этих параметров со средним значением μ и отклонением σ^2 определяется формулами $\mu = \alpha / (\alpha + \beta)$ и $(\sigma^2 = \alpha\,\beta/((\alpha + \beta)^2\,(\alpha + \beta + 1))$

Обработка трех типов распределений, а именно, неизвестное (NULL), равномерное и нормальное, должна обеспечиваться каждой системой, объявившей о поддержке типа данных PPD. Все остальные типы распределений не обязательны. Если система, интерпретирующая представление значения типа PPD, обнаруживает тип распределения, который она не может распознать, то она должна отобразить его на тип распределение «неизвестное» (NULL).

В.6.1.3 Литеральная форма

Общий синтаксис литерала значения типа PPD задается следующим образом:

```
PPD<T>.literal ST {
  PPD<T>  : T "(" type QTY ")"        { ((T)$).equal($1);
                                        $.distributionType.equal($3);
                                        $.standardDeviation.equal($4); };
  CV type   : ST                      { $.value.equal($1);

$.codeSystem.equal(2.16.840.1.113883.5.1020); };
};
```

Примеры литерал значения типа PPD<REAL> с нормальным распределением, средним значением 1.23 и стандартным отклонением 0.005 задается строкой «1.23(N0.005)». Примером литерала значения типа PPD<PQ> служит строка «1.23 m (5 mm)», описывающая неизвестное распределение со средним значением 1.23 метра и стандартным отклонением 5 миллиметров. Примером литерала значения типа PPD<TS> служит строка «2000041113(U4 h)», описывающая равномерное распределение со средним значением 13:00 1 апреля 2000 года и стандартным отклонением 4 часа.

В.6.2 Тип данных распределения вероятности вещественных значений PPD<REAL> (специализация типа данных PPD)

```
type ParametricProbabilityDistribution<REAL> alias PPD<REAL>;
```

Параметрическое распределение вероятности вещественных значений полностью определяется специализируемым типом данных. Однако для литеральных представлений и преобразований распределения вероятности вещественных значений имеются некоторые специальные возможности, обсуждаемые в настоящем подразделе.

В.6.2.1 Преобразование вещественного значения (тип данных REAL) в неопределенное вещественное значение (тип данных PPD<REAL>)

При преобразовании вещественного значения (типа REAL в значение типа PPD<REAL> стандартное отклонение PPD.standardDeviation вычисляется из порядка величины вещественного значения и из значения свойства REAL.precision (число значащих цифр). Пусть x — вещественное значение типа REAL, у которого свойство REAL. precision имеет значение n. Порядок величины e значения x можно определить по формуле e = log10 |x|, где e округляется до следующего целого числа, ближайшего к нулю (особый случай: если значение x равно нулю, то e равно нулю). Тогда значение наименьшей значащей цифры I равно 10^{e-n} и стандартное отклонение σ (то есть значение свойства PPD.standardDeviation) определяется формулой $\sigma = I / 2$.

Таблица B.50 — Примеры стандартных отклонений, вычисленных по точности n и порядку величины e

Представление	x	e	n	e – n + 1	l	σ
0	0	(0)	1	0	1	0.5
1	1	0	1	0	1	0.5
2	2	0	1	0	1	0.5
9	9	0	1	0	1	0.5
10	10	1	2	0	1	0.5
100	100	2	3	0	1	0.5
1e + 1	10	1	1	1	10	5
1e + 2	100	2	1	2	100	50
10e + 1	100	2	2	1	10	5
1.1	1.1	0	2	-1	0.1	0.05
10.1	10.1	1	3	-1	0.1	0.05
1.1e + 2	110	2	2	1	10	5
1.1e – 2	0.011	-2	2	-3	0.001	0.0005
1.1e – 4	0.00011	-4	2	-5	0.00001	0.000005
10.1e – 4	0.00101	-3	3	-5	0.00001	0.000005
0.1e – 1	0.01	-2	1	-2	0.01	0.005
0.01e – 1	0.001	-3	1	-3	0.001	0.0005
0.01e – 2	0.0001	-4	1	-4	0.0001	0.00005
0.00	0	(0)	3	-2	0.01	0.005

B.6.2.2 Краткая литеральная форма

Помимо общей литеральной формы типа PPD<T>, для типа данных PPD<REAL> определена краткая литеральная форма. Она определена таким образом, что значение свойства стандартного отклонения PPD. standardDeviation может быть выражено в терминах наименьшей значащей цифры мантиссы. Этот литерал определен как расширения литерала типа данных REAL:

```
PPD<REAL>.literal ST {
  PPD<REAL> mantissa
       : REAL.mantissa "(" type QTY ")"  { ((T)$).equal($1);
$.distributionType.equal($3);
$.standardDeviation.equal($4); }
       | REAL.mantissa                   { $.equal($1);
$.distributionType.equal($3);
$.standardDeviation.equal(
$1.leastSignificantDigit.times(0.5)); };
  CS type : ST                           { $.value.equal($1);
$.system.equal(2.16.840.1.113883.5.1019); };
};
```

Пример — Строка «1.23e-3 (U5e-6)» общей литеральной формы обозначает равномерное распределение (PPD.distributionType = «U») с центром 1.23 × 10^{-3} и стандартным отклонением PPD. standardDeviation, равным 5 × 10^{-6}. Краткая литеральная форма этого же значения имеет вид «1.230(U5) e-3».

B.6.3 Тип данных распределения вероятности физических величин PPD<PQ> (специализация типа данных PPD)

Тип данных PPD<PQ> строится из типа данных PPD. Но поскольку свойство единиц измерения PQ.unit может быть выведено из границ интервала, то типу данных PPD<PQ> можно придать дополнительную семантику и определить для него отдельную литеральную форму. Для этого тип данных PPD<PQ> надо рассматриваться как распределение вероятности вещественных значений с одной единицей измерения.

```
type ParametricProbabilityDistribution<PQ> alias PPD<PQ> {
  PPD<REAL> value;
  CS  unit;
};
```

Единица применяется как к среднему значению, так и к стандартному отклонению PPD.standardDeviation.

```
invariant(PPD<PQ> x)
    where x.nonNull {
  x.value.nonNull;
  ((REAL)x.value).equal(((PQ)x).value);
  x.unit.equal(((PQ)x).unit);
  x.value.standardDeviation.equal(x.standardDeviation.value);
  x.standardDeviation.unit.equal(x.unit);
};
```

B.6.3.1 Краткая литеральная форма

Краткая литеральная форма типа данных PPD<PQ> определяется на основе краткой литеральной формы типа данных PPD<REAL>, в которую подставляется значение физической величины (тип данных REAL). Этот литерал определяется как расширение литерала типа данных PQ.

```
PPD<PQ>.literal ST {
  PPD<PQ> : PPD<REAL> " " unit  { $.value.equal($1);
                                  $.unit.equal($3); }
};
```

Примеры — Строка «1.23e-3 m (N5e-6 m)» общей литеральной формы описывает нормальное распределение длины с центром 1.23 × 10^{-3} м и стандартным отклонением PPD.standardDeviation, равным 5 × 10^{-6} м. Краткая литеральная форма этого же значения имеет вид «1.230(N5)e-3 m». Допустима также форма «1.23e-3(N0.005e-3) m»; она представляет собой краткую литеральную форму типа данных PPD<PQ>, скомбинированную с общей литеральной формой типа данных PPD<REAL>.

B.6.4 Тип данных распределения вероятности моментов времени PPD<TS> (специализация типа данных PPD)

```
type ParametricProbabilityDistribution<TS> alias PPD<TS>;
```

Тип данных PPD<TS> полностью определяется специализируемым типом данных. Свойство стандартного отклонения PPD.standardDeviation имеет тип данных, равный значению свойства TS.diffType, а именно, длительность (тип данных PQ с размерностью времени).

B.6.4.1 Преобразование значения типа TS в значение типа PPD<TS>

При преобразовании значения типа TS в значение типа PPD<TS>, стандартное отклонение PPD. standardDeviation вычисляется из порядка значения типа TS и такой точности (число значащих цифр), при которой значение PPD.standardDeviations охватывает максимальный диапазон времени, описываемый не указанными цифрами. Например, в строке «20000609» не указанные цифры относятся к часам и более мелким единицам. В совокупности все эти цифры охватывают длительность 24 ч, поэтому стандартное отклонение PPD.standardDeviation равно 12 ч (от 20000609000000.0000... до 20000609999999.9999... (= 20000610).

Это правило отличается от указанного для типа данных REAL тем, что диапазон неопределенности лежит выше заданного момента времени. Это сделано, опираясь на суждение здравого смысла, согласно которому 9-е июня охватывает все сутки 9-го июня с центром в полдне, а не в полночи.

Приложение С
(справочное)

Словарные домены HL7

Версия: 260-20050603
Ответственная рабочая группа: Vocabulary Work Group, Health Level Seven, Inc.
Сгенерировано: RoseTree 4.2.28, 2009-02-01T11:44:00
Дата последней публикации: 20090201 14:26

С.1 Введение

Примечание для голосования в ИСО

Этот выпуск содержания словаря HL7 предназначен для обеспечения интерпретации стандарта ИСО 27932 — HL7 Clinical Document Architecture, Release 2. Его содержание актуально на момент завершения голосования в комитете HL7 по проекту документа CDA R2.

С.1.1 Обзор

При коммуникации и хранении данных в современном здравоохранении активно используется кодированная информация. В стандартах HL7 соответствующие справочники и классификаторы объединены общим понятием словаря. В этих стандартах определено несколько типов объектов, которые воплощают разные характеристики словаря. В то время как другие элементы стандартов HL7 в основном касаются структуры, в словарях основной акцент делается на содержание.

В соответствии с философией стандартов Версии 3, состоящей в последовательном ограничении абстрактной информационной модели, наименее ограниченной категорией словаря является словарный домен. [Пересмотреть словарь терминов и его связи]: словарный домен HL7 представляет собой именованную категорию сходных понятий (семантический тип), каждое из которых связано с одним или несколькими кодированными элементами. Словарные домены существуют в силу потребности ограничить назначение кодированного элемента, откладывая его привязку к конкретной кодированной терминологии на поздние этапы процесса разработки или реализации стандарта. Таким образом, словарные домены не зависят от какого-либо конкретного словаря терминов или системы кодирования.

Иерархическая категоризации словарных доменов позволяет далее ограничивать широту семантической категории, охватываемой словарным доменом. Такие ограниченные домены называются «поддоменами». Их применение позволяет далее специализировать (ограничить) значения, включенные в домен.

Список значений, включенных в домен понятий или в поддомен называется набором значений (value set). Набор значений состоит из одного или нескольких кодированных понятий. Коды этих понятий могут передаваться в значениях кодированного типа данных, передаваемых в сообщениях стандарта HL7, версии 3. Ассоциирование набора значений с данным словарным доменом понятий или поддоменом называется «связыванием». В разных обстоятельствах с одним и тем же словарным доменом могут быть связаны разные наборы значений.

Код понятия уникален только в определенном контексте. Например, в одном контексте код «М» может означать мужской пол (male), а в другом служить признаком состояния в браке (married). Контекст, в котором определено понятие, называется системой кодирования. В домене медицинских терминов уже существует большое число систем кодирования, например, МКБ-9, SNOMED-CT и т. д. По возможности в стандартах HL7 используются существующие коды понятий, а не конструируются новые.

В тот момент, когда разработчики сообщений и те, кто применяет эти сообщения на практике, примут решение об использовании конкретной терминологии для представления смыслового содержания кодированного элемента, набор значений становится связанным с кодированным элементом.

Словарные таблицы, определенные в стандарте HL7 для кодированных атрибутов классов, содержатся в словарном хранилище HL7, из которого могут извлекаться различные представления его содержания, позволяющие получить списки словарных значений, предназначенные для использования в Эталонной информационной модели HL7 RIM (Reference Information Model). Эти представления имеют табличный формат. К ним относятся словарные домены HL7, системы кодирования (как те, что ведутся комитетом HL7, так и внешние системы, используемые в стандартах HL7), наборы значений, определенные в стандартах HL7, а также таблицы перекрестных ссылок между словарными доменами и кодированными атрибутами.

В повествовательной части содержания модели RIM каждому кодированному атрибуту присвоены имя словарного домена и соответствующий квалификатор расширяемости. Эта спецификация указана в первой строке описания атрибута и имеет следующий формат:

«Словарный домен: «MyConceptDomain» (CWE)»

В тексте модели RIM имя словарного домена является гиперссылкой на соответствующую таблицу с описанием этого домена.

С.1.2 Содержание

Ниже представлено описание словаря, разработанного и привязанного к моделям. Из хранилища словаря HL7 извлечены три основных группы данных. Каждая группа состоит из индекса, элементы которого связаны с конкретными записями словаря. Каждая из этих записей представлена в табличной форме, специфичной для соответствующей словарной единицы. Каждый из приведенных ниже разделов содержит подраздел руководства, помещенный под индексом, детально описывающий столбцы таблиц, а также ключевые слова и соглашения по представлению информации, используемые для таблицы данной группы. Эти три раздела индексов описывают следующие группы данных:

- словарные домены HL7. Содержание этого раздела организовано в алфавитном порядке по имени домена. Указано также, имеются ли специализации домена в виде одного или нескольких поддоменов;
- вспомогательные системы кодирования HL7. Содержание этого раздела организовано в алфавитном порядке по имени системы кодирования. Указано также, является ли система кодирования «структурной»;
- наборы значений HL7. Существует большое число наборов значений, определенных в словаре HL7, и организация этого раздела отражает попытку улучшить представление простого алфавитного списка тысяч имен, многие из которых не являются простыми мнемоническими. Большинство наборов значений произведено исключительно или в основном от одной системы кодирования, поэтому индекс содержания раздела содержит список соответствующих систем кодирования, организованный в алфавитном порядке по имени системы кодирования. Если набор значений включает в себя коды из нескольких систем кодирования, то получить к нему доступ можно из записи индекса с именем любой из этих систем.

В каждом из этих индексов обеспечивается переход по гиперссылке на таблицу данных, содержащую информацию о соответствующем элементе индекса.

С.2 Словарные домены HL7

С.2.1 Индекс словарных доменов

Таблица С.1 содержит ручной «индекс» словарных доменов, определенных в стандартах HL7. В ней перечислены только домены верхнего уровня. Многие словарные домены являются специализациями этих доменов и могут быть найдены по гиперссылке от их предшественника. Все словарные домены, имеющие специализации, выделены ниже в таблице полужирным шрифтом. Более специфичные списки находятся в отдельном файле. Для каждого словарного домена в этом списке указаны:

- его текстовое определение;
- его специализации и генерализации, если таковые имеются;
- атрибуты модели RIM, для которых этот домен служит ограничением;
- наборы значений, основанные на этом домене.

Таблица С.1 — Индекс словарных доменов

AcknowledgementCondition	EntityCode	ProcessingID
AcknowledgementDetailType	EntityDeterminer	ProcessingMode
AcknowledgementType	EntityStatus	QueryPriority
ActClass	Ethnicity	QueryResponse
ActCode	HL7StandardVersionCode	QueryStatusCode
ActMood	HumanLanguage	Race
ActPriority	LanguageAbilityMode	RelationalOperator
ActRelationshipCheckpoint	LanguageAbilityProficiency	RelationshipConjunction
ActRelationshipJoin	ManagedParticipationStatus	ReligiousAffiliation
ActRelationshipSplit	MaritalStatus	ResponseLevel
ActRelationshipSubset	ModifyIndicator	ResponseModality
ActRelationshipType	NullFlavor	RoleClass
ActSite	ObservationInterpretation	RoleCode
ActStatus	ObservationMethod	RoleLinkType
AdministrativeGender	OrganizationIndustryClass	RoleStatus
CommunicationFunctionType	ParticipationFunction	RouteOfAdministration

Окончание таблицы С.1

Confidentiality	ParticipationMode	SQLConjunction
ContextControl	ParticipationSignature	Sequencing
EncounterDischargeDisposition	ParticipationType	TargetAwareness
EntityClass	ProcedureMethod	

С.2.2 Руководство по таблицам словарных доменов

По каждой ссылке в приведенном выше индексе можно перейти к таблице, описывающей соответствующий домен понятий. Для каждого домена понятий сгенерирована одна отдельная таблица. В таблице домена понятий одна строка повторяет имя домена, указанное в индексе, и для каждого поддомена, если такие существуют, приведены отдельные строки. В тех случаях, когда для дальнейшего ограничения поддомена определен дополнительный поддомен, эта практика продолжается для более глубоких уровней вложенности. Столбцы таблицы домена содержат информацию о значениях, входящих в домен; в приведенном ниже описании эти столбцы пронумерованы. Заголовки столбцов показаны в скобках.

1 Уровень (Lvl)

В этом столбце указан уровень поддомена, описываемого данной строкой. В первой строке указан уровень поддомена «0», означающий, что эта строка соответствует родительскому домену, указанному в индексе доменов понятий. Уровнем «1»' обозначен поддомен родительского поддомена, уровнем «2» — поддомен этого поддомена и т. д.

2 Имя и связь (Concept Domain Name Value Set Binding)

В этом столбце указано имя домена понятия и его контексты связей, если таковые имеются. Имя набрано обычным текстом, а имя набора значений — курсивом. В скобках непосредственно за именем набора значений указаны контексты связи, если таковые имеются. Если для набора значений домена контекст не классифицирован, то используется обозначение (nos); оно служит указанием, что связь все еще обсуждается в уполномоченной рабочей группе и пока еще не включена в общую совокупность связей. Имя набора значений представляет собой гиперссылку на строку набора значений, включенную в таблицу наборов значений. Обратите внимание, что для некоторых записей объектный идентификатор (ОИД) набора значений заключен в квадратные скобки; это показывает, что для этого набора значения еще не определены. Обратите также внимание, что в большинстве из этих случаев имя набора значений совпадает с именем домена понятия.

3 Связь с доменом атрибута (атрибут модели RIM)

В этом столбце указан атрибут модели RIM, тип данных и квалификатор расширяемости, связанные с доменом понятий. Имя атрибута модели RIM представляет собой гиперссылку на описание данного атрибута в модели RIM, приведенное в настоящей спецификации. Первый элемент в скобках означает тип данных, которым в каждом случае является тип данных «CD» или одна из его специализаций, предназначенные для описания кодированных данных в Версии 3. Второй элемент в скобках означает квалификатор расширяемости («CNE» или «CWE»). Эти два элемента разделены символом косой черты («/»). Обратите внимание, что в большинстве случаев атрибут указан только для домена понятия уровня «0»; большинство поддоменов получены в процессе ограничения модели и связаны с атрибутами, которые является ограничениями атрибута модели RIM. Также учтите, что в большинстве случаев домен понятий связан только с одним атрибутом модели RIM.

4 Документация (Definition/Description)

Этот столбец содержит все документирование, относящееся к данному домену понятий. Оно включает в себя определение домена, а также обсуждение или описание соответствующего класса понятий. Обратите внимание, что в некоторых случаях указано только «Требуется добавить описание», что понятно без разъяснений. В других случаях ячейка таблицы в этом столбце пуста, что означает отсутствие документации для данного домена понятий. Описание может содержать ограничения, примеры и другие компоненты документации, полезные для понимания значения и использования данного домена понятий.

С.3 Системы Кодирования, поддерживаемые в стандартах HL7

С.3.1 Индекс систем Кодирования

Таблица С.2 содержит ручной «индекс» систем кодирования, поддерживаемых в стандартах HL7. Ее строки упорядочены по именам систем кодирования. В ней показаны все системы кодирования, идентифицированные в словарном хранилище HL7. Некоторые из этих систем разработаны и сопровождаются комитетом HL7, для них приводится полное содержание. Их имена выделены в таблице полужирным шрифтом. Несколько внешних систем кодирования, содержание которых распространяется комитетом HL7 по соглашениям с их авторами, также полностью приведены. Каждая из этих внешних систем кодирования представлена в отдельном html-файле. Доступ к ней можно получить по гиперссылке с ее именем. Оставшиеся (внешние) системы кодирования перечислены в отдельном документе, их содержание не приводится. Определения различных систем кодирования могут содержать ссылки на любую из этих систем кодирования, независимо от того, находится ли она в ведении комитета HL7 или иной организации (см. следующий подраздел).

Таблица С.2 — Индекс систем кодирования

Acknowledgement Detail Type	Confidentiality	ProcessingMode
AcknowledgementCondition	ContextControl	QueryPriority
AcknowledgementType	EntityClass	QueryResponse
ActClass	EntityCode	QueryStatusCode
ActCode	EntityDeterminer	RelationalOperator
ActMood	EntityStatus	RelationshipConjunction
ActRelationshipCheckpoint	HL7StandardVersionCode	ResponseLevel
ActRelationshipJoin	ManagedParticipationStatus	ResponseModality
ActRelationshipSplit	ModifyIndicator	RoleClass
ActRelationshipSubset	NullFlavor	RoleCode
ActRelationshipType	OrderableDrugForm	RoleLinkType
ActStatus	ParticipationType	RoleStatus
CommunicationFunctionType	ProcessingID	Sequencing

С.3.2 Руководство по таблицам систем Кодирования

Каждая из гиперссылок в приведенном выше индексе указывает на таблицу с содержанием систем кодирования, архивированным в словарном хранилище HL7. Строки индекса упорядочены по именам систем кодирования. Содержание систем кодирования, имена которых выделены полужирным шрифтом, берется из словарного хранилища HL7. Это означает, что эти системы кодирования либо ведутся и сопровождаются комитетом HL7 с использованием процессов гармонизации и утверждения, либо зеркально отражены в словарном хранилище HL7 для удобства использования. К другим системам кодирования относятся те, которые ведутся внешними организациями, и часть их содержания используется здесь для различных связей в моделях и доменах понятий. Содержание таких систем не отражено в словарном хранилище HL7; для получения копии содержания следует обратиться к издателям соответствующей терминологии.

Каждая таблица с содержанием системы кодирования содержит четыре столбца; имена столбцов и детали их содержания описаны ниже; имена столбцов указаны в скобках. Каждая строка таблицы описывает один код понятия (кодированный термин).

1 Уровень иерархии и возможность выбора (Lvl-Typ)

Первая часть данных в этом столбце представляет собой целое число, указывающее глубину иерархии кода, описанного в данной строке. За ним следует дефис и одна из следующих букв верхнего регистра:

- L (Leaf — лист); термин, не имеющий потомков в иерархии специализации, который может быть выбран в инструментальных средствах HL7 и поэтому рассматривается как лист;
- A (Abstract — абстрактный); термин, имеющий потомков в иерархии специализации, но который сам по себе не может быть выбран в инструментальных средствах HL7 и поэтому рассматривается как абстрактный;
- S (Specializable — специализируемый); термин, имеющий потомков в иерархии специализации, который может быть выбран в инструментальных средствах HL7 и поэтому рассматривается как специализируемый.

2 Код/понятие и ссылка на главный код набора значений (Concept Code Head Code-defined Value Set)

Этот столбец содержит строковое значение кода понятия. В тех случаях, когда этот код используется как «главный код» набора значений, определенного как «главный код и все его потомки», то имя этого набора значений указано в той же ячейке, что и код термина, только набрано курсивом и помещено под кодом. Имя набора значений является гиперссылкой на соответствующую таблицу с описанием набора значений. Для удобства восприятия иерархии на каждом уровне вложенности коду предшествует точка (символ с кодом 0x2E).

3 Имя кодированного понятия, предназначенное для вывода на экран (Print Name)

Это строка, ассоциированная с термином кодированного понятия и используемая как основное обозначение понятия на английском языке.

4 Определение кодированного понятия, свойства и отношения (Coded Concept Definition, Properties and Relationships)

В этом столбце указаны определение кодированного понятия и все присвоенные ему свойства и связи. Свойства представляют собой дополнительные значения, которые описывают или классифицируют понятие. Отношения определяют семантические связи этого понятия с другими понятиями. Наиболее распространенными отношения понятий являются «Specializes» (является специализацией) и «Generalizes» (является генерализацией), определяющие иерархии подтипов в системе кодирования. Учтите, что многие понятия не имеют таких определений.

С.4 Наборы значений HL7

С.4.1 Индекс наборов значений

Таблица С.3 содержит ручной «индекс» наборов значений, определенных в стандартах HL7. Поскольку общее число таких наборов значений превышает 1800, и при этом они не выстроены в иерархию и не имеют иного естественного порядка, то они произвольно сгруппированы по системам кодирования, на которых они основаны. (Некоторые наборы значений основаны на нескольких системах кодирования и поэтому появятся в списках столько раз, на скольких систем кодирования они основаны).

Группировки систем кодирования отражают также способ распределения определений наборов значений по html-файлам. Первая таблица этой группы представляет собой список тех систем кодирования, на основе которых определен хотя бы один набор значений. Из нее можно перейти в документ, перечисляющий конкретные наборы значений, основанные на данной системе кодирования.

Таблица С.3 — Системы кодирования, на основе которых определены наборы значений

Acknowledgement Detail Type	Confidentiality	ProcessingMode
AcknowledgementCondition	ContextControl	QueryPriority
AcknowledgementType	EntityClass	QueryResponse
ActClass	EntityCode	QueryStatusCode
ActCode	EntityDeterminer	RelationalOperator
ActMood	EntityStatus	RelationshipConjunction
ActRelationshipCheckpoint	HL7StandardVersionCode	ResponseLevel
ActRelationshipJoin	ManagedParticipationStatus	ResponseModality
ActRelationshipSplit	ModifyIndicator	RoleClass
ActRelationshipSubset	NullFlavor	RoleCode
ActRelationshipType	OrderableDrugForm	RoleLinkType
ActStatus	ParticipationType	RoleStatus
CommunicationFunctionType	ProcessingID	Sequencing

С.4.2 Руководство по таблицам наборов значений

Для просмотра наборов значений предусмотрено навигационное средство в виде таблицы 0, в которой перечислены системы кодирования, содержание которых используется в наборах значений. Имена этих систем представляют собой гиперссылки, по которым можно перейти к вспомогательным файлам, содержащим все определения наборов значений, основанных на конкретной системе кодирования. Для системы кодирования, по которой определен единственный набор значений, страница, открываемая по гиперссылке, содержит информационную таблицу, определяющую этот набор значений. Если система кодирования используется в нескольких наборах значений, то в верхней части страницы, открываемой по гиперссылке, приводится таблица с упорядоченным по алфавиту списком имен наборов значений, использующих эту систему кодирования. Эти имена представляют собой гиперссылки, по которым можно перейти к таблице с определением соответствующего набора значений.

Блок информации, описывающий набор значений, начинается с имени набора значений, за которым в квадратных скобках указан его ОИД. Затем приводится краткая сводка систем кодирования, значения которых включены в данный набор значений. Эта сводка пронумерована; каждому имени дополнительной системы кодирования, часть содержания которой включена в набор значений, присвоен порядковый номер. За этим номером следует имя системы кодирования, являющееся гиперссылкой на таблицу с описанием системы кодирования, переход к которой обеспечен также из предшествующего раздела. После имени системы кодирования в квадратных скобках указан ее ОИД.

Затем перечисляются контекстные связи (если таковые имеются). Раздел контекстных связей начинается строкой «Context Bindings to concept domain(s):» (контекстные связи с доменами понятий), под которой следует перечисление связей (по одной связи в строке). Каждое описание связи содержит имя домена понятия (являющееся гиперссылкой на соответствующую таблицу с описанием домена понятия), за которым следуют указание «in Realm:» (в среде) и имя той среды, для которой определена эта связь. В случае, если среда не классифицирована, то есть решение о связи все еще рассматривается уполномоченной рабочей группой, вместо имени среды указывается слово «undetermined» (неопределенная).

Ниже указан состав описания набора значений, представляемого его авторами (некоторые наборы значений не имеют описаний, которые в будущем должны быть добавлены).

Вслед за описанием набора значений приводится таблица, содержащая определение его содержания. У этой таблицы пять столбцов, показывающих информацию о каждом утверждении, являющемся компонентом определения набора значений и занимающем отдельную строку. Определение содержания представлено в виде группировок, которые должны сделать его более понятным. Это определение состоит из списка «блоков содержания», каждый из которых имеет один и тот же тип формата. Блок содержания может также содержать дополнительный блок содержания с тем же типом формата, который, в свою очередь, может содержать другой блок содержания; таким образом, определение является рекурсивным, как и модель набора значений, которой оно должно соответствовать. Правильно определенный набор значений не имеет циклов в своей рекурсивной структуре.

Таблица, содержащая определение набора значений, аккуратно отображает эту рекурсивную структуру, но при этом, естественно, подвержена ограничениям, присущим представлению рекурсивных структур в виде линейной таблицы. Для некоторой компенсации этих ограничений используются специальные символы, отступы и гиперссылки. Ниже описаны ключевые слова, специальные символы, определения и другая информация, помещаемая в каждом из этих столбцов. Номера идентифицируют положение столбца в таблице, а заголовки столбцов указаны в скобках.

1 Определение глубины вложения (Lvl)

Это целое число, указывающее глубину вложения утверждения, описывающего компонент набора значений. Оно помогает визуализировать такие конструкции, как вложенные наборы значений, включаемые транзитивные замыкания, операторы в конкретном блоке содержания и т. д. Нумерация глубины вложения всегда начинается с нуля, и соответствующая строку можно рассматриваться как «корень» определения набора значений. Эти номера можно использовать для идентификации каждого определения «блока содержания». Но эти номера обозначают уровень вложения рекурсивной структуры и ссылок; конкретный блок содержания распространяется на несколько номеров, поэтому его уровень вложения следует рассматривать как «больше nn», а не равным конкретному номеру.

2 Тип содержания (Content Type)

В этом столбце указан тип утверждения, описывающего компонент определения набора значений. В нем могут быть указаны ключевые фразы, принадлежащие определенному набору и указывающее, каким образом содержание, объявленное в остальных столбцах, включается в набор значений. Некоторые из этих фраз являются флагами в утверждении, а некоторые — ссылками на множество включаемых кодов. Обратите внимание, что каждому элементу этого столбца предшествует одна или несколько точек (‘.’), показывающих глубину вложения утверждения, по одной точке на каждый уровень. В этом столбце могут быть указаны следующие ключевые фразы:

- content (содержание): это ключевой элемент набора значений; для тех наборов, что основаны на нескольких системах кодирования, в этом элементе указана первая из систем, участвующих в определении набора значений. Он всегда занимает первую строку таблицы и имеет уровень глубины 0 (нуль). Только одна строка таблицы может быть помечена таким ключом. Следующие строки представляют собой определение первого раздела содержания. Первая строка с таким ключом служит заголовком первого раздела содержания. Учтите, что определение набора значений рекурсивно, поэтому блоки содержания могут входить в состав других блоков. Поэтому блок содержания, помеченный этим ключевым словом, является определением всего набора значений, все остальные определения «вложены» в него. Каждый конкретный экземпляр содержания может содержать дочерние элементы только одного из следующих пяти типов, и только один из этих типов, а именно, codeBasedContent, допускает повторы. (Имейте в виду, что этот ограниченный список пяти подтипов распространяется также на элементы типа unionWithContent. Типы intersectionWithContent и excludedContent обсуждаются ниже при описании типа combinedContent.);

- codeBasedContent (содержание, основанное на коде): этот подтип указывает, что в набор значений должен быть включен конкретный код из системы кодирования. Идентификация системы кодирования, из которой берется этот код, указана в последней строке из числа тех, что имеют номер уровня на единицу меньше уровня текущей строки (указывающей «текущее» множество содержания системы кодирования); определение системы кодирования наследуются всеми строками блока содержания. Этот конкретный код указан в столбце, озаглавленном «Primary Reference» (основная ссылка). Блок содержания может включать в себя произвольное число строк с ключевым словом codeBasedContent, предназначенных для явного включения отдельного кода в блок содержания. В этих строках указана точная идентификация кода. Хотя в каждой из этих строк указан один код из одной системы кодирования, указание в столбце «Qualifiers/Identifiers» (квалификаторы/идентификаторы) свойства TransitiveClosure (транзитивное замыкание) может привести к заданию для одной строки коллекции кодов, являющейся поддеревом в иерархии кодов системы кодирования;

- propertyBasedContent (содержание, основанное на свойстве): этот подтип служит заголовок для произвольного числа строк, каждая из которых представляет собой спецификацию включения или исключения, основанную на свойствах понятий или на свойствах кода, определенных для системы кодирования в словарном хранилище. Для каждой из этих строк в данном столбце Content Type будет указано объявление элемента; такими объявлениями служат:

- includeWithConceptProperty (включить по свойству понятия): в набор значений включается множество кодов, получающееся с помощью выборки по значению свойства понятия. Идентификация этого свойства указана в столбце «Primary Reference». Условие выборки указано в столбце «Qualifiers/Identifiers»;

- excludeWithConceptProperty (исключить по свойству понятия): из набора значений исключается множество кодов, получающееся с помощью выборки по значению свойства понятия. Идентификация этого свойства указана в столбце «Primary Reference». Условие выборки указано в столбце «Qualifiers/Identifiers»;

- includeWithCodeProperty (включить по свойству кода): в набор значений включается множество кодов, получающееся с помощью выборки по значению свойства кода. Идентификация этого свойства указана в столбце «Primary Reference». Условие выборки указано в столбце «Qualifiers/Identifiers»;

- excludeWithCodeProperty (исключить по свойству кода): из набора значений исключается множество кодов, получающееся с помощью выборки по значению свойства кода. Идентификация этого свойства указана в столбце «Primary Reference». Условие выборки указано в столбце «Qualifiers/Identifiers».

Имейте в виду, что точный синтаксис условий выборки и аргументов, используемых в таких определениях, будет предложен позже (в настоящее время в словарном хранилище HL7 нет ни одного набора значений, в котором использовался бы такой подтип утверждения);

- codeFilterContent (содержание на основе фильтрации кодов): этот подтип идентифицирует строку, в которой указано выражение, задающее коллекцию кодов. Это выражение указано в столбце «PrimaryReference». В блоке содержания может присутствовать только одна такая строка. Определение фильтра указано в этой же строке и представляет собой выражение в той форме, которая поддерживается системой кодирования. Оно показано в столбце «Primary Reference», а за ним в скобках указан язык выражения. Если выражение, сохраненное в базе данных, не является допустимым для одного из обозначенных языков, то оно будет показано как «комментарий» на одном из языков и ему будет предшествовать фраза «invalid expression» (недопустимое выражение». В этом случае будет также зарегистрирован открытый вопрос по отношению к родительскому типу содержания;

- valueSetReference (ссылка на набор значений): этот подтип указывает, что данное определение ссылается на другой набор значений. Имя включаемого набора значений указано в столбце «Primary Reference», ОИД этого набора значений, заключенный в квадратные скобки, приводится в столбце «Qualifiers/Identifiers». Имя включаемого набора значения представляет собой гиперссылку, по которой можно перейти к таблице с его описанием, находящейся в том же файле, что и данный набор значений. Имейте в виду, что иногда имя включаемого набора значений не является гиперссылкой; это означает, что включаемый набор имеет пустое содержание (висячая ссылка, неполное определение или иной вид ошибки). В этих случаях для включаемого набора значений нет и таблицы с определением содержания;

- combinedContent (комбинированное содержание): этот подтип означает, что следующие за ним строки являются определениями дополнительного содержания, включаемого в набор значений. Каждая из строк, следующая за строкой этого подтипа, должна содержать утверждение одного из нескольких различных типов, которое должен быть применено для получения списка кодированных понятий, включаемых в набор значений. Каждая следующая строка с номером уровня, превышающим номер уровня строки с этим ключевым словом, содержит отдельные утверждения, определяющие содержание, являющееся частью данного блока содержания. Оно должно быть скомбинировано с кодами из заданной системы кодирования. В блоке содержания может быть не более одной строки такого подтипа; она служит заголовком блока содержания.

Когда в набор значений должен быть включен дополнительный блок содержания, начинающийся с ключевого слова «combinedContent», то во втором столбце могут быть указаны дополнительные ключевые слова, указывающие, каким образом конкретный блок содержания может сочетаться с набором значений. Строка, непосредственно следующая за строкой с ключевым словом «combinedContent», должна содержать одно из следующих ключевых слов, указывающих, каким образом специфицируемое в ней содержание будет сочетаться с набором значений. Учтите, что в блоке содержания, озаглавленном строкой с ключевым словом «combinedContent», каждой из следующих строк, определяющих содержание и имеющих подтипы «codeBasedContent», «valueSetReference», «propertyBasedContent», «codeFilterContent» или описывающих вложенный блок «combinedContent», должна предшествовать строка, которая в первом столбце содержит одно из трех следующих ключевых слов, указывающих, каким образом содержание, определенное следующей строкой, должно сочетаться для получения множества кодов, образующих результирующий набор значений:

- unionWithContent (объединение с содержанием): это ключевое слово означает, что содержание, определяемое следующей строкой, должно быть скомбинировано с предыдущим содержанием путем объединения. Термин «объединение» используется в теоретико-множественном смысле, поскольку набор значений обычно можно трактовать как «множество» в математике. Система кодирования, из которой берется содержание, обязательно должна быть указана в столбце «Code System» одной из этих строк;

- intersectionWithContent (пересечение с содержанием): это ключевое слово означает, что содержание, определяемое следующей строкой, должно быть скомбинировано с содержанием, определяемом предыдущими строками, путем пересечения;

- excludeContent (исключить из содержания): это ключевое слово используется, когда надо исключить часть содержания из определяемого набора значений. Чаще всего это имеет место, когда включается «дерево» кодов (например, с помощью ключевого слова codeBasedContent и транзитивного замыкания), но вершину этого дерева включать не надо. Это ключевое слово можно также использовать в ситуации, когда произвольный список кодов должен быть исключен из коллекции кодов, которое было включено ранее с помощью операции объединения.

3 Система кодирования (Code System)

В этом столбце указано имя системы кодирования, из которой должны быть взяты коды для блока содержания. Он заполняется в каждой строке с ключевым словом «content» или «unionWithContent» и должен быть пустым для других строк. Имя системы кодирования является гиперссылкой, по которой можно перейти к табличному описанию системы кодирования, находящемуся в соответствующем файле (см. предыдущий подраздел).

4 Специфичное вложенное содержание (Primary Reference)

В этом столбце указан точный элемент, который должен использоваться для определения содержания набора значений. Для строк с ключевым словом «codeBasedContent» в этом столбце указано значение кода из активной системы кодирования. Если оно содержит имя кода из системы кодирования, являющейся текущей для данного блока содержания, то это имя является гиперссылкой на конкретную строку с определением этого кода в табличном описании этой системы кодирования. Для строк с ключевым словом «valueSetReference» этот столбец содержит имя набора значения, являющееся гиперссылкой на соответствующую строку в данном разделе таблиц с наборами значений. Для других типов строк этот столбец может содержать имя свойства понятия или свойства кода. Для строк, не имеющих ссылок на словарный объект (например, строк с ключевым словом «combinedContent») или ссылающихся только на систему кодирования (указываемую в столбце «Code System»), этот столбец должен оставаться пустым.

5 Дополнительные данные (Qualifiers/Identifiers)

Этот столбец содержит дополнительную информацию, используемую для определения набора значений. Обычно в нем указаны выражения и аргументы, используемые в сочетании с определением содержания, указанным в столбце «Content Type», но в этом столбце могут использоваться специфичные ключевые слова и фразы, представляющие определенные флаги управления, выставляемые в определении набора значений, а также некоторые другие типы определяющих выражений, которые могут быть добавлены к определению набора значений его авторами. Этот столбец может содержать ОИД, заключенный в квадратные скобки и обычно присутствующий только в строках с ключевым словом «valueSetReference», хотя в большинстве случаев этот столбец не имеет структурированного содержания. Одной из специфичных ключевых фраз является:

Relationship: Generalizes With: TransitiveClosure (отношение: генерализует транзитивное замыкание)

Она указывает, что для сбора совокупности кодов, задаваемой данной строкой, должен быть выполнен транзитивный обход иерархии, начиная с кода, указанного в столбце «Primary Reference». Эта ключевая фраза используется только для строк с ключевым словом «codeBasedContent», она включается в ту же строку, что и заданный кодированный термин, и означает, что набор значений включает в себя все понятия, являющиеся потомками этого кодированного термина.

С.5 Запрещенные элементы словаря HL7

С.5.1 Список запрещенных элементов

Процесс ведения содержания словаря HL7 предусматривает две стадии изъятия (в действительности удаления) таких элементов, как домены понятий, системы кодирования, понятия, включенные в систему кодирования, и наборы значений.

Если известно или предполагается, что элемент использовался для определения формальной спецификации HL7, например, типа данных, документа или сообщения, то первой стадией его удаления является «запрещение» его дальнейшего использования для составления новых спецификаций. Тем самым пользователям запрещенного содержания дается предупреждение о необходимости изменения их спецификаций, позволяющего в будущем не зависеть от этого содержания.

На второй стадии запрещенное содержание фактически изымается или удаляется из определенного содержания словаря.

Чтобы дать точное представление о действующем содержании словаря, рабочие группы Vocabulary Work Group и Publishing Work Group договорились об исключении показа запрещенного материала (за исключением запрещенных понятий) в основном содержании словаря. Кроме того, чтобы это исключение было наглядным, они договорились предоставлять таблицу всех запрещенных доменов понятий, систем кодирования и наборов значений с указанием версии словаря, в которой начал действовать запрет. Эта таблица имеет следующий вид:

Deprecated Vocabulary Elements	
Element Name	Release First Deprecated

Приложение D
(справочное)

Опечатки в АКД, выпуск 2 (по состоянию на 20 января 2009 г.)

Подпункт 5.1.2.2 «А» в АКД

Пример 2. Иерархия приведена неправильно. Четыре нижних маркера списка должны быть перемещены на один уровень вправо.

Пункт 5.1.3 Соответствие стандарту АКД

Разъяснения к соответствию.

Документом, соответствующий стандарту АКД, считается тот документ, который как минимум соответствует схеме АКД-документов при условии, что все расширения удалены из документа, и в котором использование кодированных данных ограничено значениями, разрешенными для соответствующих словарных доменов. Однако компьютерная программа не проверить каждый аспект соответствия. Цель данного раздела — выделить те аспекты соответствия стандарту АКД, которые не могут быть проверены автоматически, в особенности те, что связаны с требованиями человекочитаемости АКД-документов.

Подраздел 5.3 Передача АКД-документов в сообщениях HL7

Таблица 4. Отображение полей сообщения HL7 V3 Medical Records на компоненты АКД-документа (должно обновляться по мере утверждения спецификации V3 Medical Records).

Подраздел 5.4 Модель АКД R-MIM

(Техническая коррекция) На диаграмме Visio не показан словарный домен, присвоенный атрибуту Observation. interpretationCode (в иерархическом описании он показан).

ConfidentialityCode — начальная буква «с» должна быть строчной — Grahame Grieve 04:49, 7 июля 2008 г. (UTC).

Подпункт 5.4.3.5 Повествовательный блок раздела

(Техническая коррекция) Для улучшения читаемости и уменьшения неоднозначности фрагмент XML-схемы повествовательного блока [<xs:complexType name=»StrucDoc.Sup» mixed=»true»/>] должен быть заменен на [<xs:element name=»StrucDoc.Sup» type=»xsd:string»/>].

Подпункт 5.4.3.6.5 Класс Organizer (группировка)

Пересмотреть ограничения (в модели RMIM и иерархическом описании) и сделать в тексте следующее пояснение:

Клон класса Organizer может быть источником отношения с компонентами, источником отношения с предусловием или со ссылкой, но не источником отношения entryRelationship (связь между подразделами).

Пункт 5.7.2 Пример АКД-документа

Там, где атрибут Observation.classCode имеет значение "COND", сделайте ссылку на последние рекомендации технического комитета Patient Care, касающиеся присваиванию значений атрибуту Observation.statusCode.

Ошибочный ОИД. Заменить ОИД "2.16.840.1.113883.5.10588" на "2.16.840.1.113883.5.111" в следующем фрагменте:

<healthCareFacility classCode="DSDLOC"> <code code="GIM" codeSystem=" 2.16.840.1.113883.5.10588 " displayName="General internal medicine clinic"/> </healthCareFacility>

Ошибочные коды LOINC. Заменить коды 10153-2, 10223-2, 10157-2 соответственно на следующие: 11348-0 HISTORY OF PAST ILLNESS, 10200-4 PHYSICAL FINDINGS HEART, 10157-6 HISTORY OF FAMILY MEMBER DISEASES.

Ошибочная дата. Заменить дату "200004071530" на "200004071430" в следующем фрагменте:

```
<observation classCode="OBS" moodCode="EVN">
    <code code="251076008" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Cuff blood pressure"/>
    <statusCode code="completed"/>
    <effectiveTime value="200004071430"/>
    <targetSiteCode code="368208006" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Left arm"/>
    <entryRelationship typeCode="COMP">
        <observation classCode="OBS" moodCode="EVN">
          <code code="271649006" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
```

```
CT" displayName="Systolic BP"/>
            <statusCode code="completed"/>
            <effectiveTime value=" 200004071530 --> 200004071430 "/>
            <value xsi:type="PQ" value="132" unit="mm[Hg]"/>
          </observation>
      </entryRelationship>
      <entryRelationship typeCode="COMP">
          <observation classCode="OBS" moodCode="EVN">
            <code code="271650006" codeSystem="2.16.840.1.113883.6.96"
 codeSystemName="SNOMED CT" displayName="Diastolic BP"/>
            <statusCode code="completed"/>
            <effectiveTime value=" 200004071530
--> 200004071430 "/>
            <value xsi:type="PQ" value="86" unit="mm[Hg]"/>
          </observation>
      </entryRelationship>
</observation>
```

Во фрагменте <priorityCode code="PRN"/> должен быть добавлен атрибут codeSystem со значением "2.16.840.1.113883.5.7".

Фрагмент <routeCode code="IPINHL" codeSystem="2.16.840.1.113883.5.112" ... должен быть заменен на <routeCode code="ORINHL" codeSystem="2.16.840.1.113883.5.112" ...

Во фрагменте <act classCode="ACT" moodCode="INT"><id/> следует удалить элемент Act.id.

Оригинал текста взят из ресурса «http://wiki.hl7.org/index.php?title=CDA_R2_Errata.» Для доступа к нему нужны имя пользователя и пароль. Используйте имя «wiki» и пароль «wikiwiki».

Приложение Е
(справочное)

Примеры АКД-документов

Е.1 Пример АКД-документа с таблицей стилей

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="CDA.xsl"?>
<!--
Читателям рекомендуется иметь в виду руководство по реализации «Using SNOMED CT in HL7 Version 3», которое пока что находится на стадии разработки. Это руководство, разрабатываемое совместно комитетом HL7 и Институтом американских патологоанатомов (College of American Pathologists), будет представлено на утверждение в комитет HL7 как справочный документ. Шаблоны использования SNOMED CT, приведенные в настоящем примере, будут переработаны в соответствии с рекомендациями окончательной версии этого руководства.
-->
<ClinicalDocument xmlns=»urn:hl7-org:v3» xmlns:voc=»urn:hl7-org:v3/voc» xmlns:xsi=»http://www.w3.org/2001/XMLSchema-instance» xsi:schemaLocation=»urn:hl7-org:v3 CDA.xsd»>
        <!--
**********************************************************************************************************
  CDA Header
**********************************************************************************************************
-->
    <typeId root="2.16.840.1.113883.1.3" extension="POCD_HD000040"/>
    <templateId root="2.16.840.1.113883.3.27.1776"/>
    <id extension="c266" root="2.16.840.1.113883.19.4"/>
    <code code="11488-4" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC" displayName="Consultation note"/>
    <title>Заключение консультанта клиники Good Health Clinic</title>
    <effectiveTime value="20000407"/>
    <confidentialityCode code="N" codeSystem="2.16.840.1.113883.5.25"/>
    <languageCode code="en-US"/>
    <setId extension="BB35" root="2.16.840.1.113883.19.7"/>
    <versionNumber value="2"/>
    <recordTarget>
        <patientRole>
            <id extension="12345" root="2.16.840.1.113883.19.5"/>
            <patient>
                <name>
                    <given>Генри</given>
                    <family>Левин</family>
                    <suffix>7-й</suffix>
                </name>
                <administrativeGenderCode code="M" codeSystem="2.16.840.1.113883.5.1"/>
                <birthTime value="19320924"/>
            </patient>
            <providerOrganization>
                <id root="2.16.840.1.113883.19.5"/>
            </providerOrganization>
        </patientRole>
    </recordTarget>
```

```
<author>
    <time value="2000040714"/>
    <assignedAuthor>
        <id extension="KP00017" root="2.16.840.1.113883.19.5"/>
        <assignedPerson>
            <name>
                <given>Роберт</given>
                <family>Долин</family>
                <suffix>д.м.н.</suffix>
            </name>
        </assignedPerson>
        <representedOrganization>
            <id root="2.16.840.1.113883.19.5"/>
        </representedOrganization>
    </assignedAuthor>
</author>
<custodian>
    <assignedCustodian>
        <representedCustodianOrganization>
            <id root="2.16.840.1.113883.19.5"/>
            <name>Good Health Clinic</name>
        </representedCustodianOrganization>
    </assignedCustodian>
</custodian>
<legalAuthenticator>
    <time value="20000408"/>
    <signatureCode code="S"/>
    <assignedEntity>
        <id extension="KP00017" root="2.16.840.1.113883.19.5"/>
        <assignedPerson>
            <name>
                <given>Роберт</given>
                <family>Долин</family>
                <suffix>д.м.н.</suffix>
            </name>
        </assignedPerson>
        <representedOrganization>
            <id root="2.16.840.1.113883.19.5"/>
        </representedOrganization>
    </assignedEntity>
</legalAuthenticator>
<relatedDocument typeCode="RPLC">
    <parentDocument>
        <id extension="a123" root="2.16.840.1.113883.19.4"/>
        <setId extension="BB35" root="2.16.840.1.113883.19.7"/>
        <versionNumber value="1"/>
    </parentDocument>
</relatedDocument>
<componentOf>
```

```
        <encompassingEncounter>
            <id extension="KPENC1332" root="2.16.840.1.113883.19.6"/>
            <effectiveTime value="20000407"/>
            <encounterParticipant typeCode="CON">
                <time value="20000407"/>
                <assignedEntity>
                    <id extension="KP00017" root="2.16.840.1.113883.19.5"/>
                    <assignedPerson>
                        <name>
                            <given>Роберт</given>
                            <family>Долин</family>
                            <suffix>д.м.н.</suffix>
                        </name>
                    </assignedPerson>
                    <representedOrganization>
                        <id root="2.16.840.1.113883.19.5"/>
                    </representedOrganization>
                </assignedEntity>
            </encounterParticipant>
            <location>
                <healthCareFacility classCode="DSDLOC">
                    <code code="GIM" codeSystem="2.16.840.1.113883.5.10588" displayName="Клиника общей
внутренней медицины"/>
                </healthCareFacility>
            </location>
        </encompassingEncounter>
    </componentOf>
    <!--
**********************************************************************************************************
Тело АКД
**********************************************************************************************************
-->
    <component>
        <structuredBody>
            <!--
**********************************************************************************************************
 Раздел истории текущего заболевания
**********************************************************************************************************
-->
            <component>
                <section>
                    <code code="10164-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                    <title>История текущего заболевания</title>
                    <text>
                        <content styleCode="Bold">Генри Левин 7-й
                        </content> 67 лет, обратился за дальнейшим лечением от астмы. Заболел, когда был <content
revised="delete">двадцатилетним</content>
                        <content revised="insert">подростком</content>.  В прошлом году был дважды госпитализиро-
ван, и уже дважды в этом году. Последние несколько месяцев не может обходиться без стероидов.
                        </text>
```

```
            </section>
        </component>
        <!--
**********************************************************************************************************
Раздел истории заболеваний
**********************************************************************************************************
-->
        <component>
            <section>
                <code code="10153-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>История заболеваний</title>
                <text>
                    <list>
                        <item>
                            <content ID="a1">Астма</content>
                        </item>
                        <item>
                            <content ID="a2">Гипертензия (детали см. в документе HTN.cda)</content>
                        </item>
                        <item>
                            <content ID="a3">Остеоартрит
                                <content ID="a4"> правого колена</content>
                            </content>
                        </item>
                    </list>
                </text>
                <entry>
                    <observation classCode="COND" moodCode="EVN">
                        <code code="195967001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Asthma">
                            <originalText>
                                <reference value="#a1"/>
                            </originalText>
                        </code>
                        <statusCode code="completed"/>
                        <effectiveTime value="1950"/>
                        <reference typeCode="XCRPT">
                            <externalObservation>
                                <id root="2.16.840.1.113883.19.1.2765"/>
                            </externalObservation>
                        </reference>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="COND" moodCode="EVN">
                        <code code="59621000" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="HTN">
                            <originalText>
                                <reference value="#a2"/>
```

```
                    </originalText>
                </code>
                <statusCode code="completed"/>
                <reference typeCode="SPRT">
                    <seperatableInd value="false"/>
                    <externalDocument>
                        <id root="2.16.840.1.113883.19.4.789"/>
                        <text mediaType="multipart/related">
                            <reference value="HTN.cda"/>
                        </text>
                        <setId root="2.16.840.1.113883.19.7.2465"/>
                        <versionNumber value="1"/>
                    </externalDocument>
                </reference>
                <reference typeCode="XCRPT">
                    <externalObservation>
                        <id root="2.16.840.1.113883.19.1.2005"/>
                    </externalObservation>
                </reference>
            </observation>
        </entry>
        <entry>
            <observation classCode="COND" moodCode="EVN">
                <code xsi:type="CD" code="396275006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Osteoarthritis">
                    <originalText>
                        <reference value="#a3"/>
                    </originalText>
                </code>
                <statusCode code="completed"/>
                <targetSiteCode code="49076000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Knee joint">
                    <originalText>
                        <reference value="#a4"/>
                    </originalText>
                    <qualifier>
                        <name code="78615007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="with laterality"/>
                        <value code="24028007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="right"/>
                    </qualifier>
                </targetSiteCode>
                <reference typeCode="XCRPT">
                    <externalObservation>
                        <id root="2.16.840.1.113883.19.1.1805"/>
                    </externalObservation>
                </reference>
            </observation>
        </entry>
    </section>
```

```
        </component>
        <!--
**********************************************************************************************************
 Раздел лекарственных назначений
**********************************************************************************************************
-->
        <component>
            <section>
                <code code="10160-0" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Медикаментозная терапия</title>
                <text>
                    <list>
                        <item>Теодур, 200 мг два раза в день</item>
                        <item>Альбутерол ингалятор, 2 вдоха, 4 раза в день, по необходимости</item>
                        <item>Преднизон, 20 мг в день</item>
                        <item>Гидрохлортиазид, 25мг в день</item>
                    </list>
                </text>
                <entry>
                    <substanceAdministration classCode="SBADM" moodCode="EVN">
                        <text>Теодур, 200 мг два раза в день</text>
                        <effectiveTime xsi:type="PIVL_TS" institutionSpecified="true">
                            <period value="12" unit="h"/>
                        </effectiveTime>
                        <routeCode code="PO" codeSystem="2.16.840.1.113883.5.112" codeSystemName="RouteOfA
dministration"/>
                        <doseQuantity value="200" unit="mg"/>
                        <consumable>
                            <manufacturedProduct>
                                <manufacturedLabeledDrug>
                                    <code code="66493003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Theophylline"/>
                                </manufacturedLabeledDrug>
                            </manufacturedProduct>
                        </consumable>
                    </substanceAdministration>
                </entry>
                <entry>
                    <substanceAdministration classCode="SBADM" moodCode="EVN">
                        <text>Альбутерол ингалятор, 2 вдоха, 4 раза в день, по необходимости</text>
                        <effectiveTime xsi:type="PIVL_TS" institutionSpecified="true">
                            <period value="6" unit="h"/>
                        </effectiveTime>
                        <priorityCode code="PRN"/>
                        <routeCode code="IPINHL" codeSystem="2.16.840.1.113883.5.112" codeSystemName="Route
OfAdministration" displayName="Inhalation, oral"/>
                        <doseQuantity value="2"/>
                        <consumable>
                            <manufacturedProduct>
                                <manufacturedLabeledDrug>
```

```
                                    <code code="91143003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Albuterol"/>
                                </manufacturedLabeledDrug>
                            </manufacturedProduct>
                        </consumable>
                    </substanceAdministration>
                </entry>
                <entry>
                    <substanceAdministration classCode="SBADM" moodCode="EVN">
                        <id root=»2.16.840.1.113883.19.8.1»/>
                        <text>Преднизон, 20 мг в день</text>
                        <effectiveTime xsi:type="PIVL_TS" institutionSpecified="true">
                            <period value="24" unit="h"/>
                        </effectiveTime>
                        <routeCode code="PO" codeSystem="2.16.840.1.113883.5.112" codeSystemName="RouteOfA
dministration"/>
                        <doseQuantity value="20" unit="mg"/>
                        <consumable>
                            <manufacturedProduct>
                                <manufacturedLabeledDrug>
                                    <code code="10312003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Prednisone preparation"/>
                                </manufacturedLabeledDrug>
                            </manufacturedProduct>
                        </consumable>
                    </substanceAdministration>
                </entry>
                <entry>
                    <substanceAdministration classCode="SBADM" moodCode="EVN">
                        <text>Гидрохлортиазид, 25мг в день</text>
                        <effectiveTime xsi:type="PIVL_TS" institutionSpecified="true">
                            <period value="24" unit="h"/>
                        </effectiveTime>
                        <routeCode code="PO" codeSystem="2.16.840.1.113883.5.112" codeSystemName="RouteOfA
dministration"/>
                        <consumable>
                            <manufacturedProduct>
                                <manufacturedLabeledDrug>
                                    <code code="376209006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Hydrochlorothiazide 25mg tablet"/>
                                </manufacturedLabeledDrug>
                            </manufacturedProduct>
                        </consumable>
                    </substanceAdministration>
                </entry>
            </section>
        </component>
        <!--
*********************************************************************************************************
 Раздел аллергий и непереносимость
*********************************************************************************************************
```

```
-->
        <component>
            <section>
                <code code="10155-0" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Аллергии и непереносимость</title>
                <text>
                    <list>
                        <item>Пенициллин — крапивница</item>
                        <item>Аспирин — одышка</item>
                        <item>Кодеин — зуд и тошнота</item>
                    </list>
                </text>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code xsi:type="CD" code="247472004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Hives"/>
                        <statusCode code="completed"/>
                        <entryRelationship typeCode="MFST">
                            <observation classCode="OBS" moodCode="EVN">
                                <code xsi:type="CD" code="91936005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Allergy to penicillin"/>
                                <statusCode code="completed"/>
                            </observation>
                        </entryRelationship>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="56018004" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Wheezing"/>
                        <statusCode code="completed"/>
                        <entryRelationship typeCode="MFST">
                            <observation classCode="OBS" moodCode="EVN">
                                <code code="293586001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Allergy to aspirin"/>
                                <statusCode code="completed"/>
                            </observation>
                        </entryRelationship>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="32738000" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Pruritis"/>
                        <statusCode code="completed"/>
                        <entryRelationship typeCode="MFST">
                            <observation classCode="OBS" moodCode="EVN">
                                <id root="2.16.840.1.113883.19.1.2010"/>
                                <code code="62014003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Adverse reaction to drug">
```

```
                                    <qualifier>
                                        <name code="246075003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="causative agent"/>
                                        <value code="1476002" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="codeine"/>
                                    </qualifier>
                                </code>
                                <statusCode code="completed"/>
                            </observation>
                        </entryRelationship>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="73879007" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Nausea"/>
                        <statusCode code="completed"/>
                        <entryRelationship typeCode="MFST">
                            <observation classCode="OBS" moodCode="EVN">
                                <id root="2.16.840.1.113883.19.1.2010"/>
                                <code code="84100007" codeSystem="2.16.840.1.113883.6.96"/>
                            </observation>
                        </entryRelationship>
                    </observation>
                </entry>
            </section>
        </component>
        <!--
**********************************************************************************************************
Раздел семейного анамнеза
**********************************************************************************************************
-->
        <component>
            <section>
                <code code="10157-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Семейный анамнез</title>
                <text>
                    <list>
                        <item>Отец умер от инфаркта миокарда чуть старше 50 лет.</item>
                        <item>Рака или диабета не было.</item>
                    </list>
                </text>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="22298006" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="MI"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="1970"/>
                        <subject>
                            <relatedSubject classCode="PRS">
```

```
                        <code code="FTH" codeSystem="2.16.840.1.113883.5.111"/>
                    </relatedSubject>
                </subject>
                <entryRelationship typeCode="CAUS">
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="399347008" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="death"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="1970"/>
                    </observation>
                </entryRelationship>
            </observation>
        </entry>
        <entry>
            <observation classCode="OBS" moodCode="EVN" negationInd="true">
                <code code="275937001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Family history of cancer"/>
                <statusCode code="completed"/>
                <effectiveTime>
                    <high value="20000407" inclusive="true"/>
                </effectiveTime>
            </observation>
        </entry>
        <entry>
            <observation classCode="OBS" moodCode="EVN">
                <code code="160274005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="No family history of diabetes"/>
                <statusCode code="completed"/>
                <effectiveTime>
                    <high value="20000407" inclusive="true"/>
                </effectiveTime>
            </observation>
        </entry>
    </section>
</component>
<!—
**********************************************************************************************************
Раздел социального анамнеза
**********************************************************************************************************
-->
<component>
    <section>
        <code code="29762-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
        <title>Социальный анамнез</title>
        <text>
            <list>
                <item>Курение : 1 пачка в день с 20-ти до 55-ти лет, затем бросил.</item>
                <item>Алкоголь : изредка</item>
            </list>
        </text>
```

```
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="266924008" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="ex-heavy cigarette smoker (20-39/day)"/>
                            <statusCode code="completed"/>
                            <effectiveTime>
                                <low value="1955"/>
                                <high value="1990"/>
                            </effectiveTime>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="160625004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Date ceased smoking"/>
                            <statusCode code="completed"/>
                            <value xsi:type="TS" value="1990"/>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="266917007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Trivial drinker -  less than 1/day"/>
                            <statusCode code="completed"/>
                        </observation>
                    </entry>
                </section>
            </component>
            <!--
*********************************************************************************************************
 Раздел физикального осмотра
*********************************************************************************************************
-->
            <component>
                <section>
                    <code code="11384-5" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                    <title>Физикальный осмотр</title>
                    <!--
        *********************************************************************************************************
         Физикальный осмотр  - жизненно важные показатели
        *********************************************************************************************************
        -->
                    <component>
                        <section>
                            <code code="8716-3" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                            <title>Жизненно важные показатели</title>
                            <text>
                                <table>
                                    <tbody>
                                        <tr>
```

```
    <th>Дата / время</th>
    <th>7 апреля 2000 14:30</th>
    <th>7 апреля 2000 15:30</th>
</tr>
<tr>
    <th>Рост</th>
    <td>177 см</td>
</tr>
<tr>
    <th>Вес</th>
    <td>88,0 кг</td>
</tr>
<tr>
    <th>ИМТ</th>
    <td>28,1 кг/м2</td>
</tr>
<tr>
    <th>ППТ</th>
    <td>2.05 м2</td>
</tr>
<tr>
    <th>Температура</th>
    <td>36.9 по Цельсию</td>
    <td>36.9 по Цельсию</td>
</tr>
<tr>
    <th>Пульс</th>
    <td>86 / мин</td>
    <td>84 / мин</td>
</tr>
<tr>
    <th>Ритм</th>
    <td>Регулярный</td>
    <td>Регулярный</td>
</tr>
<tr>
    <th>Дыхание</th>
    <td>16 / мин, незатрудненное</td>
    <td>14 / мин</td>
</tr>
<tr>
    <th>Систолическое АД</th>
    <td>132 mmHg</td>
    <td>135 mmHg</td>
</tr>
<tr>
    <th>Диастолическое АД</th>
    <td>86 mmHg</td>
    <td>88 mmHg</td>
```

```
                                    </tr>
                                    <tr>
                                        <th>Положение / манжета</th>
                                        <td>Левая рука</td>
                                        <td>Левая рука</td>
                                    </tr>
                                </tbody>
                            </table>
                        </text>
                        <entry>
                            <observation classCode="OBS" moodCode="EVN">
                                <code code="50373000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Body height measure"/>
                                <statusCode code="completed"/>
                                <effectiveTime value="200004071430"/>
                                <value xsi:type="PQ" value="1.77" unit="m">
                                    <translation value="69.7" code="[in_i]" codeSystem="2.16.840.1.113883.6.8"
codeSystemName="UCUM"/>
                                </value>
                            </observation>
                        </entry>
                        <entry>
                            <observation classCode="OBS" moodCode="EVN">
                                <code code="363808001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Body weight measure"/>
                                <statusCode code="completed"/>
                                <effectiveTime value="200004071430"/>
                                <value xsi:type="PQ" value="194.0" unit="[lb_ap]">
                                    <translation value="88.0" code="kg" codeSystem="2.16.840.1.113883.6.8"
codeSystemName="UCUM"/>
                                </value>
                            </observation>
                        </entry>
                        <entry>
                            <observation classCode="OBS" moodCode="EVN">
                                <code code="60621009" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Body mass index"/>
                                <statusCode code="completed"/>
                                <effectiveTime value="200004071430"/>
                                <value xsi:type="RTO_PQ_PQ">
                                    <numerator value="28.1" unit="kg"/>
                                    <denominator value="1" unit="ar"/>
                                </value>
                            </observation>
                        </entry>
                        <entry>
                            <observation classCode="OBS" moodCode="EVN">
                                <code code="301898006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Body surface area"/>
                                <statusCode code="completed"/>
```

```
                    <effectiveTime value="200004071430"/>
                    <value xsi:type="PQ" value="2.05" unit="ar"/>
                </observation>
            </entry>
            <entry>
                <observation classCode="OBS" moodCode="EVN">
                    <code code="386725007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Body temperature"/>
                    <statusCode code="completed"/>
                    <effectiveTime value="200004071430"/>
                    <value xsi:type="PQ" value="36.9" unit="Cel">
                        <translation value="98.5" code="[degF]" codeSystem="2.16.840.1.113883.6.8"
codeSystemName="UCUM"/>
                    </value>
                </observation>
            </entry>
            <entry>
                <observation classCode="OBS" moodCode="EVN">
                    <code code="364075005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Heart rate"/>
                    <statusCode code="completed"/>
                    <effectiveTime value="200004071430"/>
                    <value xsi:type="RTO_PQ_PQ">
                        <numerator value="86"/>
                        <denominator value="1" unit="min"/>
                    </value>
                </observation>
            </entry>
            <entry>
                <observation classCode="OBS" moodCode="EVN">
                    <code code="364075005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Heart rate"/>
                    <statusCode code="completed"/>
                    <effectiveTime value="200004071530"/>
                    <value xsi:type="RTO_PQ_PQ">
                        <numerator value="84"/>
                        <denominator value="1" unit="min"/>
                    </value>
                </observation>
            </entry>
            <entry>
                <observation classCode="OBS" moodCode="EVN">
                    <code code="364074009" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Regularity of heart rhythm"/>
                    <statusCode code="completed"/>
                    <effectiveTime value="200004071430"/>
                    <value xsi:type="CD" code="248649006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Heart regular"/>
                </observation>
            </entry>
```

```
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="364074009" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Regularity of heart rhythm"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071530"/>
                            <value xsi:type="CD" code="248649006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Heart regular"/>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="86290005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Respiratory rate"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071430"/>
                            <value xsi:type="RTO_PQ_PQ">
                                <numerator value="16"/>
                                <denominator value="1" unit="min"/>
                            </value>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="276362002" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Breathing easily"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071430"/>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="86290005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Respiratory rate"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071530"/>
                            <value xsi:type="RTO_PQ_PQ">
                                <numerator value="14"/>
                                <denominator value="1" unit="min"/>
                            </value>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="251076008" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Cuff blood pressure"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071430"/>
                            <targetSiteCode code="368208006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Left arm"/>
                            <entryRelationship typeCode="COMP">
```

```
                                <observation classCode="OBS" moodCode="EVN">
                                    <code code="271649006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Systolic BP"/>
                                    <statusCode code="completed"/>
                                    <effectiveTime value="200004071530"/>
                                    <value xsi:type="PQ" value="132" unit="mm[Hg]"/>
                                </observation>
                            </entryRelationship>
                            <entryRelationship typeCode="COMP">
                                <observation classCode="OBS" moodCode="EVN">
                                    <code code="271650006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Diastolic BP"/>
                                    <statusCode code="completed"/>
                                    <effectiveTime value="200004071530"/>
                                    <value xsi:type="PQ" value="86" unit="mm[Hg]"/>
                                </observation>
                            </entryRelationship>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="251076008" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Cuff blood pressure"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071530"/>
                            <targetSiteCode code="368208006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Left arm"/>
                            <entryRelationship typeCode="COMP">
                                <observation classCode="OBS" moodCode="EVN">
                                    <code code="271649006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Systolic BP"/>
                                    <statusCode code="completed"/>
                                    <effectiveTime value="200004071530"/>
                                    <value xsi:type="PQ" value="135" unit="mm[Hg]"/>
                                </observation>
                            </entryRelationship>
                            <entryRelationship typeCode="COMP">
                                <observation classCode="OBS" moodCode="EVN">
                                    <code code="271650006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Diastolic BP"/>
                                    <statusCode code="completed"/>
                                    <effectiveTime value="200004071530"/>
                                    <value xsi:type="PQ" value="88" unit="mm[Hg]"/>
                                </observation>
                            </entryRelationship>
                        </observation>
                    </entry>
                </section>
            </component>
            <!--
```

```
        ********************************************************************************************************
         Физикальный осмотр - кожа
        ********************************************************************************************************
        -->
                    <component>
                        <section>
                            <code code="8709-8" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                            <title>Осмотр кожи</title>
                            <text>Эритемная сыпь, поверхность ладони, левый указательный палец.
                                <renderMultiMedia referencedObject="MM1"/>
                            </text>
                            <entry>
                                <observation classCode="OBS" moodCode="EVN">
                                    <code code="271807003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Rash"/>
                                    <statusCode code="completed"/>
                                    <methodCode code="32750006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Inspection"/>
                                    <targetSiteCode code="48856004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Skin of palmer surface of index finger">
                                        <qualifier>
                                            <name code="78615007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="with laterality"/>
                                            <value code="7771000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="left"/>
                                        </qualifier>
                                    </targetSiteCode>
                                    <entryRelationship typeCode="SPRT">
                                        <regionOfInterest classCode="ROIOVL" moodCode="EVN" ID="MM1">
                                            <id root="2.16.840.1.113883.19.3.1"/>
                                            <code code="ELLIPSE"/>
                                            <value value="3"/>
                                            <value value="1"/>
                                            <value value="3"/>
                                            <value value="7"/>
                                            <value value="2"/>
                                            <value value="4"/>
                                            <value value="4"/>
                                            <value value="4"/>
                                            <entryRelationship typeCode="SUBJ">
                                                <observationMedia classCode="OBS" moodCode="EVN">
                                                    <id root="2.16.840.1.113883.19.2.1"/>
                                                    <value mediaType="image/gif">
                                                        <reference value="lefthand.gif"/>
                                                    </value>
                                                </observationMedia>
                                            </entryRelationship>
                                        </regionOfInterest>
                                    </entryRelationship>
                                </observation>
```

```
                    </entry>
                </section>
            </component>
            <!--
    ********************************************************************************************************
     Физикальный осмотр - легкие
    ********************************************************************************************************
    -->
            <component>
                <section>
                    <code code="8710-6" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                    <title>Легкие</title>
                    <text>Чистые, без одышки. Дыхание хорошее.</text>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="48348007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Chest clear"/>
                            <statusCode code="completed"/>
                            <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN" negationInd="true">
                            <code code="56018004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Wheezing"/>
                            <statusCode code="completed"/>
                            <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                        </observation>
                    </entry>
                </section>
            </component>
            <!--
    ********************************************************************************************************
     Физикальный осмотр  - сердце
    ********************************************************************************************************
    -->
            <component>
                <section>
                    <code code="10223-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                    <title>Аускультация сердца</title>
                    <text>Нормальный ритм без шумов, нет S3, нет S4.</text>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="76863003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Normal heart rate"/>
                            <statusCode code="completed"/>
                            <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
```

```
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN" negationInd="true">
                        <code code="88610006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="heart murmur"/>
                        <statusCode code="completed"/>
                        <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN" negationInd="true">
                        <code code="277455002" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Third heart sound"/>
                        <statusCode code="completed"/>
                        <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="60721002" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Fourth heart sound inaudible"/>
                        <statusCode code="completed"/>
                        <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                    </observation>
                </entry>
            </section>
        </component>
    </section>
</component>
<!--
**********************************************************************************************************
Раздел лабораторных и диагностических исследований
**********************************************************************************************************
-->
<component>
    <section>
        <code code="11502-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
        <title>Лабораторные и диагностические исследования</title>
        <text>
            <list>
                <item>Рентген грудной клетки 02/03/1999: легкие растянуты. Нормальные очертания серд-
ца, легкие чистые.</item>
                <item>Пикфлоу сегодня: 260 л/мин</item>
            </list>
        </text>
        <entry>
            <observation classCode="OBS" moodCode="EVN">
```

```
                    <code code="282290005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Imaging interpretation"/>
                    <statusCode code="completed"/>
                    <entryRelationship typeCode="COMP">
                        <observation classCode="OBS" moodCode="EVN">
                            <id root="2.16.840.1.113883.19.1.3005"/>
                            <code code="249674001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Chest hyperinflated"/>
                        </observation>
                    </entryRelationship>
                    <entryRelationship typeCode="COMP">
                        <observation classCode="OBS" moodCode="EVN">
                            <id root="2.16.840.1.113883.19.1.5505"/>
                            <code codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT"
nullFlavor="OTH">
                                <originalText>normal cardiac silhouette</originalText>
                            </code>
                        </observation>
                    </entryRelationship>
                    <entryRelationship typeCode="COMP">
                        <observation classCode="OBS" moodCode="EVN" negationInd="true">
                            <id root="2.16.840.1.113883.19.1.6675"/>
                            <code codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT"
nullFlavor="OTH">
                                <originalText>radiopacities</originalText>
                            </code>
                        </observation>
                    </entryRelationship>
                    <reference typeCode="SPRT">
                        <externalObservation classCode="DGIMG">
                            <id root="2.16.840.1.113883.19.1.14"/>
                            <code code="56350004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Chest-X-ray"/>
                        </externalObservation>
                    </reference>
                </observation>
            </entry>
            <entry>
                <observation classCode="OBS" moodCode="EVN">
                    <id root="2.16.840.1.113883.19.1.7005"/>
                    <code code="313193002" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Peak flow"/>
                    <statusCode code="completed"/>
                    <effectiveTime value="20000407"/>
                    <value xsi:type="RTO_PQ_PQ">
                        <numerator value="260" unit="l"/>
                        <denominator value="1" unit="min"/>
                    </value>
                </observation>
            </entry>
```

```
            </section>
        </component>
        <!--
**********************************************************************************************************
 Раздел выполненных процедур
**********************************************************************************************************
-->
        <component>
            <section>
                <code code="29554-3" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Выполненные процедуры</title>
                <text>
                    <list>
                        <item>Снятие швов на левом предплечье.</item>
                    </list>
                </text>
                <entry>
                    <procedure classCode="PROC" moodCode="EVN">
                        <code code="30549001" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Suture removal"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071430"/>
                        <targetSiteCode code="66480008" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Left forearm"/>
                    </procedure>
                </entry>
            </section>
        </component>
        <!--
**********************************************************************************************************
Раздел заключения
**********************************************************************************************************
-->
        <component>
            <section>
                <code code="11496-7" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Заключение</title>
                <text>
                    <list>
                        <item>Астма, с предшествующей историей курения. С трудом отвыкает от стероидов. По-
пробует постепенный отказ.</item>
                        <item>Гипертензия, хорошо контролируемая.</item>
                        <item>Контактный дерматит пальца.</item>
                    </list>
                </text>
                <entry>
                    <observation classCode="COND" moodCode="EVN">
                        <code code="14657009" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Established diagnosis"/>
                        <statusCode code="completed"/>
```

```
                    <effectiveTime value="200004071530"/>
                    <value xsi:type="CD" code="195967001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Asthma">
                        <translation code="49390" codeSystem="2.16.840.1.113883.6.2"
codeSystemName="ICD9CM" displayName="ASTHMA W/O STATUS ASTHMATICUS"/>
                    </value>
                    <reference typeCode="ELNK">
                        <externalObservation classCode="COND">
                            <id root="2.16.840.1.113883.19.1.35"/>
                        </externalObservation>
                    </reference>
                </observation>
            </entry>
            <entry>
                <observation classCode="COND" moodCode="EVN">
                    <code code="14657009" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Established diagnosis"/>
                    <statusCode code="completed"/>
                    <effectiveTime value="200004071530"/>
                    <value xsi:type="CD" code="59621000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Essential hypertension">
                        <translation code="4019" codeSystem="2.16.840.1.113883.6.2"
codeSystemName="ICD9CM" displayName="HYPERTENSION NOS"/>
                    </value>
                    <reference typeCode="ELNK">
                        <externalObservation classCode="COND">
                            <id root="2.16.840.1.113883.19.1.37"/>
                        </externalObservation>
                    </reference>
                </observation>
            </entry>
            <entry>
                <observation classCode="COND" moodCode="EVN">
                    <code code="14657009" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Established diagnosis"/>
                    <statusCode code="completed"/>
                    <effectiveTime value="200004071530"/>
                    <value xsi:type="CD" code="40275004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Contact dermatitis">
                        <translation code="692.9" codeSystem="2.16.840.1.113883.6.2"
codeSystemName="ICD9CM" displayName="Contact Dermatitis, NOS"/>
                    </value>
                    <targetSiteCode code="48856004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Skin of palmer surface of index finger">
                        <qualifier>
                            <name code="78615007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="with laterality"/>
                            <value code="7771000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="left"/>
                        </qualifier>
                    </targetSiteCode>
```

```
                    </observation>
                </entry>
            </section>
        </component>
        <!--
**********************************************************************************************************
 Раздел рекомендаций
**********************************************************************************************************
-->
        <component>
            <section>
                <templateId root="2.16.840.1.113883.3.27.354"/>
                <code code="18776-5" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Рекомендации</title>
                <text>
                    <list>
                        <item>Полное исследование ФВД с определением объема легких.</item>
                        <item>Панель Chem-7 завтра.</item>
                        <item>Обучение измерению пикфлоу.</item>
                        <item>Уменьшить преднизон до 20 мг через день, чередуя с 18 мг через день.</item>
                        <item>Гидрокортизон крем для пальца 2 раза в день.</item>
                        <item>Повторный визит через неделю.</item>
                    </list>
                </text>
                <entry>
                    <act classCode="ACT" moodCode="INT">
                        <id/>
                        <code code="23426006" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Pulmonary function test"/>
                        <text>Полное исследование ФВД с определением объема легких.</text>
                        <entryRelationship typeCode="COMP">
                            <act classCode="ACT" moodCode="INT">
                                <code code="252472004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Lung volume test"/>
                            </act>
                        </entryRelationship>
                    </act>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="INT">
                        <code code="24320-4" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC">
                            <originalText>Chem-7</originalText>
                            <translation code="aYU7t6" codeSystem="2.16.840.1.113883.19.278.47" codeSystemName
="MyLocalCodeSystem" displayName="Chem7"/>
                        </code>
                        <text>Панель Chem-7 завтра</text>
                        <effectiveTime value="20000408"/>
                    </observation>
                </entry>
                <entry>
```

```
                    <act classCode="ACT" moodCode="INT">
                        <id/>
                        <code code="223468009" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Teaching of skills">
                            <qualifier>
                                <name code="363702006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="has focus"/>
                                <value code="29893006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Peak flow rate measurement"/>
                            </qualifier>
                        </code>
                    </act>
                </entry>
                <entry>
                    <substanceAdministration classCode="SBADM" moodCode="RQO">
                        <text>Уменьшить преднизон до 20 мг через день, чередуя с 18 мг через день.</text>
                        <routeCode code="PO" codeSystem="2.16.840.1.113883.5.112" codeSystemName="RouteOfA
dministration"/>
                        <consumable>
                            <manufacturedProduct>
                                <manufacturedLabeledDrug>
                                    <code code="10312003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Prednisone preparation"/>
                                </manufacturedLabeledDrug>
                            </manufacturedProduct>
                        </consumable>
                    </substanceAdministration>
                </entry>
                <entry>
                    <substanceAdministration classCode="SBADM" moodCode="RQO">
                        <text>Гидрокортизон крем для пальца 2 раза в день.</text>
                        <effectiveTime xsi:type="PIVL_TS" institutionSpecified="true">
                            <period value="12" unit="h"/>
                        </effectiveTime>
                        <routeCode code="SKIN" codeSystem="2.16.840.1.113883.5.112" codeSystemName="RouteOf
Administration" displayName="Topical application, skin"/>
                        <approachSiteCode code="48856004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Skin of palmer surface of index finger">
                            <qualifier>
                                <name code="78615007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="with laterality"/>
                                <value code="7771000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="left"/>
                            </qualifier>
                        </approachSiteCode>
                        <consumable>
                            <manufacturedProduct>
                                <manufacturedLabeledDrug>
                                    <code code="331646005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Hydrocortisone cream"/>
                                </manufacturedLabeledDrug>
```

```
                        </manufacturedProduct>
                    </consumable>
                </substanceAdministration>
            </entry>
            <entry>
                <encounter classCode="ENC" moodCode="RQO">
                    <code code="185389009" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Follow-up visit"/>
                    <effectiveTime>
                        <low value="20000412"/>
                        <high value="20000417"/>
                    </effectiveTime>
                </encounter>
            </entry>
        </section>
    </component>
</structuredBody>    </component>
</ClinicalDocument>
```

Пример АКД-документа без таблицы стилей

```
<?xml version="1.0"?>
<!--?xml-stylesheet type="text/xsl" href="CDA.xsl"?-->
<!--
Читателям рекомендуется иметь в виду руководство по реализации «Using SNOMED CT in HL7 Version 3», которое пока что находится
на стадии разработки. Это руководство, разрабатываемое совместно комитетом HL7 и Институтом американских патологоанатомов
(College of American Pathologists), будет представлено на утверждение в комитет HL7 как справочный документ. Шаблоны использования
SNOMED CT, приведенные в настоящем примере, будут переработаны в соответствии с рекомендациями окончательной версии
этого руководства.
-->
<ClinicalDocument xmlns=»urn:hl7-org:v3» xmlns:voc=»urn:hl7-org:v3/voc» xmlns:xsi=»http://www.w3.org/2001/
XMLSchema-instance» xsi:schemaLocation=»urn:hl7-org:v3 CDA.xsd»>
    <!--
**********************************************************************************************************
  CDA Header
**********************************************************************************************************
-->
    <typeId root="2.16.840.1.113883.1.3" extension="POCD_HD000040"/>
    <templateId root="2.16.840.1.113883.3.27.1776"/>
    <id extension="c266" root="2.16.840.1.113883.19.4"/>
    <code code="11488-4" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"
displayName="Consultation note"/>
    <title>Заключение консультанта клиники Good Health Clinic</title>
    <effectiveTime value="20000407"/>
    <confidentialityCode code="N" codeSystem="2.16.840.1.113883.5.25"/>
    <languageCode code="en-US"/>
    <setId extension="BB35" root="2.16.840.1.113883.19.7"/>
    <versionNumber value="2"/>
```

```
<recordTarget>
    <patientRole>
        <id extension="12345" root="2.16.840.1.113883.19.5"/>
        <patient>
            <name>
                <given>Генри</given>
                <family>Левин</family>
                <suffix>7-й</suffix>
            </name>
            <administrativeGenderCode code="M" codeSystem="2.16.840.1.113883.5.1"/>
            <birthTime value="19320924"/>
        </patient>
        <providerOrganization>
            <id root="2.16.840.1.113883.19.5"/>
        </providerOrganization>
    </patientRole>
</recordTarget>
<author>
    <time value="2000040714"/>
    <assignedAuthor>
        <id extension="KP00017" root="2.16.840.1.113883.19.5"/>
        <assignedPerson>
            <name>
                <given>Роберт</given>
                <family>Долин</family>
                <suffix>д.м.н.</suffix>
            </name>
        </assignedPerson>
        <representedOrganization>
            <id root="2.16.840.1.113883.19.5"/>
        </representedOrganization>
    </assignedAuthor>
</author>
<custodian>
    <assignedCustodian>
        <representedCustodianOrganization>
            <id root="2.16.840.1.113883.19.5"/>
            <name>Good Health Clinic</name>
        </representedCustodianOrganization>
    </assignedCustodian>
</custodian>
<legalAuthenticator>
    <time value="20000408"/>
    <signatureCode code="S"/>
    <assignedEntity>
        <id extension="KP00017" root="2.16.840.1.113883.19.5"/>
        <assignedPerson>
            <name>
                <given>Роберт</given>
```

```
                    <family>Долин</family>
                    <suffix>д.м.н.</suffix>
                </name>
            </assignedPerson>
            <representedOrganization>
                <id root="2.16.840.1.113883.19.5"/>
            </representedOrganization>
        </assignedEntity>
    </legalAuthenticator>
    <relatedDocument typeCode="RPLC">
        <parentDocument>
            <id extension="a123" root="2.16.840.1.113883.19.4"/>
            <setId extension="BB35" root="2.16.840.1.113883.19.7"/>
            <versionNumber value="1"/>
        </parentDocument>
    </relatedDocument>
    <componentOf>
        <encompassingEncounter>
            <id extension="KPENC1332" root="2.16.840.1.113883.19.6"/>
            <effectiveTime value="20000407"/>
            <encounterParticipant typeCode="CON">
                <time value="20000407"/>
                <assignedEntity>
                    <id extension="KP00017" root="2.16.840.1.113883.19.5"/>
                    <assignedPerson>
                        <name>
                            <given>Роберт</given>
                            <family>Долин</family>
                            <suffix>д.м.н.</suffix>
                        </name>
                    </assignedPerson>
                    <representedOrganization>
                        <id root="2.16.840.1.113883.19.5"/>
                    </representedOrganization>
                </assignedEntity>
            </encounterParticipant>
            <location>
                <healthCareFacility classCode="DSDLOC">
                    <code code="GIM" codeSystem="2.16.840.1.113883.5.10588" displayName="Клиника общей вну-
тренней медицины"/>
                </healthCareFacility>
            </location>
        </encompassingEncounter>
    </componentOf>
    <!--
**********************************************************************************************************
Тело АКД
**********************************************************************************************************
-->
```

```
    <component>
        <structuredBody>
            <!--
**********************************************************************************************************
 Раздел истории текущего заболевания
**********************************************************************************************************
-->
            <component>
                <section>
                    <code code="10164-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                    <title>История текущего заболевания</title>
                    <text>
                        <content styleCode="Bold">Генри Левин 7-й
                        </content> 67 лет, обратился за дальнейшим лечением от астмы. Заболел, когда был <content
revised="delete">двадцатилетним</content>
                        <content revised="insert">подростком</content>.  В прошлом году был дважды госпитализиро-
ван, и уже дважды в этом году. Последние несколько месяцев не может обходиться без стероидов.
                    </text>
                </section>
            </component>
            <!--
**********************************************************************************************************
Раздел истории заболеваний
**********************************************************************************************************
-->
            <component>
                <section>
                    <code code="10153-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                    <title>История заболеваний</title>
                    <text>
                        <list>
                            <item>
                                <content ID="a1">Астма</content>
                            </item>
                            <item>
                                <content ID="a2">Гипертензия (детали см. в документе HTN.cda)</content>
                            </item>
                            <item>
                                <content ID="a3">Остеоартрит
                                    <content ID="a4"> правого колена</content>
                                </content>
                            </item>
                        </list>
                    </text>
                    <entry>
                        <observation classCode="COND" moodCode="EVN">
                            <code code="195967001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Asthma">
                                <originalText>
                                    <reference value="#a1"/>
```

```
                    </originalText>
                </code>
                <statusCode code="completed"/>
                <effectiveTime value="1950"/>
                <reference typeCode="XCRPT">
                    <externalObservation>
                        <id root="2.16.840.1.113883.19.1.2765"/>
                    </externalObservation>
                </reference>
            </observation>
        </entry>
        <entry>
            <observation classCode="COND" moodCode="EVN">
                <code code="59621000" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="HTN">
                    <originalText>
                        <reference value="#a2"/>
                    </originalText>
                </code>
                <statusCode code="completed"/>
                <reference typeCode="SPRT">
                    <seperatableInd value="false"/>
                    <externalDocument>
                        <id root="2.16.840.1.113883.19.4.789"/>
                        <text mediaType="multipart/related">
                            <reference value="HTN.cda"/>
                        </text>
                        <setId root="2.16.840.1.113883.19.7.2465"/>
                        <versionNumber value="1"/>
                    </externalDocument>
                </reference>
                <reference typeCode="XCRPT">
                    <externalObservation>
                        <id root="2.16.840.1.113883.19.1.2005"/>
                    </externalObservation>
                </reference>
            </observation>
        </entry>
        <entry>
            <observation classCode="COND" moodCode="EVN">
                <code xsi:type="CD" code="396275006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Osteoarthritis">
                    <originalText>
                        <reference value="#a3"/>
                    </originalText>
                </code>
                <statusCode code="completed"/>
                <targetSiteCode code="49076000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Knee joint">
                    <originalText>
```

```
                        <reference value="#a4"/>
                    </originalText>
                    <qualifier>
                        <name code="78615007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="with laterality"/>
                        <value code="24028007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="right"/>
                    </qualifier>
                </targetSiteCode>
                <reference typeCode="XCRPT">
                    <externalObservation>
                        <id root="2.16.840.1.113883.19.1.1805"/>
                    </externalObservation>
                </reference>
            </observation>
        </entry>
    </section>
</component>
<!--
**********************************************************************************************************
 Раздел лекарственных назначений
**********************************************************************************************************
-->
<component>
    <section>
        <code code="10160-0" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
        <title>Медикаментозная терапия</title>
        <text>
            <list>
                <item>Теодур, 200 мг два раза в день</item>
                <item>Альбутерол ингалятор, 2 вдоха, 4 раза в день, по необходимости</item>
                <item>Преднизон, 20 мг в день</item>
                <item>Гидрохлортиазид, 25мг в день</item>
            </list>
        </text>
        <entry>
            <substanceAdministration classCode="SBADM" moodCode="EVN">
                <text>Теодур, 200 мг два раза в день</text>
                <effectiveTime xsi:type="PIVL_TS" institutionSpecified="true">
                    <period value="12" unit="h"/>
                </effectiveTime>
                <routeCode code="PO" codeSystem="2.16.840.1.113883.5.112" codeSystemName="RouteOfA
dministration"/>
                <doseQuantity value="200" unit="mg"/>
                <consumable>
                    <manufacturedProduct>
                        <manufacturedLabeledDrug>
                            <code code="66493003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Theophylline"/>
                        </manufacturedLabeledDrug>
```

```
                        </manufacturedProduct>
                    </consumable>
                </substanceAdministration>
            </entry>
            <entry>
                <substanceAdministration classCode="SBADM" moodCode="EVN">
                    <text>Альбутерол ингалятор, 2 вдоха, 4 раза в день, по необходимости</text>
                    <effectiveTime xsi:type="PIVL_TS" institutionSpecified="true">
                        <period value="6" unit="h"/>
                    </effectiveTime>
                    <priorityCode code="PRN"/>
                    <routeCode code="IPINHL" codeSystem="2.16.840.1.113883.5.112" codeSystemName="Route
OfAdministration" displayName="Inhalation, oral"/>
                    <doseQuantity value="2"/>
                    <consumable>
                        <manufacturedProduct>
                            <manufacturedLabeledDrug>
                                <code code="91143003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Albuterol"/>
                            </manufacturedLabeledDrug>
                        </manufacturedProduct>
                    </consumable>
                </substanceAdministration>
            </entry>
            <entry>
                <substanceAdministration classCode="SBADM" moodCode="EVN">
                    <id root=»2.16.840.1.113883.19.8.1»/>
                    <text>Преднизон, 20 мг в день</text>
                    <effectiveTime xsi:type="PIVL_TS" institutionSpecified="true">
                        <period value="24" unit="h"/>
                    </effectiveTime>
                    <routeCode code="PO" codeSystem="2.16.840.1.113883.5.112" codeSystemName="RouteOfA
dministration"/>
                    <doseQuantity value="20" unit="mg"/>
                    <consumable>
                        <manufacturedProduct>
                            <manufacturedLabeledDrug>
                                <code code="10312003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Prednisone preparation"/>
                            </manufacturedLabeledDrug>
                        </manufacturedProduct>
                    </consumable>
                </substanceAdministration>
            </entry>
            <entry>
                <substanceAdministration classCode="SBADM" moodCode="EVN">
                    <text>Гидрохлортиазид, 25мг в день</text>
                    <effectiveTime xsi:type="PIVL_TS" institutionSpecified="true">
                        <period value="24" unit="h"/>
                    </effectiveTime>
```

```
                    <routeCode code="PO" codeSystem="2.16.840.1.113883.5.112" codeSystemName="RouteOfA
dministration"/>
                    <consumable>
                        <manufacturedProduct>
                            <manufacturedLabeledDrug>
                                <code code="376209006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Hydrochlorothiazide 25mg tablet"/>
                            </manufacturedLabeledDrug>
                        </manufacturedProduct>
                    </consumable>
                </substanceAdministration>
            </entry>
        </section>
    </component>
    <!--
**********************************************************************************************************
  Раздел аллергий и непереносимости
**********************************************************************************************************
-->
    <component>
        <section>
            <code code="10155-0" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
            <title>Аллергии и непереносимость</title>
            <text>
                <list>
                    <item>Пенициллин — крапивница</item>
                    <item>Аспирин — одышка</item>
                    <item>Кодеин — зуд и тошнота</item>
                </list>
            </text>
            <entry>
                <observation classCode="OBS" moodCode="EVN">
                    <code xsi:type="CD" code="247472004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Hives"/>
                    <statusCode code="completed"/>
                    <entryRelationship typeCode="MFST">
                        <observation classCode="OBS" moodCode="EVN">
                            <code xsi:type="CD" code="91936005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Allergy to penicillin"/>
                            <statusCode code="completed"/>
                        </observation>
                    </entryRelationship>
                </observation>
            </entry>
            <entry>
                <observation classCode="OBS" moodCode="EVN">
                    <code code="56018004" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Wheezing"/>
                    <statusCode code="completed"/>
                    <entryRelationship typeCode="MFST">
```

```
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="293586001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Allergy to aspirin"/>
                            <statusCode code="completed"/>
                        </observation>
                    </entryRelationship>
                </observation>
            </entry>
            <entry>
                <observation classCode="OBS" moodCode="EVN">
                    <code code="32738000" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Pruritis"/>
                    <statusCode code="completed"/>
                    <entryRelationship typeCode="MFST">
                        <observation classCode="OBS" moodCode="EVN">
                            <id root="2.16.840.1.113883.19.1.2010"/>
                            <code code="62014003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Adverse reaction to drug">
                                <qualifier>
                                    <name code="246075003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="causative agent"/>
                                    <value code="1476002" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="codeine"/>
                                </qualifier>
                            </code>
                            <statusCode code="completed"/>
                        </observation>
                    </entryRelationship>
                </observation>
            </entry>
            <entry>
                <observation classCode="OBS" moodCode="EVN">
                    <code code="73879007" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Nausea"/>
                    <statusCode code="completed"/>
                    <entryRelationship typeCode="MFST">
                        <observation classCode="OBS" moodCode="EVN">
                            <id root="2.16.840.1.113883.19.1.2010"/>
                            <code code="84100007" codeSystem="2.16.840.1.113883.6.96"/>
                        </observation>
                    </entryRelationship>
                </observation>
            </entry>
        </section>
    </component>
    <!--
**********************************************************************************************************
Раздел семейного анамнеза
**********************************************************************************************************
-->
    <component>
```

```
            <section>
                <code code="10157-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Семейный анамнез</title>
                <text>
                    <list>
                        <item>Отец умер от инфаркта миокарда чуть старше 50 лет.</item>
                        <item>Рака или диабета не было.</item>
                    </list>
                </text>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="22298006" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="MI"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="1970"/>
                        <subject>
                            <relatedSubject classCode="PRS">
                                <code code="FTH" codeSystem="2.16.840.1.113883.5.111"/>
                            </relatedSubject>
                        </subject>
                        <entryRelationship typeCode="CAUS">
                            <observation classCode="OBS" moodCode="EVN">
                                <code code="399347008" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="death"/>
                                <statusCode code="completed"/>
                                <effectiveTime value="1970"/>
                            </observation>
                        </entryRelationship>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN" negationInd="true">
                        <code code="275937001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Family history of cancer"/>
                        <statusCode code="completed"/>
                        <effectiveTime>
                            <high value="20000407" inclusive="true"/>
                        </effectiveTime>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="160274005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="No family history of diabetes"/>
                        <statusCode code="completed"/>
                        <effectiveTime>
                            <high value="20000407" inclusive="true"/>
                        </effectiveTime>
                    </observation>
                </entry>
```

```
            </section>
        </component>
        <!--
**********************************************************************************************************
Раздел социального анамнеза
**********************************************************************************************************
-->
        <component>
            <section>
                <code code="29762-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Социальный анамнез</title>
                <text>
                    <list>
                        <item>Курение : 1 пачка в день с 20-ти до 55-ти лет, затем бросил.</item>
                        <item>Алкоголь : изредка</item>
                    </list>
                </text>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="266924008" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="ex-heavy cigarette smoker (20-39/day)"/>
                        <statusCode code="completed"/>
                        <effectiveTime>
                            <low value="1955"/>
                            <high value="1990"/>
                        </effectiveTime>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="160625004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Date ceased smoking"/>
                        <statusCode code="completed"/>
                        <value xsi:type="TS" value="1990"/>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="266917007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Trivial drinker -  less than 1/day"/>
                        <statusCode code="completed"/>
                    </observation>
                </entry>
            </section>
        </component>
        <!--
**********************************************************************************************************
 Раздел физикального осмотра
**********************************************************************************************************
-->
        <component>
```

```
        <section>
            <code code="11384-5" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
            <title>Физикальный осмотр</title>
            <!--
**********************************************************************************************************
 Физикальный осмотр  - жизненно важные показатели
**********************************************************************************************************
-->
            <component>
                <section>
                    <code code="8716-3" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                    <title>Жизненно важные показатели</title>
                    <text>
                        <table>
                            <tbody>
                                <tr>
                                    <th>Дата / время</th>
                                    <th>7 апреля 2000 14:30</th>
                                    <th>7 апреля 2000 15:30</th>
                                </tr>
                                <tr>
                                    <th>Рост</th>
                                    <td>177 см</td>
                                </tr>
                                <tr>
                                    <th>Вес</th>
                                    <td>88,0 кг</td>
                                </tr>
                                <tr>
                                    <th>ИМТ</th>
                                    <td>28,1 кг/м2</td>
                                </tr>
                                <tr>
                                    <th>ППТ</th>
                                    <td>2.05 м2</td>
                                </tr>
                                <tr>
                                    <th>Температура</th>
                                    <td>36.9 по Цельсию</td>
                                    <td>36.9 по Цельсию</td>
                                </tr>
                                <tr>
                                    <th>Пульс</th>
                                    <td>86 / мин</td>
                                    <td>84 / мин</td>
                                </tr>
                                <tr>
                                    <th>Ритм</th>
                                    <td>Регулярный</td>
```

```
                                    <td>Регулярный</td>
                                </tr>
                                <tr>
                                    <th>Дыхание</th>
                                    <td>16 / мин, незатрудненное</td>
                                    <td>14 / мин</td>
                                </tr>
                                <tr>
                                    <th>Систолическое АД</th>
                                    <td>132 mmHg</td>
                                    <td>135 mmHg</td>
                                </tr>
                                <tr>
                                    <th>Диастолическое АД</th>
                                    <td>86 mmHg</td>
                                    <td>88 mmHg</td>
                                </tr>
                                <tr>
                                    <th>Положение / манжета</th>
                                    <td>Левая рука</td>
                                    <td>Левая рука</td>
                                </tr>
                            </tbody>
                        </table>
                    </text>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="50373000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Body height measure"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071430"/>
                            <value xsi:type="PQ" value="1.77" unit="m">
                                <translation value="69.7" code="[in_i]" codeSystem="2.16.840.1.113883.6.8"
codeSystemName="UCUM"/>
                            </value>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="363808001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Body weight measure"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071430"/>
                            <value xsi:type="PQ" value="194.0" unit="[lb_ap]">
                                <translation value="88.0" code="kg" codeSystem="2.16.840.1.113883.6.8"
codeSystemName="UCUM"/>
                            </value>
                        </observation>
                    </entry>
                    <entry>
```

```
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="60621009" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Body mass index"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071430"/>
                        <value xsi:type="RTO_PQ_PQ">
                            <numerator value="28.1" unit="kg"/>
                            <denominator value="1" unit="ar"/>
                        </value>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="301898006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Body surface area"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071430"/>
                        <value xsi:type="PQ" value="2.05" unit="ar"/>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="386725007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Body temperature"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071430"/>
                        <value xsi:type="PQ" value="36.9" unit="Cel">
                            <translation value="98.5" code="[degF]" codeSystem="2.16.840.1.113883.6.8"
codeSystemName="UCUM"/>
                        </value>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="364075005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Heart rate"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071430"/>
                        <value xsi:type="RTO_PQ_PQ">
                            <numerator value="86"/>
                            <denominator value="1" unit="min"/>
                        </value>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="364075005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Heart rate"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071530"/>
                        <value xsi:type="RTO_PQ_PQ">
```

```
                                <numerator value="84"/>
                                <denominator value="1" unit="min"/>
                            </value>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="364074009" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Regularity of heart rhythm"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071430"/>
                            <value xsi:type="CD" code="248649006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Heart regular"/>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="364074009" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Regularity of heart rhythm"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071530"/>
                            <value xsi:type="CD" code="248649006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Heart regular"/>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="86290005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Respiratory rate"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071430"/>
                            <value xsi:type="RTO_PQ_PQ">
                                <numerator value="16"/>
                                <denominator value="1" unit="min"/>
                            </value>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="276362002" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Breathing easily"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071430"/>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="86290005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Respiratory rate"/>
                            <statusCode code="completed"/>
                            <effectiveTime value="200004071530"/>
```

```
                        <value xsi:type="RTO_PQ_PQ">
                            <numerator value="14"/>
                            <denominator value="1" unit="min"/>
                        </value>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="251076008" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Cuff blood pressure"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071430"/>
                        <targetSiteCode code="368208006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Left arm"/>
                        <entryRelationship typeCode="COMP">
                            <observation classCode="OBS" moodCode="EVN">
                                <code code="271649006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Systolic BP"/>
                                <statusCode code="completed"/>
                                <effectiveTime value="200004071530"/>
                                <value xsi:type="PQ" value="132" unit="mm[Hg]"/>
                            </observation>
                        </entryRelationship>
                        <entryRelationship typeCode="COMP">
                            <observation classCode="OBS" moodCode="EVN">
                                <code code="271650006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Diastolic BP"/>
                                <statusCode code="completed"/>
                                <effectiveTime value="200004071530"/>
                                <value xsi:type="PQ" value="86" unit="mm[Hg]"/>
                            </observation>
                        </entryRelationship>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="251076008" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Cuff blood pressure"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071530"/>
                        <targetSiteCode code="368208006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Left arm"/>
                        <entryRelationship typeCode="COMP">
                            <observation classCode="OBS" moodCode="EVN">
                                <code code="271649006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Systolic BP"/>
                                <statusCode code="completed"/>
                                <effectiveTime value="200004071530"/>
                                <value xsi:type="PQ" value="135" unit="mm[Hg]"/>
                            </observation>
                        </entryRelationship>
```

```
                        <entryRelationship typeCode="COMP">
                            <observation classCode="OBS" moodCode="EVN">
                                <code code="271650006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Diastolic BP"/>
                                <statusCode code="completed"/>
                                <effectiveTime value="200004071530"/>
                                <value xsi:type="PQ" value="88" unit="mm[Hg]"/>
                            </observation>
                        </entryRelationship>
                    </observation>
                </entry>
            </section>
        </component>
        <!--
    ********************************************************************************************************
     Физикальный осмотр - кожа
    ********************************************************************************************************
    -->
        <component>
            <section>
                <code code="8709-8" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Осмотр кожи</title>
                <text>Эритемная сыпь, поверхность ладони, левый указательный палец.
                    <renderMultiMedia referencedObject="MM1"/>
                </text>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="271807003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Rash"/>
                        <statusCode code="completed"/>
                        <methodCode code="32750006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Inspection"/>
                        <targetSiteCode code="48856004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Skin of palmer surface of index finger">
                            <qualifier>
                                <name code="78615007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="with laterality"/>
                                <value code="7771000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="left"/>
                            </qualifier>
                        </targetSiteCode>
                        <entryRelationship typeCode="SPRT">
                            <regionOfInterest classCode="ROIOVL" moodCode="EVN" ID="MM1">
                                <id root="2.16.840.1.113883.19.3.1"/>
                                <code code="ELLIPSE"/>
                                <value value="3"/>
                                <value value="1"/>
                                <value value="3"/>
                                <value value="7"/>
                                <value value="2"/>
                                <value value="4"/>
```

```
                                    <value value="4"/>
                                    <value value="4"/>
                                    <entryRelationship typeCode="SUBJ">
                                        <observationMedia classCode="OBS" moodCode="EVN">
                                            <id root="2.16.840.1.113883.19.2.1"/>
                                            <value mediaType="image/gif">
                                                <reference value="lefthand.gif"/>
                                            </value>
                                        </observationMedia>
                                    </entryRelationship>
                                </regionOfInterest>
                            </entryRelationship>
                        </observation>
                    </entry>
                </section>
            </component>
            <!--
    ********************************************************************************************************
     Физикальный осмотр - легкие
    ********************************************************************************************************
    -->
            <component>
                <section>
                    <code code="8710-6" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                    <title>Легкие</title>
                    <text>Чистые, без одышки. Дыхание хорошее.</text>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN">
                            <code code="48348007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Chest clear"/>
                            <statusCode code="completed"/>
                            <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                        </observation>
                    </entry>
                    <entry>
                        <observation classCode="OBS" moodCode="EVN" negationInd="true">
                            <code code="56018004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Wheezing"/>
                            <statusCode code="completed"/>
                            <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                        </observation>
                    </entry>
                </section>
            </component>
            <!--
    ********************************************************************************************************
     Физикальный осмотр  - сердце
    ********************************************************************************************************
    -->
```

```
                    <component>
                        <section>
                            <code code="10223-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                            <title>Аускультация сердца</title>
                            <text>Нормальный ритм без шумов, нет S3, нет S4.</text>
                            <entry>
                                <observation classCode="OBS" moodCode="EVN">
                                    <code code="76863003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Normal heart rate"/>
                                    <statusCode code="completed"/>
                                    <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                                </observation>
                            </entry>
                            <entry>
                                <observation classCode="OBS" moodCode="EVN" negationInd="true">
                                    <code code="88610006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="heart murmur"/>
                                    <statusCode code="completed"/>
                                    <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                                </observation>
                            </entry>
                            <entry>
                                <observation classCode="OBS" moodCode="EVN" negationInd="true">
                                    <code code="277455002" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Third heart sound"/>
                                    <statusCode code="completed"/>
                                    <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                                </observation>
                            </entry>
                            <entry>
                                <observation classCode="OBS" moodCode="EVN">
                                    <code code="60721002" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Fourth heart sound inaudible"/>
                                    <statusCode code="completed"/>
                                    <methodCode code="37931006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Auscultation"/>
                                </observation>
                            </entry>
                        </section>
                    </component>
                </section>
            </component>
            <!--
**********************************************************************************************************
Раздел лабораторных и диагностических исследований
**********************************************************************************************************
-->
            <component>
```

```
            <section>
                <code code="11502-2" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Лабораторные и диагностические исследования</title>
                <text>
                    <list>
                        <item>Рентген грудной клетки 02/03/1999: легкие растянуты. Нормальные очертания серд-
ца, легкие чистые.</item>
                        <item>Пикфлоу сегодня: 260 л/мин</item>
                    </list>
                </text>
                <entry>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="282290005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Imaging interpretation"/>
                        <statusCode code="completed"/>
                        <entryRelationship typeCode="COMP">
                            <observation classCode="OBS" moodCode="EVN">
                                <id root="2.16.840.1.113883.19.1.3005"/>
                                <code code="249674001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Chest hyperinflated"/>
                            </observation>
                        </entryRelationship>
                        <entryRelationship typeCode="COMP">
                            <observation classCode="OBS" moodCode="EVN">
                                <id root="2.16.840.1.113883.19.1.5505"/>
                                <code codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT"
nullFlavor="OTH">
                                    <originalText>normal cardiac silhouette</originalText>
                                </code>
                            </observation>
                        </entryRelationship>
                        <entryRelationship typeCode="COMP">
                            <observation classCode="OBS" moodCode="EVN" negationInd="true">
                                <id root="2.16.840.1.113883.19.1.6675"/>
                                <code codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT"
nullFlavor="OTH">
                                    <originalText>radiopacities</originalText>
                                </code>
                            </observation>
                        </entryRelationship>
                        <reference typeCode="SPRT">
                            <externalObservation classCode="DGIMG">
                                <id root="2.16.840.1.113883.19.1.14"/>
                                <code code="56350004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Chest-X-ray"/>
                            </externalObservation>
                        </reference>
                    </observation>
                </entry>
                <entry>
```

```
                    <observation classCode="OBS" moodCode="EVN">
                        <id root="2.16.840.1.113883.19.1.7005"/>
                        <code code="313193002" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Peak flow"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="20000407"/>
                        <value xsi:type="RTO_PQ_PQ">
                            <numerator value="260" unit="l"/>
                            <denominator value="1" unit="min"/>
                        </value>
                    </observation>
                </entry>
            </section>
        </component>
        <!--
*********************************************************************************************************
 Раздел выполненных процедур
*********************************************************************************************************
-->
        <component>
            <section>
                <code code="29554-3" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Выполненные процедуры</title>
                <text>
                    <list>
                        <item>Снятие швов на левом предплечье.</item>
                    </list>
                </text>
                <entry>
                    <procedure classCode="PROC" moodCode="EVN">
                        <code code="30549001" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Suture removal"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071430"/>
                        <targetSiteCode code="66480008" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Left forearm"/>
                    </procedure>
                </entry>
            </section>
        </component>
        <!--
*********************************************************************************************************
Раздел заключения
*********************************************************************************************************
-->
        <component>
            <section>
                <code code="11496-7" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
                <title>Заключение</title>
                <text>
```

```
                    <list>
                        <item>Астма, с предшествующей историей курения. С трудом отвыкает от стероидов. По-
пробует постепенный отказ.</item>
                        <item>Гипертензия, хорошо контролируемая.</item>
                        <item>Контактный дерматит пальца.</item>
                    </list>
                </text>
                <entry>
                    <observation classCode="COND" moodCode="EVN">
                        <code code="14657009" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Established diagnosis"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071530"/>
                        <value xsi:type="CD" code="195967001" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Asthma">
                            <translation code="49390" codeSystem="2.16.840.1.113883.6.2"
codeSystemName="ICD9CM" displayName="ASTHMA W/O STATUS ASTHMATICUS"/>
                        </value>
                        <reference typeCode="ELNK">
                            <externalObservation classCode="COND">
                                <id root="2.16.840.1.113883.19.1.35"/>
                            </externalObservation>
                        </reference>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="COND" moodCode="EVN">
                        <code code="14657009" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Established diagnosis"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071530"/>
                        <value xsi:type="CD" code="59621000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Essential hypertension">
                            <translation code="4019" codeSystem="2.16.840.1.113883.6.2"
codeSystemName="ICD9CM" displayName="HYPERTENSION NOS"/>
                        </value>
                        <reference typeCode="ELNK">
                            <externalObservation classCode="COND">
                                <id root="2.16.840.1.113883.19.1.37"/>
                            </externalObservation>
                        </reference>
                    </observation>
                </entry>
                <entry>
                    <observation classCode="COND" moodCode="EVN">
                        <code code="14657009" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Established diagnosis"/>
                        <statusCode code="completed"/>
                        <effectiveTime value="200004071530"/>
                        <value xsi:type="CD" code="40275004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Contact dermatitis">
```

```
                        <translation code="692.9" codeSystem="2.16.840.1.113883.6.2"
codeSystemName="ICD9CM" displayName="Contact Dermatitis, NOS"/>
                    </value>
                    <targetSiteCode code="48856004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Skin of palmer surface of index finger">
                        <qualifier>
                            <name code="78615007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="with laterality"/>
                            <value code="7771000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="left"/>
                        </qualifier>
                    </targetSiteCode>
                </observation>
            </entry>
        </section>
    </component>
    <!--
**********************************************************************************************************
 Раздел рекомендаций
**********************************************************************************************************
-->
    <component>
        <section>
            <templateId root="2.16.840.1.113883.3.27.354"/>
            <code code="18776-5" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"/>
            <title>Рекомендации</title>
            <text>
                <list>
                    <item>Полное исследование ФВД с определением объема легких.</item>
                    <item>Панель Chem-7 завтра.</item>
                    <item>Обучение измерению пикфлоу.</item>
                    <item>Уменьшить преднизон до 20 мг через день, чередуя с 18 мг через день.</item>
                    <item>Гидрокортизон крем для пальца 2 раза в день.</item>
                    <item>Повторный визит через неделю.</item>
                </list>
            </text>
            <entry>
                <act classCode="ACT" moodCode="INT">
                    <id/>
                    <code code="23426006" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED
CT" displayName="Pulmonary function test"/>
                    <text>Полное исследование ФВД с определением объема легких.</text>
                    <entryRelationship typeCode="COMP">
                        <act classCode="ACT" moodCode="INT">
                            <code code="252472004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Lung volume test"/>
                        </act>
                    </entryRelationship>
                </act>
            </entry>
```

```
                <entry>
                    <observation classCode="OBS" moodCode="INT">
                        <code code="24320-4" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC">
                            <originalText>Chem-7</originalText>
                            <translation code="aYU7t6" codeSystem="2.16.840.1.113883.19.278.47" codeSystemName
="MyLocalCodeSystem" displayName="Chem7"/>
                        </code>
                        <text>Панель Chem-7 завтра</text>
                        <effectiveTime value="20000408"/>
                    </observation>
                </entry>
                <entry>
                    <act classCode="ACT" moodCode="INT">
                        <id/>
                        <code code="223468009" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Teaching of skills">
                            <qualifier>
                                <name code="363702006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="has focus"/>
                                <value code="29893006" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Peak flow rate measurement"/>
                            </qualifier>
                        </code>
                    </act>
                </entry>
                <entry>
                    <substanceAdministration classCode="SBADM" moodCode="RQO">
                        <text>Уменьшить преднизон до 20 мг через день, чередуя с 18 мг через день.</text>
                        <routeCode code="PO" codeSystem="2.16.840.1.113883.5.112" codeSystemName="RouteOfA
dministration"/>
                        <consumable>
                            <manufacturedProduct>
                                <manufacturedLabeledDrug>
                                    <code code="10312003" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Prednisone preparation"/>
                                </manufacturedLabeledDrug>
                            </manufacturedProduct>
                        </consumable>
                    </substanceAdministration>
                </entry>
                <entry>
                    <substanceAdministration classCode="SBADM" moodCode="RQO">
                        <text>Гидрокортизон крем для пальца 2 раза в день.</text>
                        <effectiveTime xsi:type="PIVL_TS" institutionSpecified="true">
                            <period value="12" unit="h"/>
                        </effectiveTime>
                        <routeCode code="SKIN" codeSystem="2.16.840.1.113883.5.112" codeSystemName="RouteOf
Administration" displayName="Topical application, skin"/>
                        <approachSiteCode code="48856004" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Skin of palmer surface of index finger">
                            <qualifier>
```

```
                                <name code="78615007" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="with laterality"/>
                                <value code="7771000" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="left"/>
                            </qualifier>
                        </approachSiteCode>
                        <consumable>
                            <manufacturedProduct>
                                <manufacturedLabeledDrug>
                                    <code code="331646005" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Hydrocortisone cream"/>
                                </manufacturedLabeledDrug>
                            </manufacturedProduct>
                        </consumable>
                    </substanceAdministration>
                </entry>
                <entry>
                    <encounter classCode="ENC" moodCode="RQO">
                        <code code="185389009" codeSystem="2.16.840.1.113883.6.96"
codeSystemName="SNOMED CT" displayName="Follow-up visit"/>
                        <effectiveTime>
                            <low value="20000412"/>
                            <high value="20000417"/>
                        </effectiveTime>
                    </encounter>
                </entry>
            </section>
        </component>
    </structuredBody>
  </component>
</ClinicalDocument>
```

E.2 Пример XSLT-преобразования для визуализации АКД-документа

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:n3="http://www.w3.org/1999/
xhtml" xmlns:n1="urn:hl7-org:v3" xmlns:n2="urn:hl7-org:v3/meta/voc" xmlns:voc="urn:hl7-org:v3/voc" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance">
<xsl:output method="html" indent="yes" version="4.01" encoding="ISO-8859-1" doctype-public="-//W3C//DTD HTML
4.01//EN"/>

<!-- АКД-документ -->

<xsl:variable name="tableWidth">50%</xsl:variable>

<xsl:variable name="title">
  <xsl:choose>
    <xsl:when test="/n1:ClinicalDocument/n1:title">
      <xsl:value-of select="/n1:ClinicalDocument/n1:title"/>
    </xsl:when>
      <xsl:otherwise>Clinical Document</xsl:otherwise>
  </xsl:choose>
</xsl:variable>
```

```
<xsl:template match="/">
<xsl:apply-templates select="n1:ClinicalDocument"/>
</xsl:template>

<xsl:template match="n1:ClinicalDocument">
    <html>
      <head>
      <!-- <meta name='Generator' content='&CDA-Stylesheet;'/> -->
        <xsl:comment>
            Не редактируйте этот HTML-файл, он получен с помощью XSLt
                преобразования из исходного документа АКД, выпуск 2.
        </xsl:comment>
        <title>
          <xsl:value-of select="$title"/>
        </title>
     </head>
      <xsl:comment>
        За основу взято преобразоване HL7 Finland R2 Tyylitiedosto: Tyyli_R2_B3_01.xslt
        Отредактировано:  Calvin E. Beebe,  Mayo Clinic - Rochester Mn.
      </xsl:comment>
      <body>

        <h2 align="center"><xsl:value-of select="$title"/></h2>
      <table width='100%'>
        <tr><td width='10%'><big><b><xsl:text>Пациент: </xsl:text></b></big></td>
          <td width='40%'><big><xsl:call-template name="getName">
                  <xsl:with-param name="name"
                select="/n1:ClinicalDocument/n1:recordTarget/n1:patientRole/n1:patient/n1:name"/>
                 </xsl:call-template></big></td>
            <td width='25%' align='right'><b><xsl:text>М/карта: </xsl:text></b></td>
          <td width='25%'><xsl:value-of select="/n1:ClinicalDocument/n1:recordTarget/n1:patientRole/n1:id/@
extension"/></td>
        </tr>

        <tr><td width='10%'><b><xsl:text>Дата рождения: </xsl:text></b></td>
            <td width='40%'><xsl:call-template name="formatDate">
                  <xsl:with-param name="date"
              select="/n1:ClinicalDocument/n1:recordTarget/n1:patientRole/n1:patient/n1:birthTime/@value"/>
                </xsl:call-template></td>
          <td width='25%' align='right'><b><xsl:text>Пол: </xsl:text></b></td>
        <td width='25%'><xsl:variable name="sex"
                  select="/n1:ClinicalDocument/n1:recordTarget/n1:patientRole/n1:patient/
n1:administrativeGenderCode/@code"/>
              <xsl:choose>
                <xsl:when test="$sex='M'">Муж</xsl:when>
                <xsl:when test="$sex='F'">Жен</xsl:when>
              </xsl:choose></td>
        </tr>
```

```
            <tr><td width='10%'><b><xsl:text>Консультант: </xsl:text></b></td>
            <td width='40%'>

                <xsl:choose>
                    <xsl:when test="/n1:ClinicalDocument/n1:responsibleParty/n1:assignedEntity/n1:assignedPerson/
n1:name">
                     <xsl:call-template name="getName">
                     <xsl:with-param name="name"
                        select="/n1:ClinicalDocument/n1:responsibleParty/n1:assignedEntity/n1:assignedPerson/
n1:name"/>
                    </xsl:call-template>
                    </xsl:when>
                    <xsl:otherwise>
                      <xsl:call-template name="getName">
                     <xsl:with-param name="name"
                        select="/n1:ClinicalDocument/n1:legalAuthenticator/n1:assignedEntity/n1:assignedPerson/
n1:name"/>
                    </xsl:call-template>
                    </xsl:otherwise>
                  </xsl:choose> </td>
                <td width='25%' align='right'><b><xsl:text>Дата: </xsl:text></b></td>
              <td width='25%'><xsl:call-template name="formatDate">
                        <xsl:with-param name="date"
                    select="/n1:ClinicalDocument/n1:effectiveTime/@value"/>
                  </xsl:call-template></td>
          </tr>
        </table>
        <br/>
        <xsl:apply-templates select="n1:component/n1:structuredBody"/>
        <xsl:call-template name="bottomline"/>
         </body>
      </html>
</xsl:template>

<!– Извлечь ФИО –>
<xsl:template name="getName">
   <xsl:param name="name"/>
   <xsl:choose>
      <xsl:when test="$name/n1:family">
         <xsl:value-of select="$name/n1:given"/>
         <xsl:text> </xsl:text>
         <xsl:value-of select="$name/n1:family"/>
         <xsl:text> </xsl:text>
         <xsl:if test="$name/n1:suffix">
            <xsl:text>, </xsl:text>
            <xsl:value-of select="$name/n1:suffix"/>
         </xsl:if>
       </xsl:when>
       <xsl:otherwise>
          <xsl:value-of select="$name"/>
```

```
        </xsl:otherwise>
    </xsl:choose>
</xsl:template>

<!-- Формат даты

Выводит дату в формате «месяц, день, год»
     например, 19991207  ==> декабрь 07, 1999
-->
<xsl:template name="formatDate">
       <xsl:param name="date"/>
       <xsl:variable name="month" select="substring ($date, 5, 2)"/>
       <xsl:choose>
            <xsl:when test="$month='01'">
                 <xsl:text>январь </xsl:text>
            </xsl:when>
            <xsl:when test="$month='02'">
                 <xsl:text>февраль </xsl:text>
            </xsl:when>
            <xsl:when test="$month='03'">
                 <xsl:text>март </xsl:text>
            </xsl:when>
            <xsl:when test="$month='04'">
                 <xsl:text>апрель </xsl:text>
            </xsl:when>
            <xsl:when test="$month='05'">
                 <xsl:text>май </xsl:text>
            </xsl:when>
            <xsl:when test="$month='06'">
                 <xsl:text>июнь </xsl:text>
            </xsl:when>
            <xsl:when test="$month='07'">
                 <xsl:text>июль </xsl:text>
            </xsl:when>
            <xsl:when test="$month='08'">
                 <xsl:text>август </xsl:text>
            </xsl:when>
            <xsl:when test="$month='09'">
                 <xsl:text>сентябрь </xsl:text>
            </xsl:when>
            <xsl:when test="$month='10'">
                 <xsl:text>октябрь </xsl:text>
            </xsl:when>
            <xsl:when test="$month='11'">
                 <xsl:text>ноябрь </xsl:text>
            </xsl:when>
            <xsl:when test="$month='12'">
                 <xsl:text>декабрь </xsl:text>
            </xsl:when>
```

```
        </xsl:choose>
        <xsl:choose>
            <xsl:when test='substring ($date, 7, 1)="0"'>
                <xsl:value-of select="substring ($date, 8, 1)"/>
                <xsl:text>, </xsl:text>
            </xsl:when>
            <xsl:otherwise>
                <xsl:value-of select="substring ($date, 7, 2)"/>
                <xsl:text>, </xsl:text>
            </xsl:otherwise>
        </xsl:choose>
        <xsl:value-of select="substring ($date, 1, 4)"/>
</xsl:template>

<!-- Структурированное тело -->
<xsl:template match="n1:component/n1:structuredBody">
          <xsl:apply-templates select="n1:component/n1:section"/>
</xsl:template>

<!-- Компонент/раздел -->
<xsl:template match="n1:component/n1:section">
     <xsl:apply-templates select="n1:title"/>

     <xsl:apply-templates select="n1:text"/>

       <xsl:apply-templates select="n1:component/n1:section"/>


</xsl:template>

<!--   Заголовок  -->
<xsl:template match="n1:title">
     <h3><span style="font-weight:bold;">
     <xsl:value-of select="."></xsl:value-of>
     </span></h3>
</xsl:template>

<!--   Текст   -->
<xsl:template match="n1:text">
     <xsl:apply-templates />
</xsl:template>

<!--   абзац  -->
<xsl:template match="n1:paragraph">
     <xsl:apply-templates/>
     <br/>
</xsl:template>

<!--     Содержание с удаленным текстом скрыто -->
<xsl:template match="n1:content[@revised='delete']"/>
```

```
<!-- содержание -->
<xsl:template match="n1:content">
    <xsl:apply-templates/>
</xsl:template>

<!-- Iсписок -->
<xsl:template match="n1:list">
   <xsl:if test="n1:caption">
      <span style="font-weight:bold; ">
      <xsl:apply-templates select="n1:caption"/>
      </span>
   </xsl:if>
  <ul>
   <xsl:for-each select="n1:item">
    <li>
       <xsl:apply-templates />
    </li>
   </xsl:for-each>
   </ul>
</xsl:template>

<xsl:template match="n1:list[@listType='ordered']">
   <xsl:if test="n1:caption">
      <span style="font-weight:bold; ">
      <xsl:apply-templates select="n1:caption"/>
      </span>
   </xsl:if>
  <ol>
   <xsl:for-each select="n1:item">
    <li>
       <xsl:apply-templates />
    </li>
   </xsl:for-each>
   </ol>
</xsl:template>


<!-- заголовок -->
<xsl:template match="n1:caption">
    <xsl:apply-templates/>
    <xsl:text>: </xsl:text>
</xsl:template>

    <!--    Таблицы   -->
    <xsl:template match="n1:table/@*|n1:thead/@*|n1:tfoot/@*|n1:tbody/@*|n1:colgroup/@*|n1:col/@*|n1:tr/@*|n1:th/@*
|n1:td/@*">
        <xsl:copy>
            <xsl:apply-templates/>
```

```
        </xsl:copy>
    </xsl:template>

    <xsl:template match="n1:table">
        <table>
            <xsl:apply-templates/>
        </table>
    </xsl:template>

    <xsl:template match="n1:thead">
        <thead>
            <xsl:apply-templates/>
        </thead>
    </xsl:template>

    <xsl:template match="n1:tfoot">
        <tfoot>
            <xsl:apply-templates/>
        </tfoot>
    </xsl:template>

    <xsl:template match="n1:tbody">
        <tbody>
            <xsl:apply-templates/>
        </tbody>
    </xsl:template>

    <xsl:template match="n1:colgroup">
        <colgroup>
            <xsl:apply-templates/>
        </colgroup>
    </xsl:template>

    <xsl:template match="n1:col">
        <col>
            <xsl:apply-templates/>
        </col>
    </xsl:template>

    <xsl:template match="n1:tr">
        <tr>
            <xsl:apply-templates/>
        </tr>
    </xsl:template>

    <xsl:template match="n1:th">
        <th>
            <xsl:apply-templates/>
        </th>
    </xsl:template>
```

```
    <xsl:template match="n1:td">
        <td>
            <xsl:apply-templates/>
        </td>
    </xsl:template>

    <xsl:template match="n1:table/n1:caption">
        <span style="font-weight:bold; ">
            <xsl:apply-templates/>
        </span>
    </xsl:template>

<!--  Отображение мультимедийных данных

      В настоящее время отображает только форматы GIF и JPEG. Может быть
     расширено на другие типы MIME изображений в предикате
     и/или генерируя тег  <object> либо <applet> с правильными
     параметрами, зависящими от типа среды MIME  @ID  =$imageRef     referencedObject
 -->
    <xsl:template match="n1:renderMultiMedia">
    <xsl:variable name="imageRef" select="@referencedObject"/>
      <xsl:choose>
        <xsl:when test="//n1:regionOfInterest[@ID=$imageRef]">
        <!-- Сюда будет включено изображение области интереса -->
            <xsl:if test='//n1:regionOfInterest[@ID=$imageRef]//n1:observationMedia/n1:value[@mediaType="image/gif"
or @mediaType="image/jpeg"]'>
             <br clear='all'/>
              <xsl:element name='img'>
                <xsl:attribute name='src'>graphics/
                 <xsl:value-of select='//n1:regionOfInterest[@ID=$imageRef]//n1:observationMedia/n1:value/
n1:reference/@value'/>
                </xsl:attribute>
              </xsl:element>
            </xsl:if>
          </xsl:when>
          <xsl:otherwise>
          <!-- Сюда будет включено само изображение -->
             <xsl:if test='//n1:observationMedia[@ID=$imageRef]/n1:value[@mediaType="image/gif" or @
mediaType="image/jpeg"]'>
             <br clear='all'/>
              <xsl:element name='img'>
                <xsl:attribute name='src'>graphics/
                 <xsl:value-of select='//n1:observationMedia[@ID=$imageRef]/n1:value/n1:reference/@value'/>
                </xsl:attribute>
              </xsl:element>
            </xsl:if>
          </xsl:otherwise>
      </xsl:choose>
    </xsl:template>
```

```
<!--    Обработка кодов стиля Stylecode.
      Поддерживает полужирный (Bold), подчеркивание (Underline) и курсив (Italics)

-->

    <xsl:template match="//n1:*[@styleCode]">

    <xsl:if test="@styleCode='Bold'">
        <xsl:element name='b'>
            <xsl:apply-templates/>
        </xsl:element>
    </xsl:if>

    <xsl:if test="@styleCode='Italics'">
        <xsl:element name='i'>
            <xsl:apply-templates/>
        </xsl:element>
    </xsl:if>

    <xsl:if test="@styleCode='Underline'">
        <xsl:element name='u'>
            <xsl:apply-templates/>
        </xsl:element>
    </xsl:if>

      <xsl:if test="contains(@styleCode,'Bold') and contains(@styleCode,'Italics') and not (contains(@styleCode,
'Underline'))">
        <xsl:element name='b'>
         <xsl:element name='i'>
            <xsl:apply-templates/>
         </xsl:element>
        </xsl:element>
      </xsl:if>

      <xsl:if test="contains(@styleCode,'Bold') and contains(@styleCode,'Underline') and not (contains(@styleCode,
'Italics'))">
        <xsl:element name='b'>
         <xsl:element name='u'>
            <xsl:apply-templates/>
         </xsl:element>
        </xsl:element>
      </xsl:if>

      <xsl:if test="contains(@styleCode,'Italics') and contains(@styleCode,'Underline') and not (contains(@styleCode,
'Bold'))">
        <xsl:element name='i'>
         <xsl:element name='u'>
            <xsl:apply-templates/>
         </xsl:element>
        </xsl:element>
      </xsl:if>
```

```
        <xsl:if test="contains(@styleCode,'Italics') and contains(@styleCode,'Underline') and contains(@styleCode,
'Bold')">
          <xsl:element name='b'>
          <xsl:element name='i'>
          <xsl:element name='u'>
               <xsl:apply-templates/>
          </xsl:element>
          </xsl:element>
          </xsl:element>
        </xsl:if>

    </xsl:template>

<!--     Верхний или нижний индекс   -->
    <xsl:template match="n1:sup">
        <xsl:element name='sup'>
            <xsl:apply-templates/>
        </xsl:element>
    </xsl:template>
    <xsl:template match="n1:sub">
        <xsl:element name='sub'>
            <xsl:apply-templates/>
        </xsl:element>
    </xsl:template>

    <!-- Нижняя строка  -->
    <xsl:template name="bottomline">
    <b><xsl:text>Подпись: </xsl:text></b>
    <xsl:call-template name="getName">
        <xsl:with-param name="name"
            select="/n1:ClinicalDocument/n1:legalAuthenticator/n1:assignedEntity/n1:assignedPerson/n1:name"/>
      </xsl:call-template>
      <xsl:text> Дата: </xsl:text>
          <xsl:call-template name="formatDate">
              <xsl:with-param name="date"
                  select="//n1:ClinicalDocument/n1:legalAuthenticator/n1:time/@value"/>
      </xsl:call-template>
    </xsl:template>

</xsl:stylesheet>
```

Приложение F
(справочное)

XML-схемы

F.1 XML-схема cda.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSPY v2004 rel. 3 U (http://www.xmlspy.com) by Bob Dolin (HL7 CDA TC) -->
<xs:schema elementFormDefault="qualified" xmlns:mif="urn:hl7-org:v3/mif" xmlns="urn:hl7-org:v3" xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="urn:hl7-org:v3">
    <xs:include schemaLocation="POCD_MT000040.xsd"/>
    <xs:element type="POCD_MT000040.ClinicalDocument" name="ClinicalDocument"/>
</xs:schema>
```

F.2 XML-схема Datatypes.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- $Id: $ -->
<!--
  This schema is generated from a Generic Schema Definition (GSD)
  by gsd2xsl. Do not edit this file.
 -->
<xs:schema xmlns:sch="http://www.ascc.net/xml/schematron" xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
    <xs:annotation>
        <xs:documentation>
        Copyright (c) 2001, 2002, 2003, 2004, 2005 Health Level Seven.
        All rights reserved.

        Redistribution and use in source and binary forms, with or
        without modification, are permitted provided that the following
        conditions are met:
        1. Redistributions of source code must retain the above
           copyright notice, this list of conditions and the following
           disclaimer.
        2. Redistributions in binary form must reproduce the above
           copyright notice, this list of conditions and the following
           disclaimer in the documentation and/or other materials
           provided with the distribution.
        3. All advertising materials mentioning features or use of this
           software must display the following acknowledgement:

        This product includes software developed by Health Level Seven.

        THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS
        ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT
        NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
        FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.  IN NO EVENT
        SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
```

```
        INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
        DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
        GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
        INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
        WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
        NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
        OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
        DAMAGE.

        Generated by $Id: gsd2xsd.xsl,v 1.4 2005/04/17 03:20:15 lmckenzi Exp $
</xs:documentation>
    </xs:annotation>
    <xs:include schemaLocation="datatypes-base.xsd"/>
    <!--
    Instantiated templates
   -->
    <xs:complexType name="PIVL_TS">
        <xs:annotation>
            <xs:documentation>
        Note: because this type is defined as an extension of SXCM_T,
        all of the attributes and elements accepted for T are also
        accepted by this definition.  However, they are NOT allowed
        by the normative description of this type.  Unfortunately,
        we cannot write a general purpose schematron contraints to
        provide that extra validation, thus applications must be
        aware that instance (fragments) that pass validation with
        this might might still not be legal.
      </xs:documentation>
        </xs:annotation>
        <xs:complexContent>
            <xs:extension base="SXCM_TS">
                <xs:sequence>
                    <xs:element name="phase" type="IVL_TS" minOccurs="0" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>
                A prototype of the repeating interval specifying the
                duration of each occurrence and anchors the periodic
                interval sequence at a certain point in time.
              </xs:documentation>
                        </xs:annotation>
                    </xs:element>
                    <xs:element name="period" type="PQ" minOccurs="0" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>
                A time duration specifying a reciprocal measure of
                the frequency at which the periodic interval repeats.
              </xs:documentation>
                        </xs:annotation>
                    </xs:element>
```

```
            </xs:sequence>
            <xs:attribute name="alignment" type="CalendarCycle" use="optional">
                <xs:annotation>
                    <xs:documentation>
                Specifies if and how the repetitions are aligned to
                the cycles of the underlying calendar (e.g., to
                distinguish every 30 days from "the 5th of every
                month".) A non-aligned periodic interval recurs
                independently from the calendar. An aligned periodic
                interval is synchronized with the calendar.
            </xs:documentation>
                </xs:annotation>
            </xs:attribute>
            <xs:attribute name="institutionSpecified" type="bl" use="optional" default="false">
                <xs:annotation>
                    <xs:documentation>
                Indicates whether the exact timing is up to the party
                executing the schedule (e.g., to distinguish "every 8
                hours" from "3 times a day".)
            </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="EIVL_TS">
    <xs:annotation>
        <xs:documentation>
    Note: because this type is defined as an extension of SXCM_T,
    all of the attributes and elements accepted for T are also
    accepted by this definition.  However, they are NOT allowed
    by the normative description of this type.  Unfortunately,
    we cannot write a general purpose schematron contraints to
    provide that extra validation, thus applications must be
    aware that instance (fragments) that pass validation with
    this might might still not be legal.
  </xs:documentation>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="SXCM_TS">
            <xs:sequence>
                <xs:element name="event" type="EIVL.event" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
                A code for a common (periodical) activity of daily
                living based on which the event related periodic
                interval is specified.
            </xs:documentation>
                    </xs:annotation>
```

```
            </xs:element>
            <xs:element name="offset" type="IVL_PQ" minOccurs="0" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
          An interval of elapsed time (duration, not absolute
          point in time) that marks the offsets for the
          beginning, width and end of the event-related periodic
          interval measured from the time each such event
          actually occurred.
        </xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="IVL_PQ">
  <xs:complexContent>
    <xs:extension base="SXCM_PQ">
      <xs:choice minOccurs="0">
        <xs:sequence>
          <xs:element name="low" type="IVXB_PQ" minOccurs="1" maxOccurs="1">
            <xs:annotation>
              <xs:documentation>
          The low limit of the interval.
        </xs:documentation>
              </xs:annotation>
          </xs:element>
          <xs:choice minOccurs="0">
            <xs:element name="width" type="PQ" minOccurs="0" maxOccurs="1">
              <xs:annotation>
                <xs:documentation>
          The difference between high and low boundary. The
          purpose of distinguishing a width property is to
          handle all cases of incomplete information
          symmetrically. In any interval representation only
          two of the three properties high, low, and width need
          to be stated and the third can be derived.
        </xs:documentation>
              </xs:annotation>
            </xs:element>
            <xs:element name="high" type="IVXB_PQ" minOccurs="0" maxOccurs="1">
              <xs:annotation>
                <xs:documentation>
          The high limit of the interval.
        </xs:documentation>
              </xs:annotation>
            </xs:element>
          </xs:choice>
```

```
    </xs:sequence>
    <xs:element name="high" type="IVXB_PQ" minOccurs="1" maxOccurs="1">
        <xs:annotation>
            <xs:documentation/>
        </xs:annotation>
    </xs:element>
    <xs:sequence>
        <xs:element name="width" type="PQ" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>
  The difference between high and low boundary. The
  purpose of distinguishing a width property is to
  handle all cases of incomplete information
  symmetrically. In any interval representation only
  two of the three properties high, low, and width need
  to be stated and the third can be derived.
</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="high" type="IVXB_PQ" minOccurs="0" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>
  The high limit of the interval.
</xs:documentation>
            </xs:annotation>
        </xs:element>
    </xs:sequence>
    <xs:sequence>
        <xs:element name="center" type="PQ" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>
  The arithmetic mean of the interval (low plus high
  divided by 2). The purpose of distinguishing the center
  as a semantic property is for conversions of intervals
  from and to point values.
</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="width" type="PQ" minOccurs="0" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>
  The difference between high and low boundary. The
  purpose of distinguishing a width property is to
  handle all cases of incomplete information
  symmetrically. In any interval representation only
  two of the three properties high, low, and width need
  to be stated and the third can be derived.
</xs:documentation>
            </xs:annotation>
```

```
                </xs:element>
            </xs:sequence>
        </xs:choice>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="SXCM_PQ">
    <xs:complexContent>
        <xs:extension base="PQ">
            <xs:attribute name="operator" type="SetOperator" use="optional" default="I">
                <xs:annotation>
                    <xs:documentation>
          A code specifying whether the set component is included
          (union) or excluded (set-difference) from the set, or
          other set operations with the current set component and
          the set as constructed from the representation stream
          up to the current point.
         </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="IVXB_PQ">
    <xs:complexContent>
        <xs:extension base="PQ">
            <xs:attribute name="inclusive" type="bl" use="optional" default="true">
                <xs:annotation>
                    <xs:documentation>
          Specifies whether the limit is included in the
          interval (interval is closed) or excluded from the
          interval (interval is open).
         </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="PPD_TS">
    <xs:annotation>
        <xs:appinfo>
            <diff>PPD_PQ</diff>
        </xs:appinfo>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="TS">
            <xs:sequence>
                <xs:element name="standardDeviation" type="PQ" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
```

```
                    <xs:documentation>
            The primary measure of variance/uncertainty of the
            value (the square root of the sum of the squares of
            the differences between all data points and the mean).
            The standard deviation is used to normalize the data
            for computing the distribution function. Applications
            that cannot deal with probability distributions can
            still get an idea about the confidence level by looking
            at the standard deviation.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
            <xs:attribute name="distributionType" type="ProbabilityDistributionType" use="optional">
                <xs:annotation>
                    <xs:documentation>
            A code specifying the type of probability distribution.
            Possible values are as shown in the attached table.
            The NULL value (unknown) for the type code indicates
            that the probability distribution type is unknown. In
            that case, the standard deviation has the meaning of an
            informal guess.
        </xs:documentation>
                    </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="PPD_PQ">
    <xs:annotation>
        <xs:appinfo>
            <diff>PPD_PQ</diff>
        </xs:appinfo>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="PQ">
            <xs:sequence>
                <xs:element name="standardDeviation" type="PQ" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
            The primary measure of variance/uncertainty of the
            value (the square root of the sum of the squares of
            the differences between all data points and the mean).
            The standard deviation is used to normalize the data
            for computing the distribution function. Applications
            that cannot deal with probability distributions can
            still get an idea about the confidence level by looking
            at the standard deviation.
            </xs:documentation>
```

```
                </xs:annotation>
            </xs:element>
        </xs:sequence>
        <xs:attribute name="distributionType" type="ProbabilityDistributionType" use="optional">
            <xs:annotation>
                <xs:documentation>
    A code specifying the type of probability distribution.
    Possible values are as shown in the attached table.
    The NULL value (unknown) for the type code indicates
    that the probability distribution type is unknown. In
    that case, the standard deviation has the meaning of an
    informal guess.
     </xs:documentation>
            </xs:annotation>
        </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="PIVL_PPD_TS">
  <xs:annotation>
    <xs:documentation>
    Note: because this type is defined as an extension of SXCM_T,
    all of the attributes and elements accepted for T are also
    accepted by this definition.  However, they are NOT allowed
    by the normative description of this type.  Unfortunately,
    we cannot write a general purpose schematron contraints to
    provide that extra validation, thus applications must be
    aware that instance (fragments) that pass validation with
    this might might still not be legal.
  </xs:documentation>
  </xs:annotation>
  <xs:complexContent>
    <xs:extension base="SXCM_PPD_TS">
      <xs:sequence>
        <xs:element name="phase" type="IVL_PPD_TS" minOccurs="0" maxOccurs="1">
          <xs:annotation>
            <xs:documentation>
      A prototype of the repeating interval specifying the
      duration of each occurrence and anchors the periodic
      interval sequence at a certain point in time.
       </xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="period" type="PPD_PQ" minOccurs="0" maxOccurs="1">
          <xs:annotation>
            <xs:documentation>
      A time duration specifying a reciprocal measure of
      he frequency at which the periodic interval repeats.
       </xs:documentation>
```

```
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
            <xs:attribute name="alignment" type="CalendarCycle" use="optional">
                <xs:annotation>
                    <xs:documentation>
            Specifies if and how the repetitions are aligned to
            the cycles of the underlying calendar (e.g., to
            distinguish every 30 days from "the 5th of every
            month".) A non-aligned periodic interval recurs
            independently from the calendar. An aligned periodic
            interval is synchronized with the calendar.
        </xs:documentation>
                </xs:annotation>
            </xs:attribute>
            <xs:attribute name="institutionSpecified" type="bl" use="optional" default="false">
                <xs:annotation>
                    <xs:documentation>
            Indicates whether the exact timing is up to the party
            executing the schedule (e.g., to distinguish "every 8
            hours" from "3 times a day".)
        </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="SXCM_PPD_TS">
    <xs:complexContent>
        <xs:extension base="PPD_TS">
            <xs:attribute name="operator" type="SetOperator" use="optional" default="I">
                <xs:annotation>
                    <xs:documentation>
            A code specifying whether the set component is included
            (union) or excluded (set-difference) from the set, or
            other set operations with the current set component and
            the set as constructed from the representation stream
            up to the current point.
        </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="IVL_PPD_TS">
    <xs:complexContent>
        <xs:extension base="SXCM_PPD_TS">
            <xs:choice minOccurs="0">
                <xs:sequence>
```

```
        <xs:element name="low" type="IVXB_PPD_TS" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>
  The low limit of the interval.
</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:choice minOccurs="0">
            <xs:element name="width" type="PPD_PQ" minOccurs="0" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
  The difference between high and low boundary. The
  purpose of distinguishing a width property is to
  handle all cases of incomplete information
  symmetrically. In any interval representation only
  two of the three properties high, low, and width need
  to be stated and the third can be derived.
</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="high" type="IVXB_PPD_TS" minOccurs="0" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
  The high limit of the interval.
</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:choice>
    </xs:sequence>
    <xs:element name="high" type="IVXB_PPD_TS" minOccurs="1" maxOccurs="1">
        <xs:annotation>
            <xs:documentation/>
        </xs:annotation>
    </xs:element>
    <xs:sequence>
        <xs:element name="width" type="PPD_PQ" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>
  The difference between high and low boundary. The
  purpose of distinguishing a width property is to
  handle all cases of incomplete information
  symmetrically. In any interval representation only
  two of the three properties high, low, and width need
  to be stated and the third can be derived.
</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="high" type="IVXB_PPD_TS" minOccurs="0" maxOccurs="1">
            <xs:annotation>
```

```
                    <xs:documentation>
                The high limit of the interval.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
            <xs:sequence>
                <xs:element name="center" type="PPD_TS" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
                The arithmetic mean of the interval (low plus high
                divided by 2). The purpose of distinguishing the center
                as a semantic property is for conversions of intervals
                from and to point values.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="width" type="PPD_PQ" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
                The difference between high and low boundary. The
                purpose of distinguishing a width property is to
                handle all cases of incomplete information
                symmetrically. In any interval representation only
                two of the three properties high, low, and width need
                to be stated and the third can be derived.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:choice>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="IVXB_PPD_TS">
    <xs:complexContent>
        <xs:extension base="PPD_TS">
            <xs:attribute name="inclusive" type="bl" use="optional" default="true">
                <xs:annotation>
                    <xs:documentation>
            Specifies whether the limit is included in the
            interval (interval is closed) or excluded from the
            interval (interval is open).
        </xs:documentation>
                    </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
```

```
<xs:complexType name="EIVL_PPD_TS">
    <xs:annotation>
        <xs:documentation>
    Note: because this type is defined as an extension of SXCM_T,
    all of the attributes and elements accepted for T are also
    accepted by this definition.  However, they are NOT allowed
    by the normative description of this type.  Unfortunately,
    we cannot write a general purpose schematron contraints to
    provide that extra validation, thus applications must be
    aware that instance (fragments) that pass validation with
    this might might still not be legal.
  </xs:documentation>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="SXCM_PPD_TS">
            <xs:sequence>
                <xs:element name="event" type="EIVL.event" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
            A code for a common (periodical) activity of daily
            living based on which the event related periodic
            interval is specified.
          </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="offset" type="IVL_PPD_PQ" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
            An interval of elapsed time (duration, not absolute
            point in time) that marks the offsets for the
            beginning, width and end of the event-related periodic
            interval measured from the time each such event
            actually occurred.
          </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="IVL_PPD_PQ">
    <xs:complexContent>
        <xs:extension base="SXCM_PPD_PQ">
            <xs:choice minOccurs="0">
                <xs:sequence>
                    <xs:element name="low" type="IVXB_PPD_PQ" minOccurs="1" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>
              The low limit of the interval.
```

```
</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:choice minOccurs="0">
            <xs:element name="width" type="PPD_PQ" minOccurs="0" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
  The difference between high and low boundary. The
  purpose of distinguishing a width property is to
  handle all cases of incomplete information
  symmetrically. In any interval representation only
  two of the three properties high, low, and width need
  to be stated and the third can be derived.
</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="high" type="IVXB_PPD_PQ" minOccurs="0" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
  The high limit of the interval.
</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:choice>
    </xs:sequence>
    <xs:element name="high" type="IVXB_PPD_PQ" minOccurs="1" maxOccurs="1">
        <xs:annotation>
            <xs:documentation/>
        </xs:annotation>
    </xs:element>
    <xs:sequence>
        <xs:element name="width" type="PPD_PQ" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>
  The difference between high and low boundary. The
  purpose of distinguishing a width property is to
  handle all cases of incomplete information
  symmetrically. In any interval representation only
  two of the three properties high, low, and width need
  to be stated and the third can be derived.
</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="high" type="IVXB_PPD_PQ" minOccurs="0" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>
  The high limit of the interval.
</xs:documentation>
            </xs:annotation>
```

```
                </xs:element>
            </xs:sequence>
            <xs:sequence>
                <xs:element name="center" type="PPD_PQ" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
            The arithmetic mean of the interval (low plus high
            divided by 2). The purpose of distinguishing the center
            as a semantic property is for conversions of intervals
            from and to point values.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="width" type="PPD_PQ" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
            The difference between high and low boundary. The
            purpose of distinguishing a width property is to
            handle all cases of incomplete information
            symmetrically. In any interval representation only
            two of the three properties high, low, and width need
            to be stated and the third can be derived.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:choice>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="SXCM_PPD_PQ">
  <xs:complexContent>
    <xs:extension base="PPD_PQ">
      <xs:attribute name="operator" type="SetOperator" use="optional" default="I">
        <xs:annotation>
          <xs:documentation>
        A code specifying whether the set component is included
        (union) or excluded (set-difference) from the set, or
        other set operations with the current set component and
        the set as constructed from the representation stream
        up to the current point.
        </xs:documentation>
        </xs:annotation>
      </xs:attribute>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="IVXB_PPD_PQ">
  <xs:complexContent>
```

```
            <xs:extension base="PPD_PQ">
                <xs:attribute name="inclusive" type="bl" use="optional" default="true">
                    <xs:annotation>
                        <xs:documentation>
                    Specifies whether the limit is included in the
                    interval (interval is closed) or excluded from the
                    interval (interval is open).
                 </xs:documentation>
                    </xs:annotation>
                </xs:attribute>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="SXPR_TS">
        <xs:complexContent>
            <xs:extension base="SXCM_TS">
                <xs:sequence>
                    <xs:element name="comp" type="SXCM_TS" minOccurs="2" maxOccurs="unbounded">
                        <xs:annotation>
                            <xs:documentation/>
                        </xs:annotation>
                    </xs:element>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="SXCM_CD">
        <xs:complexContent>
            <xs:extension base="CD">
                <xs:attribute name="operator" type="SetOperator" use="optional" default="I">
                    <xs:annotation>
                        <xs:documentation>
                    A code specifying whether the set component is included
                    (union) or excluded (set-difference) from the set, or
                    other set operations with the current set component and
                    the set as constructed from the representation stream
                    up to the current point.
                 </xs:documentation>
                    </xs:annotation>
                </xs:attribute>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="SXCM_MO">
        <xs:complexContent>
            <xs:extension base="MO">
                <xs:attribute name="operator" type="SetOperator" use="optional" default="I">
                    <xs:annotation>
                        <xs:documentation>
```

```
            A code specifying whether the set component is included
            (union) or excluded (set-difference) from the set, or
            other set operations with the current set component and
            the set as constructed from the representation stream
            up to the current point.
          </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="SXCM_INT">
    <xs:complexContent>
        <xs:extension base="INT">
            <xs:attribute name="operator" type="SetOperator" use="optional" default="I">
                <xs:annotation>
                    <xs:documentation>
            A code specifying whether the set component is included
            (union) or excluded (set-difference) from the set, or
            other set operations with the current set component and
            the set as constructed from the representation stream
            up to the current point.
          </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="SXCM_REAL">
    <xs:complexContent>
        <xs:extension base="REAL">
            <xs:attribute name="operator" type="SetOperator" use="optional" default="I">
                <xs:annotation>
                    <xs:documentation>
            A code specifying whether the set component is included
            (union) or excluded (set-difference) from the set, or
            other set operations with the current set component and
            the set as constructed from the representation stream
            up to the current point.
          </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="IVL_INT">
    <xs:complexContent>
        <xs:extension base="SXCM_INT">
            <xs:choice minOccurs="0">
```

```
    <xs:sequence>
        <xs:element name="low" type="IVXB_INT" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>
  The low limit of the interval.
</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:choice minOccurs="0">
            <xs:element name="width" type="INT" minOccurs="0" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
  The difference between high and low boundary. The
  purpose of distinguishing a width property is to
  handle all cases of incomplete information
  symmetrically. In any interval representation only
  two of the three properties high, low, and width need
  to be stated and the third can be derived.
</xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="high" type="IVXB_INT" minOccurs="0" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
  The high limit of the interval.
</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:choice>
    </xs:sequence>
    <xs:element name="high" type="IVXB_INT" minOccurs="1" maxOccurs="1">
        <xs:annotation>
            <xs:documentation/>
        </xs:annotation>
    </xs:element>
    <xs:sequence>
        <xs:element name="width" type="INT" minOccurs="1" maxOccurs="1">
            <xs:annotation>
                <xs:documentation>
  The difference between high and low boundary. The
  purpose of distinguishing a width property is to
  handle all cases of incomplete information
  symmetrically. In any interval representation only
  two of the three properties high, low, and width need
  to be stated and the third can be derived.
</xs:documentation>
            </xs:annotation>
        </xs:element>
        <xs:element name="high" type="IVXB_INT" minOccurs="0" maxOccurs="1">
```

```
                    <xs:annotation>
                        <xs:documentation>
                The high limit of the interval.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
            <xs:sequence>
                <xs:element name="center" type="INT" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
                The arithmetic mean of the interval (low plus high
                divided by 2). The purpose of distinguishing the center
                as a semantic property is for conversions of intervals
                from and to point values.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="width" type="INT" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
                The difference between high and low boundary. The
                purpose of distinguishing a width property is to
                handle all cases of incomplete information
                symmetrically. In any interval representation only
                two of the three properties high, low, and width need
                to be stated and the third can be derived.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:choice>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="IVXB_INT">
    <xs:complexContent>
        <xs:extension base="INT">
            <xs:attribute name="inclusive" type="bl" use="optional" default="true">
                <xs:annotation>
                    <xs:documentation>
            Specifies whether the limit is included in the
            interval (interval is closed) or excluded from the
            interval (interval is open).
        </xs:documentation>
                    </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
```

```
</xs:complexType>
<xs:complexType name="IVL_REAL">
    <xs:complexContent>
        <xs:extension base="SXCM_REAL">
            <xs:choice minOccurs="0">
                <xs:sequence>
                    <xs:element name="low" type="IVXB_REAL" minOccurs="1" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>
                The low limit of the interval.
                </xs:documentation>
                        </xs:annotation>
                    </xs:element>
                    <xs:choice minOccurs="0">
                        <xs:element name="width" type="REAL" minOccurs="0" maxOccurs="1">
                            <xs:annotation>
                                <xs:documentation>
                The difference between high and low boundary. The
                purpose of distinguishing a width property is to
                handle all cases of incomplete information
                symmetrically. In any interval representation only
                two of the three properties high, low, and width need
                to be stated and the third can be derived.
                </xs:documentation>
                            </xs:annotation>
                        </xs:element>
                        <xs:element name="high" type="IVXB_REAL" minOccurs="0" maxOccurs="1">
                            <xs:annotation>
                                <xs:documentation>
                The high limit of the interval.
                </xs:documentation>
                            </xs:annotation>
                        </xs:element>
                    </xs:choice>
                </xs:sequence>
                <xs:element name="high" type="IVXB_REAL" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation/>
                    </xs:annotation>
                </xs:element>
                <xs:sequence>
                    <xs:element name="width" type="REAL" minOccurs="1" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>
                The difference between high and low boundary. The
                purpose of distinguishing a width property is to
                handle all cases of incomplete information
                symmetrically. In any interval representation only
                two of the three properties high, low, and width need
```

```
            to be stated and the third can be derived.
          </xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="high" type="IVXB_REAL" minOccurs="0" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
            The high limit of the interval.
          </xs:documentation>
                </xs:annotation>
            </xs:element>
          </xs:sequence>
          <xs:sequence>
            <xs:element name="center" type="REAL" minOccurs="1" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
            The arithmetic mean of the interval (low plus high
            divided by 2). The purpose of distinguishing the center
            as a semantic property is for conversions of intervals
            from and to point values.
          </xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="width" type="REAL" minOccurs="0" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
            The difference between high and low boundary. The
            purpose of distinguishing a width property is to
            handle all cases of incomplete information
            symmetrically. In any interval representation only
            two of the three properties high, low, and width need
            to be stated and the third can be derived.
          </xs:documentation>
                </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:choice>
      </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="IVXB_REAL">
    <xs:complexContent>
      <xs:extension base="REAL">
        <xs:attribute name="inclusive" type="bl" use="optional" default="true">
            <xs:annotation>
                <xs:documentation>
          Specifies whether the limit is included in the
          interval (interval is closed) or excluded from the
          interval (interval is open).
```

```
            </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="IVL_MO">
    <xs:complexContent>
        <xs:extension base="SXCM_MO">
            <xs:choice minOccurs="0">
                <xs:sequence>
                    <xs:element name="low" type="IVXB_MO" minOccurs="1" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>
                The low limit of the interval.
            </xs:documentation>
                        </xs:annotation>
                    </xs:element>
                    <xs:choice minOccurs="0">
                        <xs:element name="width" type="MO" minOccurs="0" maxOccurs="1">
                            <xs:annotation>
                                <xs:documentation>
                The difference between high and low boundary. The
                purpose of distinguishing a width property is to
                handle all cases of incomplete information
                symmetrically. In any interval representation only
                two of the three properties high, low, and width need
                to be stated and the third can be derived.
            </xs:documentation>
                            </xs:annotation>
                        </xs:element>
                        <xs:element name="high" type="IVXB_MO" minOccurs="0" maxOccurs="1">
                            <xs:annotation>
                                <xs:documentation>
                The high limit of the interval.
            </xs:documentation>
                            </xs:annotation>
                        </xs:element>
                    </xs:choice>
                </xs:sequence>
                <xs:element name="high" type="IVXB_MO" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation/>
                    </xs:annotation>
                </xs:element>
                <xs:sequence>
                    <xs:element name="width" type="MO" minOccurs="1" maxOccurs="1">
                        <xs:annotation>
                            <xs:documentation>
```

```
                The difference between high and low boundary. The
                purpose of distinguishing a width property is to
                handle all cases of incomplete information
                symmetrically. In any interval representation only
                two of the three properties high, low, and width need
                to be stated and the third can be derived.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="high" type="IVXB_MO" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
                The high limit of the interval.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
            <xs:sequence>
                <xs:element name="center" type="MO" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
                The arithmetic mean of the interval (low plus high
                divided by 2). The purpose of distinguishing the center
                as a semantic property is for conversions of intervals
                from and to point values.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="width" type="MO" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
                The difference between high and low boundary. The
                purpose of distinguishing a width property is to
                handle all cases of incomplete information
                symmetrically. In any interval representation only
                two of the three properties high, low, and width need
                to be stated and the third can be derived.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:choice>
    </xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="IVXB_MO">
    <xs:complexContent>
        <xs:extension base="MO">
            <xs:attribute name="inclusive" type="bl" use="optional" default="true">
```

```
                <xs:annotation>
                    <xs:documentation>
            Specifies whether the limit is included in the
            interval (interval is closed) or excluded from the
            interval (interval is open).
          </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="HXIT_PQ">
    <xs:complexContent>
        <xs:extension base="PQ">
            <xs:sequence>
                <xs:element name="validTime" type="IVL_TS" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
            The time interval during which the given information
            was, is, or is expected to be valid. The interval can
            be open or closed, as well as infinite or undefined on
            either side.
          </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="HXIT_CE">
    <xs:complexContent>
        <xs:extension base="CE">
            <xs:sequence>
                <xs:element name="validTime" type="IVL_TS" minOccurs="0" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
            The time interval during which the given information
            was, is, or is expected to be valid. The interval can
            be open or closed, as well as infinite or undefined on
            either side.
          </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="BXIT_CD">
    <xs:complexContent>
```

```
        <xs:extension base="CD">
            <xs:attribute name="qty" type="int" use="optional" default="1">
                <xs:annotation>
                    <xs:documentation>
          The quantity in which the bag item occurs in its containing bag.
        </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="BXIT_IVL_PQ">
    <xs:complexContent>
        <xs:extension base="IVL_PQ">
            <xs:attribute name="qty" type="int" use="optional" default="1">
                <xs:annotation>
                    <xs:documentation>
          The quantity in which the bag item occurs in its containing bag.
        </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="SLIST_PQ">
    <xs:complexContent>
        <xs:extension base="ANY">
            <xs:sequence>
                <xs:element name="origin" type="PQ" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
          The origin of the list item value scale, i.e., the
          physical quantity that a zero-digit in the sequence
          would represent.
        </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="scale" type="PQ" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
          A ratio-scale quantity that is factored out of the
          digit sequence.
        </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="digits" type="list_int" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
          A sequence of raw digits for the sample values. This is
```

```
            typically the raw output of an A/D converter.
          </xs:documentation>
                </xs:annotation>
            </xs:element>
          </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:simpleType name="list_int">
    <xs:list itemType="int"/>
</xs:simpleType>
<xs:complexType name="SLIST_TS">
    <xs:complexContent>
        <xs:extension base="ANY">
            <xs:sequence>
                <xs:element name="origin" type="TS" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
            The origin of the list item value scale, i.e., the
            physical quantity that a zero-digit in the sequence
            would represent.
          </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="scale" type="PQ" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
            A ratio-scale quantity that is factored out of the
            digit sequence.
          </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="digits" type="list_int" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
            A sequence of raw digits for the sample values. This is
            typically the raw output of an A/D converter.
          </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="GLIST_TS">
    <xs:complexContent>
        <xs:extension base="ANY">
            <xs:sequence>
                <xs:element name="head" type="TS" minOccurs="1" maxOccurs="1">
```

```
                <xs:annotation>
                    <xs:documentation>
          This is the start-value of the generated list.
        </xs:documentation>
                </xs:annotation>
            </xs:element>
            <xs:element name="increment" type="PQ" minOccurs="1" maxOccurs="1">
                <xs:annotation>
                    <xs:documentation>
          The difference between one value and its previous
          different value. For example, to generate the sequence
          (1; 4; 7; 10; 13; ...) the increment is 3; likewise to
          generate the sequence (1; 1; 4; 4; 7; 7; 10; 10; 13;
          13; ...) the increment is also 3.
        </xs:documentation>
                </xs:annotation>
            </xs:element>
          </xs:sequence>
          <xs:attribute name="period" type="int" use="optional">
                <xs:annotation>
                    <xs:documentation>
          If non-NULL, specifies that the sequence alternates,
          i.e., after this many increments, the sequence item
          values roll over to start from the initial sequence
          item value. For example, the sequence (1; 2; 3; 1; 2;
          3; 1; 2; 3; ...) has period 3; also the sequence
          (1; 1; 2; 2; 3; 3; 1; 1; 2; 2; 3; 3; ...) has period
          3 too.
        </xs:documentation>
                </xs:annotation>
          </xs:attribute>
          <xs:attribute name="denominator" type="int" use="optional">
                <xs:annotation>
                    <xs:documentation>
          The integer by which the index for the sequence is
          divided, effectively the number of times the sequence
          generates the same sequence item value before
          incrementing to the next sequence item value. For
          example, to generate the sequence (1; 1; 1; 2; 2; 2; 3; 3;
          3; ...)  the denominator is 3.
        </xs:documentation>
                </xs:annotation>
          </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="GLIST_PQ">
    <xs:complexContent>
        <xs:extension base="ANY">
```

```
            <xs:sequence>
                <xs:element name="head" type="PQ" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
          This is the start-value of the generated list.
        </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="increment" type="PQ" minOccurs="1" maxOccurs="1">
                    <xs:annotation>
                        <xs:documentation>
          The difference between one value and its previous
          different value. For example, to generate the sequence
          (1; 4; 7; 10; 13; ...) the increment is 3; likewise to
          generate the sequence (1; 1; 4; 4; 7; 7; 10; 10; 13;
          13; ...) the increment is also 3.
        </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
            <xs:attribute name="period" type="int" use="optional">
                <xs:annotation>
                    <xs:documentation>
          If non-NULL, specifies that the sequence alternates,
          i.e., after this many increments, the sequence item
          values roll over to start from the initial sequence
          item value. For example, the sequence (1; 2; 3; 1; 2;
          3; 1; 2; 3; ...) has period 3; also the sequence
          (1; 1; 2; 2; 3; 3; 1; 1; 2; 2; 3; 3; ...) has period
          3 too.
        </xs:documentation>
                </xs:annotation>
            </xs:attribute>
            <xs:attribute name="denominator" type="int" use="optional">
                <xs:annotation>
                    <xs:documentation>
          The integer by which the index for the sequence is
          divided, effectively the number of times the sequence
          generates the same sequence item value before
          incrementing to the next sequence item value. For
          example, to generate the sequence (1; 1; 1; 2; 2; 2; 3; 3;
          3; ...)  the denominator is 3.
        </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="RTO_PQ_PQ">
```

```
    <xs:annotation>
        <xs:appinfo>
            <diff>RTO_PQ_PQ</diff>
        </xs:appinfo>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="QTY">
            <xs:sequence>
                <xs:element name="numerator" type="PQ">
                    <xs:annotation>
                        <xs:documentation>
             The quantity that is being divided in the ratio.  The
             default is the integer number 1 (one).
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="denominator" type="PQ">
                    <xs:annotation>
                        <xs:documentation>
             The quantity that devides the numerator in the ratio.
             The default is the integer number 1 (one).
             The denominator must not be zero.
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="RTO_MO_PQ">
    <xs:annotation>
        <xs:appinfo>
            <diff>RTO_MO_PQ</diff>
        </xs:appinfo>
    </xs:annotation>
    <xs:complexContent>
        <xs:extension base="QTY">
            <xs:sequence>
                <xs:element name="numerator" type="MO">
                    <xs:annotation>
                        <xs:documentation>
             The quantity that is being divided in the ratio.  The
             default is the integer number 1 (one).
            </xs:documentation>
                    </xs:annotation>
                </xs:element>
                <xs:element name="denominator" type="PQ">
                    <xs:annotation>
                        <xs:documentation>
```

```
                The quantity that devides the numerator in the ratio.
                The default is the integer number 1 (one).
                The denominator must not be zero.
              </xs:documentation>
                    </xs:annotation>
                </xs:element>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:complexType name="UVP_TS">
    <xs:complexContent>
        <xs:extension base="TS">
            <xs:attribute name="probability" type="probability" use="optional">
                <xs:annotation>
                    <xs:documentation>
      The probability assigned to the value, a decimal number
      between 0 (very uncertain) and 1 (certain).
     </xs:documentation>
                </xs:annotation>
            </xs:attribute>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
</xs:schema>
```

F.3 XML-схема voc.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:mif="urn:hl7-org:v3/mif">
    <xs:annotation>
        <xs:documentation> $Id: Vocabulary.xml,v 1.1 2005/08/25 11:33:46 mcraig Exp $
RoseTree XML to Schema: $Id: VocabXMLtoXSD.xsl,v 1.6 2005/05/24 00:14:18 lmckenzi Exp $</xs:documentation>
    </xs:annotation>
    <xs:include schemaLocation="datatypes.xsd"/>
    <xs:simpleType name="Classes">
        <xs:restriction base="cs"/>
    </xs:simpleType>
    <xs:annotation>
        <xs:documentation>The following types are used internally in data types</xs:documentation>
    </xs:annotation>
    <xs:simpleType name="AddressPartType">
        <xs:annotation>
            <xs:documentation>vocSet: D10642 (C-0-D10642-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="AdditionalLocator DeliveryAddressLine StreetAddressLine">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="CAR"/>
                    <xs:enumeration value="CEN"/>
```

```
                <xs:enumeration value="CNT"/>
                <xs:enumeration value="CPA"/>
                <xs:enumeration value="CTY"/>
                <xs:enumeration value="DEL"/>
                <xs:enumeration value="POB"/>
                <xs:enumeration value="PRE"/>
                <xs:enumeration value="STA"/>
                <xs:enumeration value="ZIP"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="AdditionalLocator">
    <xs:annotation>
        <xs:documentation>specDomain: V10651 (C-0-D10642-V10651-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="ADL"/>
        <xs:enumeration value="UNID"/>
        <xs:enumeration value="UNIT"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="DeliveryAddressLine">
    <xs:annotation>
        <xs:documentation>specDomain: V17887 (C-0-D10642-V17887-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="DAL"/>
        <xs:enumeration value="DINST"/>
        <xs:enumeration value="DINSTA"/>
        <xs:enumeration value="DINSTQ"/>
        <xs:enumeration value="DMOD"/>
        <xs:enumeration value="DMODID"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="StreetAddressLine">
    <xs:annotation>
        <xs:documentation>specDomain: V14822 (C-0-D10642-V14822-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="BuildingNumber StreetName">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="SAL"/>
                <xs:enumeration value="DIR"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="BuildingNumber">
```

```
        <xs:annotation>
            <xs:documentation>specDomain: V10649 (C-0-D10642-V14822-V10649-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="BNR"/>
            <xs:enumeration value="BNN"/>
            <xs:enumeration value="BNS"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="StreetName">
        <xs:annotation>
            <xs:documentation>specDomain: V10648 (C-0-D10642-V14822-V10648-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="STR"/>
            <xs:enumeration value="STB"/>
            <xs:enumeration value="STTYP"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="CalendarCycle">
        <xs:annotation>
            <xs:documentation>vocSet: D10684 (C-0-D10684-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="CalendarCycleOneLetter CalendarCycleTwoLetter"/>
    </xs:simpleType>
    <xs:simpleType name="CalendarCycleOneLetter">
        <xs:annotation>
            <xs:documentation>abstDomain: V10701 (C-0-D10684-V10701-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="D"/>
            <xs:enumeration value="H"/>
            <xs:enumeration value="J"/>
            <xs:enumeration value="M"/>
            <xs:enumeration value="N"/>
            <xs:enumeration value="S"/>
            <xs:enumeration value="W"/>
            <xs:enumeration value="Y"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="CalendarCycleTwoLetter">
        <xs:annotation>
            <xs:documentation>abstDomain: V10685 (C-0-D10684-V10685-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="GregorianCalendarCycle">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="CD"/>
                    <xs:enumeration value="CH"/>
```

```
                <xs:enumeration value="CM"/>
                <xs:enumeration value="CN"/>
                <xs:enumeration value="CS"/>
                <xs:enumeration value="CW"/>
                <xs:enumeration value="CY"/>
                <xs:enumeration value="DM"/>
                <xs:enumeration value="DW"/>
                <xs:enumeration value="DY"/>
                <xs:enumeration value="HD"/>
                <xs:enumeration value="MY"/>
                <xs:enumeration value="NH"/>
                <xs:enumeration value="SN"/>
                <xs:enumeration value="WY"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="GregorianCalendarCycle">
    <xs:annotation>
        <xs:documentation>abstDomain: V10758 (C-0-D10684-V10685-V10758-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs"/>
</xs:simpleType>
<xs:simpleType name="CompressionAlgorithm">
    <xs:annotation>
        <xs:documentation>vocSet: D10620 (C-0-D10620-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="DF"/>
        <xs:enumeration value="GZ"/>
        <xs:enumeration value="Z"/>
        <xs:enumeration value="ZL"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Currency">
    <xs:annotation>
        <xs:documentation>vocSet: D17388 (C-0-D17388-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="ARS"/>
        <xs:enumeration value="AUD"/>
        <xs:enumeration value="BRL"/>
        <xs:enumeration value="CAD"/>
        <xs:enumeration value="CHF"/>
        <xs:enumeration value="CLF"/>
        <xs:enumeration value="CNY"/>
        <xs:enumeration value="DEM"/>
        <xs:enumeration value="ESP"/>
        <xs:enumeration value="EUR"/>
```

```
            <xs:enumeration value="FIM"/>
            <xs:enumeration value="FRF"/>
            <xs:enumeration value="GBP"/>
            <xs:enumeration value="ILS"/>
            <xs:enumeration value="INR"/>
            <xs:enumeration value="JPY"/>
            <xs:enumeration value="KRW"/>
            <xs:enumeration value="MXN"/>
            <xs:enumeration value="NLG"/>
            <xs:enumeration value="NZD"/>
            <xs:enumeration value="PHP"/>
            <xs:enumeration value="RUR"/>
            <xs:enumeration value="THB"/>
            <xs:enumeration value="TRL"/>
            <xs:enumeration value="TWD"/>
            <xs:enumeration value="USD"/>
            <xs:enumeration value="ZAR"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="EntityNamePartQualifier">
        <xs:annotation>
            <xs:documentation>vocSet: D15888 (C-0-D15888-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="OrganizationNamePartQualifier PersonNamePartQualifier"/>
    </xs:simpleType>
    <xs:simpleType name="OrganizationNamePartQualifier">
        <xs:annotation>
            <xs:documentation>abstDomain: V15889 (C-0-D15888-V15889-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="LS"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="PersonNamePartQualifier">
        <xs:annotation>
            <xs:documentation>abstDomain: V10659 (C-0-D15888-V10659-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="PersonNamePartAffixTypes PersonNamePartChangeQualifier
PersonNamePartMiscQualifier">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="IN"/>
                    <xs:enumeration value="TITLE"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="PersonNamePartAffixTypes">
        <xs:annotation>
```

```
            <xs:documentation>abstDomain: V10666 (C-0-D15888-V10659-V10666-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="AC"/>
            <xs:enumeration value="NB"/>
            <xs:enumeration value="PR"/>
            <xs:enumeration value="VV"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="PersonNamePartChangeQualifier">
        <xs:annotation>
            <xs:documentation>abstDomain: V10660 (C-0-D15888-V10659-V10660-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="AD"/>
            <xs:enumeration value="BR"/>
            <xs:enumeration value="SP"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="PersonNamePartMiscQualifier">
        <xs:annotation>
            <xs:documentation>abstDomain: V10671 (C-0-D15888-V10659-V10671-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="CL"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="EntityNamePartType">
        <xs:annotation>
            <xs:documentation>vocSet: D15880 (C-0-D15880-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="x_OrganizationNamePartType x_PersonNamePartType">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="DEL"/>
                    <xs:enumeration value="FAM"/>
                    <xs:enumeration value="GIV"/>
                    <xs:enumeration value="PFX"/>
                    <xs:enumeration value="SFX"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="x_OrganizationNamePartType">
        <xs:annotation>
            <xs:documentation>abstDomain: V15881 (C-0-D15880-V15881-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="DEL"/>
```

```
            <xs:enumeration value="PFX"/>
            <xs:enumeration value="SFX"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_PersonNamePartType">
        <xs:annotation>
            <xs:documentation>abstDomain: V10653 (C-0-D15880-V10653-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="DEL"/>
            <xs:enumeration value="FAM"/>
            <xs:enumeration value="GIV"/>
            <xs:enumeration value="PFX"/>
            <xs:enumeration value="SFX"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="EntityNameUse">
        <xs:annotation>
            <xs:documentation>vocSet: D15913 (C-0-D15913-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="EntityNameSearchUse NameRepresentationUse OrganizationNameUse
PersonNameUse">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="C"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="OrganizationNameUse">
        <xs:annotation>
            <xs:documentation>abstDomain: V15914 (C-0-D15913-V15914-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="EntityNameSearchUse NameRepresentationUse">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="C"/>
                    <xs:enumeration value="L"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="PersonNameUse">
        <xs:annotation>
            <xs:documentation>abstDomain: V200 (C-0-D15913-V200-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="EntityNameSearchUse NamePseudonymUse NameRepresentationUse">
            <xs:simpleType>
                <xs:restriction base="cs">
```

```
                <xs:enumeration value="A"/>
                <xs:enumeration value="ASGN"/>
                <xs:enumeration value="C"/>
                <xs:enumeration value="I"/>
                <xs:enumeration value="L"/>
                <xs:enumeration value="R"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="EntityNameSearchUse">
    <xs:annotation>
        <xs:documentation>specDomain: V19619 (C-0-D15913-V200-V19619-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="SRCH"/>
        <xs:enumeration value="PHON"/>
        <xs:enumeration value="SNDX"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="NamePseudonymUse">
    <xs:annotation>
        <xs:documentation>specDomain: V19591 (C-0-D15913-V200-V19591-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="P"/>
        <xs:enumeration value="A"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="IntegrityCheckAlgorithm">
    <xs:annotation>
        <xs:documentation>vocSet: D17385 (C-0-D17385-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="SHA-1"/>
        <xs:enumeration value="SHA-256"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="MediaType">
    <xs:annotation>
        <xs:documentation>vocSet: D14824 (C-0-D14824-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="ApplicationMediaType AudioMediaType ImageMediaType ModelMediaType
MultipartMediaType TextMediaType VideoMediaType"/>
</xs:simpleType>
<xs:simpleType name="ApplicationMediaType">
    <xs:annotation>
        <xs:documentation>abstDomain: V14832 (C-0-D14824-V14832-cpt)</xs:documentation>
    </xs:annotation>
```

```
        <xs:restriction base="cs">
            <xs:enumeration value="application/dicom"/>
            <xs:enumeration value="application/msword"/>
            <xs:enumeration value="application/pdf"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="AudioMediaType">
        <xs:annotation>
            <xs:documentation>abstDomain: V14835 (C-0-D14824-V14835-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="audio/basic"/>
            <xs:enumeration value="audio/k32adpcm"/>
            <xs:enumeration value="audio/mpeg"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ImageMediaType">
        <xs:annotation>
            <xs:documentation>abstDomain: V14839 (C-0-D14824-V14839-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="image/g3fax"/>
            <xs:enumeration value="image/gif"/>
            <xs:enumeration value="image/jpeg"/>
            <xs:enumeration value="image/png"/>
            <xs:enumeration value="image/tiff"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ModelMediaType">
        <xs:annotation>
            <xs:documentation>abstDomain: V14848 (C-0-D14824-V14848-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="model/vrml"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="MultipartMediaType">
        <xs:annotation>
            <xs:documentation>abstDomain: V14850 (C-0-D14824-V14850-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="multipart/x-hl7-cda-level1"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="TextMediaType">
        <xs:annotation>
            <xs:documentation>abstDomain: V14825 (C-0-D14824-V14825-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
```

```
        <xs:enumeration value="text/html"/>
        <xs:enumeration value="text/plain"/>
        <xs:enumeration value="text/rtf"/>
        <xs:enumeration value="text/sgml"/>
        <xs:enumeration value="text/x-hl7-ft"/>
        <xs:enumeration value="text/xml"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="VideoMediaType">
    <xs:annotation>
        <xs:documentation>abstDomain: V14845 (C-0-D14824-V14845-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="video/mpeg"/>
        <xs:enumeration value="video/x-avi"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="PostalAddressUse">
    <xs:annotation>
        <xs:documentation>vocSet: D10637 (C-0-D10637-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="AddressUse NameRepresentationUse">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="PHYS"/>
                <xs:enumeration value="PST"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="NameRepresentationUse">
    <xs:annotation>
        <xs:documentation>abstDomain: V17860 (C-0-D10637-V17860-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="ABC"/>
        <xs:enumeration value="IDE"/>
        <xs:enumeration value="SYL"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ProbabilityDistributionType">
    <xs:annotation>
        <xs:documentation>vocSet: D10747 (C-0-D10747-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="B"/>
        <xs:enumeration value="E"/>
        <xs:enumeration value="F"/>
        <xs:enumeration value="G"/>
```

```
            <xs:enumeration value="LN"/>
            <xs:enumeration value="N"/>
            <xs:enumeration value="T"/>
            <xs:enumeration value="U"/>
            <xs:enumeration value="X2"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="SetOperator">
        <xs:annotation>
            <xs:documentation>vocSet: D17416 (C-0-D17416-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="A"/>
            <xs:enumeration value="E"/>
            <xs:enumeration value="H"/>
            <xs:enumeration value="I"/>
            <xs:enumeration value="P"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="TelecommunicationAddressUse">
        <xs:annotation>
            <xs:documentation>vocSet: D201 (C-0-D201-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="AddressUse">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="AS"/>
                    <xs:enumeration value="EC"/>
                    <xs:enumeration value="MC"/>
                    <xs:enumeration value="PG"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="AddressUse">
        <xs:annotation>
            <xs:documentation>abstDomain: V190 (C-0-D201-V190-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="HomeAddressUse WorkPlaceAddressUse">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="BAD"/>
                    <xs:enumeration value="TMP"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="HomeAddressUse">
        <xs:annotation>
```

```
            <xs:documentation>specDomain: V10628 (C-0-D201-V190-V10628-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="H"/>
            <xs:enumeration value="HP"/>
            <xs:enumeration value="HV"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="WorkPlaceAddressUse">
        <xs:annotation>
            <xs:documentation>specDomain: V19613 (C-0-D201-V190-V19613-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="WP"/>
            <xs:enumeration value="DIR"/>
            <xs:enumeration value="PUB"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="TimingEvent">
        <xs:annotation>
            <xs:documentation>vocSet: D10706 (C-0-D10706-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="AC"/>
            <xs:enumeration value="ACD"/>
            <xs:enumeration value="ACM"/>
            <xs:enumeration value="ACV"/>
            <xs:enumeration value="HS"/>
            <xs:enumeration value="IC"/>
            <xs:enumeration value="ICD"/>
            <xs:enumeration value="ICM"/>
            <xs:enumeration value="ICV"/>
            <xs:enumeration value="PC"/>
            <xs:enumeration value="PCD"/>
            <xs:enumeration value="PCM"/>
            <xs:enumeration value="PCV"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="URLScheme">
        <xs:annotation>
            <xs:documentation>vocSet: D14866 (C-0-D14866-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="fax"/>
            <xs:enumeration value="file"/>
            <xs:enumeration value="ftp"/>
            <xs:enumeration value="http"/>
            <xs:enumeration value="mailto"/>
            <xs:enumeration value="mllp"/>
```

```
            <xs:enumeration value="modem"/>
            <xs:enumeration value="nfs"/>
            <xs:enumeration value="tel"/>
            <xs:enumeration value="telnet"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:annotation>
        <xs:documentation>The following types are used for structural RIM attributes</xs:documentation>
    </xs:annotation>
    <xs:simpleType name="ActClass">
        <xs:annotation>
            <xs:documentation>vocSet: D11527 (C-0-D11527-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ActClassRoot"/>
    </xs:simpleType>
    <xs:simpleType name="ActClassRoot">
        <xs:annotation>
            <xs:documentation>specDomain: V13856 (C-0-D11527-V13856-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ActClassContract ActClassControlAct ActClassObservation ActClassSupply
ActContainer x_ActClassDocumentEntryAct x_ActClassDocumentEntryOrganizer">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="ACT"/>
                    <xs:enumeration value="ACCM"/>
                    <xs:enumeration value="ACCT"/>
                    <xs:enumeration value="ACSN"/>
                    <xs:enumeration value="ADJUD"/>
                    <xs:enumeration value="CONS"/>
                    <xs:enumeration value="CONTREG"/>
                    <xs:enumeration value="CTTEVENT"/>
                    <xs:enumeration value="DISPACT"/>
                    <xs:enumeration value="ENC"/>
                    <xs:enumeration value="INC"/>
                    <xs:enumeration value="INFRM"/>
                    <xs:enumeration value="INVE"/>
                    <xs:enumeration value="LIST"/>
                    <xs:enumeration value="MPROT"/>
                    <xs:enumeration value="PCPR"/>
                    <xs:enumeration value="PROC"/>
                    <xs:enumeration value="REG"/>
                    <xs:enumeration value="REV"/>
                    <xs:enumeration value="SBADM"/>
                    <xs:enumeration value="SPCTRT"/>
                    <xs:enumeration value="SUBST"/>
                    <xs:enumeration value="TRNS"/>
                    <xs:enumeration value="VERIF"/>
                    <xs:enumeration value="XACT"/>
                </xs:restriction>
```

```
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="ActClassContract">
    <xs:annotation>
        <xs:documentation>specDomain: V14002 (C-0-D11527-V13856-V14002-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="ActClassFinancialContract">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="CNTRCT"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="ActClassFinancialContract">
    <xs:annotation>
        <xs:documentation>specDomain: V14003 (C-0-D11527-V13856-V14002-V14003-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="FCNTRCT"/>
        <xs:enumeration value="COV"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ActClassControlAct">
    <xs:annotation>
        <xs:documentation>specDomain: V11534 (C-0-D11527-V13856-V11534-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="CACT"/>
        <xs:enumeration value="ACTN"/>
        <xs:enumeration value="INFO"/>
        <xs:enumeration value="STC"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ActClassObservation">
    <xs:annotation>
        <xs:documentation>specDomain: V11529 (C-0-D11527-V13856-V11529-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="ActClassCondition ActClassObservationSeries ActClassROI">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="OBS"/>
                <xs:enumeration value="ALRT"/>
                <xs:enumeration value="CLNTRL"/>
                <xs:enumeration value="CNOD"/>
                <xs:enumeration value="DGIMG"/>
                <xs:enumeration value="INVSTG"/>
                <xs:enumeration value="SPCOBS"/>
```

```
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="ActClassCondition">
        <xs:annotation>
            <xs:documentation>specDomain: V19580 (C-0-D11527-V13856-V11529-V19580-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ActClassPublicHealthCase">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="COND"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="ActClassPublicHealthCase">
        <xs:annotation>
            <xs:documentation>specDomain: V11530 (C-0-D11527-V13856-V11529-V19580-V11530-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="CASE"/>
            <xs:enumeration value="OUTB"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActClassObservationSeries">
        <xs:annotation>
            <xs:documentation>specDomain: V18875 (C-0-D11527-V13856-V11529-V18875-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="OBSSER"/>
            <xs:enumeration value="OBSCOR"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActClassROI">
        <xs:annotation>
            <xs:documentation>abstDomain: V17893 (C-0-D11527-V13856-V11529-V17893-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="ROIBND"/>
            <xs:enumeration value="ROIOVL"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActClassSupply">
        <xs:annotation>
            <xs:documentation>specDomain: V11535 (C-0-D11527-V13856-V11535-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
```

```
            <xs:enumeration value="SPLY"/>
            <xs:enumeration value="DIET"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActContainer">
        <xs:annotation>
            <xs:documentation>abstDomain: V19445 (C-0-D11527-V13856-V19445-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ActClassComposition ActClassEntry ActClassExtract ActClassOrganizer">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="FOLDER"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="ActClassComposition">
        <xs:annotation>
            <xs:documentation>specDomain: V19442 (C-0-D11527-V13856-V19445-V19442-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ActClassDocument">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="COMPOSITION"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="ActClassDocument">
        <xs:annotation>
            <xs:documentation>specDomain: V18938 (C-0-D11527-V13856-V19445-V19442-V18938-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ActClinicalDocument">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="DOC"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="ActClinicalDocument">
        <xs:annotation>
            <xs:documentation>specDomain: V13948 (C-0-D11527-V13856-V19445-V19442-V18938-V13948-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="DOCCLIN"/>
            <xs:enumeration value="CDALVLONE"/>
```

```
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActClassEntry">
        <xs:annotation>
            <xs:documentation>specDomain: V19444 (C-0-D11527-V13856-V19445-V19444-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="ENTRY"/>
            <xs:enumeration value="BATTERY"/>
            <xs:enumeration value="CLUSTER"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActClassExtract">
        <xs:annotation>
            <xs:documentation>specDomain: V19441 (C-0-D11527-V13856-V19445-V19441-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="EXTRACT"/>
            <xs:enumeration value="EHR"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActClassOrganizer">
        <xs:annotation>
            <xs:documentation>specDomain: V19443 (C-0-D11527-V13856-V19445-V19443-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="ORGANIZER"/>
            <xs:enumeration value="CATEGORY"/>
            <xs:enumeration value="DOCBODY"/>
            <xs:enumeration value="DOCSECT"/>
            <xs:enumeration value="TOPIC"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_ActClassDocumentEntryAct">
        <xs:annotation>
            <xs:documentation>abstDomain: V19604 (C-0-D11527-V13856-V19604-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="ACT"/>
            <xs:enumeration value="ACCM"/>
            <xs:enumeration value="CONS"/>
            <xs:enumeration value="CTTEVENT"/>
            <xs:enumeration value="INC"/>
            <xs:enumeration value="INFRM"/>
            <xs:enumeration value="PCPR"/>
            <xs:enumeration value="REG"/>
            <xs:enumeration value="SPCTRT"/>
        </xs:restriction>
    </xs:simpleType>
```

```
<xs:simpleType name="x_ActClassDocumentEntryOrganizer">
    <xs:annotation>
        <xs:documentation>abstDomain: V19603 (C-0-D11527-V13856-V19603-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="BATTERY"/>
        <xs:enumeration value="CLUSTER"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ActMood">
    <xs:annotation>
        <xs:documentation>vocSet: D10196 (C-0-D10196-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="ActMoodCompletionTrack ActMoodPredicate x_ActMoodDefEvn x_
ActMoodDefEvnRqoPrmsPrp x_ActMoodDocumentObservation x_ActMoodEvnOrdPrmsPrp x_ActMoodIntentEvent x_
ActMoodOrdPrms x_ActMoodOrdPrmsEvn x_ActMoodRqoPrpAptArq x_DocumentActMood x_DocumentEncounterMood
x_DocumentProcedureMood x_DocumentSubstanceMood"/>
</xs:simpleType>
<xs:simpleType name="ActMoodCompletionTrack">
    <xs:annotation>
        <xs:documentation>abstDomain: V10197 (C-0-D10196-V10197-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="ActMoodIntent">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="DEF"/>
                <xs:enumeration value="EVN"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="ActMoodPredicate">
    <xs:annotation>
        <xs:documentation>abstDomain: V10202 (C-0-D10196-V10202-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="EVN.CRT"/>
        <xs:enumeration value="GOL"/>
        <xs:enumeration value="OPT"/>
        <xs:enumeration value="PERM"/>
        <xs:enumeration value="PERMRQ"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ActMoodDefEvn">
    <xs:annotation>
        <xs:documentation>abstDomain: V19375 (C-0-D10196-V19375-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="DEF"/>
        <xs:enumeration value="EVN"/>
```

```
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_ActMoodDefEvnRqoPrmsPrp">
        <xs:annotation>
            <xs:documentation>abstDomain: V19371 (C-0-D10196-V19371-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="DEF"/>
            <xs:enumeration value="EVN"/>
            <xs:enumeration value="PRMS"/>
            <xs:enumeration value="PRP"/>
            <xs:enumeration value="RQO"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_ActMoodDocumentObservation">
        <xs:annotation>
            <xs:documentation>abstDomain: V18943 (C-0-D10196-V18943-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="INT"/>
            <xs:enumeration value="DEF"/>
            <xs:enumeration value="EVN"/>
            <xs:enumeration value="GOL"/>
            <xs:enumeration value="PRMS"/>
            <xs:enumeration value="PRP"/>
            <xs:enumeration value="RQO"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_ActMoodEvnOrdPrmsPrp">
        <xs:annotation>
            <xs:documentation>abstDomain: V18965 (C-0-D10196-V18965-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="EVN"/>
            <xs:enumeration value="PRMS"/>
            <xs:enumeration value="PRP"/>
            <xs:enumeration value="RQO"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_ActMoodIntentEvent">
        <xs:annotation>
            <xs:documentation>abstDomain: V16742 (C-0-D10196-V16742-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ActMoodIntent">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="EVN"/>
                </xs:restriction>
            </xs:simpleType>
```

```
    </xs:union>
</xs:simpleType>
<xs:simpleType name="ActMoodIntent">
    <xs:annotation>
        <xs:documentation>specDomain: V10199 (C-0-D10196-V16742-V10199-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="INT"/>
        <xs:enumeration value="APT"/>
        <xs:enumeration value="ARQ"/>
        <xs:enumeration value="PRMS"/>
        <xs:enumeration value="PRP"/>
        <xs:enumeration value="RQO"/>
        <xs:enumeration value="SLOT"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ActMoodOrdPrms">
    <xs:annotation>
        <xs:documentation>abstDomain: V16735 (C-0-D10196-V16735-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="PRMS"/>
        <xs:enumeration value="RQO"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ActMoodOrdPrmsEvn">
    <xs:annotation>
        <xs:documentation>abstDomain: V16730 (C-0-D10196-V16730-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="EVN"/>
        <xs:enumeration value="PRMS"/>
        <xs:enumeration value="RQO"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ActMoodRqoPrpAptArq">
    <xs:annotation>
        <xs:documentation>abstDomain: V19372 (C-0-D10196-V19372-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="APT"/>
        <xs:enumeration value="ARQ"/>
        <xs:enumeration value="PRP"/>
        <xs:enumeration value="RQO"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_DocumentActMood">
    <xs:annotation>
        <xs:documentation>abstDomain: V19458 (C-0-D10196-V19458-cpt)</xs:documentation>
```

```
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="INT"/>
            <xs:enumeration value="APT"/>
            <xs:enumeration value="ARQ"/>
            <xs:enumeration value="DEF"/>
            <xs:enumeration value="EVN"/>
            <xs:enumeration value="PRMS"/>
            <xs:enumeration value="PRP"/>
            <xs:enumeration value="RQO"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_DocumentEncounterMood">
        <xs:annotation>
            <xs:documentation>abstDomain: V19459 (C-0-D10196-V19459-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="INT"/>
            <xs:enumeration value="APT"/>
            <xs:enumeration value="ARQ"/>
            <xs:enumeration value="EVN"/>
            <xs:enumeration value="PRMS"/>
            <xs:enumeration value="PRP"/>
            <xs:enumeration value="RQO"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_DocumentProcedureMood">
        <xs:annotation>
            <xs:documentation>abstDomain: V19460 (C-0-D10196-V19460-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="INT"/>
            <xs:enumeration value="APT"/>
            <xs:enumeration value="ARQ"/>
            <xs:enumeration value="DEF"/>
            <xs:enumeration value="EVN"/>
            <xs:enumeration value="PRMS"/>
            <xs:enumeration value="PRP"/>
            <xs:enumeration value="RQO"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_DocumentSubstanceMood">
        <xs:annotation>
            <xs:documentation>abstDomain: V19461 (C-0-D10196-V19461-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="INT"/>
            <xs:enumeration value="EVN"/>
            <xs:enumeration value="PRMS"/>
```

```
            <xs:enumeration value="PRP"/>
            <xs:enumeration value="RQO"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActRelationshipType">
        <xs:annotation>
            <xs:documentation>vocSet: D10317 (C-0-D10317-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ActRelationshipConditional ActRelationshipHasComponent ActRelationshipOutcome
ActRelationshipPertains ActRelationshipSequel x_ActRelationshipDocument x_ActRelationshipEntry x_Ac
tRelationshipEntryRelationship x_ActRelationshipExternalReference x_ActRelationshipPatientTransport
x_ActRelationshipPertinentInfo"/>
    </xs:simpleType>
    <xs:simpleType name="ActRelationshipConditional">
        <xs:annotation>
            <xs:documentation>abstDomain: V18977 (C-0-D10317-V18977-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ActRelationshipReason">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="CIND"/>
                    <xs:enumeration value="PRCN"/>
                    <xs:enumeration value="TRIG"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="ActRelationshipReason">
        <xs:annotation>
            <xs:documentation>specDomain: V19376 (C-0-D10317-V18977-V19376-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="RSON"/>
            <xs:enumeration value="MITGT"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActRelationshipHasComponent">
        <xs:annotation>
            <xs:documentation>specDomain: V10318 (C-0-D10317-V10318-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="COMP"/>
            <xs:enumeration value="ARR"/>
            <xs:enumeration value="CTRLV"/>
            <xs:enumeration value="DEP"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActRelationshipOutcome">
        <xs:annotation>
            <xs:documentation>specDomain: V10324 (C-0-D10317-V10324-cpt)</xs:documentation>
```

```
        </xs:annotation>
        <xs:union memberTypes="ActRelationshipObjective">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="OUTC"/>
                    <xs:enumeration value="GOAL"/>
                    <xs:enumeration value="RISK"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="ActRelationshipObjective">
        <xs:annotation>
            <xs:documentation>abstDomain: V19625 (C-0-D10317-V10324-V19625-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="OBJC"/>
            <xs:enumeration value="OBJF"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ActRelationshipPertains">
        <xs:annotation>
            <xs:documentation>specDomain: V10329 (C-0-D10317-V10329-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ActRelationshipAccounting TemporallyPertains hasSupport">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="PERT"/>
                    <xs:enumeration value="AUTH"/>
                    <xs:enumeration value="CAUS"/>
                    <xs:enumeration value="COVBY"/>
                    <xs:enumeration value="DRIV"/>
                    <xs:enumeration value="EXPL"/>
                    <xs:enumeration value="ITEMSLOC"/>
                    <xs:enumeration value="LIMIT"/>
                    <xs:enumeration value="MFST"/>
                    <xs:enumeration value="NAME"/>
                    <xs:enumeration value="PREV"/>
                    <xs:enumeration value="REFR"/>
                    <xs:enumeration value="REFV"/>
                    <xs:enumeration value="SUBJ"/>
                    <xs:enumeration value="SUMM"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="ActRelationshipAccounting">
        <xs:annotation>
            <xs:documentation>abstDomain: V14900 (C-0-D10317-V10329-V14900-cpt)</xs:documentation>
```

```
    </xs:annotation>
    <xs:union memberTypes="ActRelationshipCostTracking ActRelationshipPosting cs"/>
</xs:simpleType>
<xs:simpleType name="ActRelationshipCostTracking">
    <xs:annotation>
        <xs:documentation>abstDomain: V19618 (C-0-D10317-V10329-V14900-V19618-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="CHRG"/>
        <xs:enumeration value="COST"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ActRelationshipPosting">
    <xs:annotation>
        <xs:documentation>abstDomain: V19617 (C-0-D10317-V10329-V14900-V19617-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="CREDIT"/>
        <xs:enumeration value="DEBIT"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="TemporallyPertains">
    <xs:annotation>
        <xs:documentation>abstDomain: V19590 (C-0-D10317-V10329-V19590-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="SAS"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="hasSupport">
    <xs:annotation>
        <xs:documentation>specDomain: V10330 (C-0-D10317-V10329-V10330-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="SPRT"/>
        <xs:enumeration value="SPRTBND"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ActRelationshipSequel">
    <xs:annotation>
        <xs:documentation>specDomain: V10337 (C-0-D10317-V10337-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="ActRelationshipExcerpt ActRelationshipFulfills ActRelationshipReplacement">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="SEQL"/>
                <xs:enumeration value="APND"/>
                <xs:enumeration value="DOC"/>
                <xs:enumeration value="ELNK"/>
```

```
                <xs:enumeration value="GEN"/>
                <xs:enumeration value="GEVL"/>
                <xs:enumeration value="INST"/>
                <xs:enumeration value="MTCH"/>
                <xs:enumeration value="OPTN"/>
                <xs:enumeration value="REV"/>
                <xs:enumeration value="UPDT"/>
                <xs:enumeration value="XFRM"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="ActRelationshipExcerpt">
    <xs:annotation>
        <xs:documentation>specDomain: V18660 (C-0-D10317-V10337-V18660-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="XCRPT"/>
        <xs:enumeration value="VRXCRPT"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ActRelationshipFulfills">
    <xs:annotation>
        <xs:documentation>specDomain: V10342 (C-0-D10317-V10337-V10342-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="FLFS"/>
        <xs:enumeration value="OCCR"/>
        <xs:enumeration value="OREF"/>
        <xs:enumeration value="SCH"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ActRelationshipReplacement">
    <xs:annotation>
        <xs:documentation>specDomain: V10338 (C-0-D10317-V10337-V10338-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="RPLC"/>
        <xs:enumeration value="SUCC"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ActRelationshipDocument">
    <xs:annotation>
        <xs:documentation>abstDomain: V11610 (C-0-D10317-V11610-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="RPLC"/>
        <xs:enumeration value="APND"/>
        <xs:enumeration value="XFRM"/>
```

```
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ActRelationshipEntry">
    <xs:annotation>
        <xs:documentation>abstDomain: V19446 (C-0-D10317-V19446-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="COMP"/>
        <xs:enumeration value="DRIV"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ActRelationshipEntryRelationship">
    <xs:annotation>
        <xs:documentation>abstDomain: V19447 (C-0-D10317-V19447-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="XCRPT"/>
        <xs:enumeration value="COMP"/>
        <xs:enumeration value="RSON"/>
        <xs:enumeration value="SPRT"/>
        <xs:enumeration value="CAUS"/>
        <xs:enumeration value="GEVL"/>
        <xs:enumeration value="MFST"/>
        <xs:enumeration value="REFR"/>
        <xs:enumeration value="SAS"/>
        <xs:enumeration value="SUBJ"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ActRelationshipExternalReference">
    <xs:annotation>
        <xs:documentation>abstDomain: V19000 (C-0-D10317-V19000-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="XCRPT"/>
        <xs:enumeration value="RPLC"/>
        <xs:enumeration value="SPRT"/>
        <xs:enumeration value="ELNK"/>
        <xs:enumeration value="REFR"/>
        <xs:enumeration value="SUBJ"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ActRelationshipPatientTransport">
    <xs:annotation>
        <xs:documentation>abstDomain: V19005 (C-0-D10317-V19005-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="ARR"/>
        <xs:enumeration value="DEP"/>
    </xs:restriction>
```

```
    </xs:simpleType>
    <xs:simpleType name="x_ActRelationshipPertinentInfo">
        <xs:annotation>
            <xs:documentation>abstDomain: V19562 (C-0-D10317-V19562-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="SPRT"/>
            <xs:enumeration value="CAUS"/>
            <xs:enumeration value="MFST"/>
            <xs:enumeration value="REFR"/>
            <xs:enumeration value="SUBJ"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="CommunicationFunctionType">
        <xs:annotation>
            <xs:documentation>vocSet: D16031 (C-0-D16031-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="RCV"/>
            <xs:enumeration value="RSP"/>
            <xs:enumeration value="SND"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ContextControl">
        <xs:annotation>
            <xs:documentation>vocSet: D16478 (C-0-D16478-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ContextControlAdditive ContextControlNonPropagating ContextControlOverriding
ContextControlPropagating"/>
    </xs:simpleType>
    <xs:simpleType name="ContextControlAdditive">
        <xs:annotation>
            <xs:documentation>abstDomain: V18934 (C-0-D16478-V18934-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="AN"/>
            <xs:enumeration value="AP"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ContextControlNonPropagating">
        <xs:annotation>
            <xs:documentation>abstDomain: V18937 (C-0-D16478-V18937-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="AN"/>
            <xs:enumeration value="ON"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ContextControlOverriding">
```

```
        <xs:annotation>
            <xs:documentation>abstDomain: V18935 (C-0-D16478-V18935-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="ON"/>
            <xs:enumeration value="OP"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ContextControlPropagating">
        <xs:annotation>
            <xs:documentation>abstDomain: V18936 (C-0-D16478-V18936-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="AP"/>
            <xs:enumeration value="OP"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="EntityClass">
        <xs:annotation>
            <xs:documentation>vocSet: D10882 (C-0-D10882-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="EntityClassRoot x_EntityClassDocumentReceiving x_EntityClassPersonOrOrgReceivi
ng"/>
    </xs:simpleType>
    <xs:simpleType name="EntityClassRoot">
        <xs:annotation>
            <xs:documentation>specDomain: V13922 (C-0-D10882-V13922-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="EntityClassLivingSubject EntityClassMaterial EntityClassOrganization
EntityClassPlace">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="ENT"/>
                    <xs:enumeration value="HCE"/>
                    <xs:enumeration value="RGRP"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="EntityClassLivingSubject">
        <xs:annotation>
            <xs:documentation>specDomain: V10884 (C-0-D10882-V13922-V10884-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="EntityClassNonPersonLivingSubject">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="LIV"/>
                    <xs:enumeration value="PSN"/>
                </xs:restriction>
```

```
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="EntityClassNonPersonLivingSubject">
        <xs:annotation>
            <xs:documentation>specDomain: V11621 (C-0-D10882-V13922-V10884-V11621-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="NLIV"/>
            <xs:enumeration value="ANM"/>
            <xs:enumeration value="MIC"/>
            <xs:enumeration value="PLNT"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="EntityClassMaterial">
        <xs:annotation>
            <xs:documentation>specDomain: V10883 (C-0-D10882-V13922-V10883-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="EntityClassManufacturedMaterial">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="MAT"/>
                    <xs:enumeration value="CHEM"/>
                    <xs:enumeration value="FOOD"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="EntityClassManufacturedMaterial">
        <xs:annotation>
            <xs:documentation>specDomain: V13934 (C-0-D10882-V13922-V10883-V13934-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="EntityClassContainer EntityClassDevice">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="MMAT"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="EntityClassContainer">
        <xs:annotation>
            <xs:documentation>specDomain: V11622 (C-0-D10882-V13922-V10883-V13934-V11622-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="CONT"/>
            <xs:enumeration value="HOLD"/>
        </xs:restriction>
```

```
    </xs:simpleType>
    <xs:simpleType name="EntityClassDevice">
        <xs:annotation>
            <xs:documentation>specDomain: V11623 (C-0-D10882-V13922-V10883-V13934-V11623-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="DEV"/>
            <xs:enumeration value="CER"/>
            <xs:enumeration value="MODDV"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="EntityClassPlace">
        <xs:annotation>
            <xs:documentation>specDomain: V10892 (C-0-D10882-V13922-V10892-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="PLC"/>
            <xs:enumeration value="CITY"/>
            <xs:enumeration value="COUNTRY"/>
            <xs:enumeration value="COUNTY"/>
            <xs:enumeration value="PROVINCE"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_EntityClassDocumentReceiving">
        <xs:annotation>
            <xs:documentation>abstDomain: V19462 (C-0-D10882-V19462-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="EntityClassOrganization">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="HCE"/>
                    <xs:enumeration value="PSN"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="x_EntityClassPersonOrOrgReceiving">
        <xs:annotation>
            <xs:documentation>abstDomain: V19463 (C-0-D10882-V19463-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="EntityClassOrganization">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="PSN"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
```

```
<xs:simpleType name="EntityClassOrganization">
    <xs:annotation>
        <xs:documentation>specDomain: V10889 (C-0-D10882-V19463-V10889-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="State">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="ORG"/>
                <xs:enumeration value="PUB"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="State">
    <xs:annotation>
        <xs:documentation>specDomain: V19455 (C-0-D10882-V19463-V10889-V19455-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="STATE"/>
        <xs:enumeration value="NAT"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="EntityDeterminer">
    <xs:annotation>
        <xs:documentation>vocSet: D10878 (C-0-D10878-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="EntityDeterminerDetermined x_DeterminerInstanceKind">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="INSTANCE"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="EntityDeterminerDetermined">
    <xs:annotation>
        <xs:documentation>specDomain: V10879 (C-0-D10878-V10879-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="KIND"/>
        <xs:enumeration value="QUANTIFIED_KIND"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_DeterminerInstanceKind">
    <xs:annotation>
        <xs:documentation>abstDomain: V19647 (C-0-D10878-V19647-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="KIND"/>
```

```
        <xs:enumeration value="INSTANCE"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="NullFlavor">
    <xs:annotation>
        <xs:documentation>vocSet: D10609 (C-0-D10609-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="NoInformation">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="NP"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="NoInformation">
    <xs:annotation>
        <xs:documentation>specDomain: V10610 (C-0-D10609-V10610-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="Other Unknown">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="NI"/>
                <xs:enumeration value="MSK"/>
                <xs:enumeration value="NA"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="Other">
    <xs:annotation>
        <xs:documentation>specDomain: V10616 (C-0-D10609-V10610-V10616-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="OTH"/>
        <xs:enumeration value="NINF"/>
        <xs:enumeration value="PINF"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Unknown">
    <xs:annotation>
        <xs:documentation>specDomain: V10612 (C-0-D10609-V10610-V10612-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="AskedButUnknown">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="UNK"/>
                <xs:enumeration value="NASK"/>
                <xs:enumeration value="TRC"/>
```

```
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="AskedButUnknown">
    <xs:annotation>
        <xs:documentation>specDomain: V10614 (C-0-D10609-V10610-V10612-V10614-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="ASKU"/>
        <xs:enumeration value="NAV"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ParticipationType">
    <xs:annotation>
        <xs:documentation>vocSet: D10901 (C-0-D10901-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="ParticipationAncillary ParticipationIndirectTarget ParticipationInformationGenerator
ParticipationInformationRecipient ParticipationPhysicalPerformer ParticipationTargetDirect ParticipationTargetLocation
ParticipationVerifier x_EncounterParticipant x_EncounterPerformerParticipation x_InformationRecipient x_
ParticipationAuthorPerformer x_ParticipationEntVrf x_ParticipationPrfEntVrf x_ParticipationVrfRespSprfWit
x_ServiceEventPerformer">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="CST"/>
                <xs:enumeration value="RESP"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="ParticipationAncillary">
    <xs:annotation>
        <xs:documentation>abstDomain: V10247 (C-0-D10901-V10247-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="ADM"/>
        <xs:enumeration value="ATND"/>
        <xs:enumeration value="CALLBCK"/>
        <xs:enumeration value="CON"/>
        <xs:enumeration value="DIS"/>
        <xs:enumeration value="ESC"/>
        <xs:enumeration value="REF"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ParticipationIndirectTarget">
    <xs:annotation>
        <xs:documentation>specDomain: V19032 (C-0-D10901-V19032-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="IND"/>
```

```
            <xs:enumeration value="BEN"/>
            <xs:enumeration value="COV"/>
            <xs:enumeration value="HLD"/>
            <xs:enumeration value="RCT"/>
            <xs:enumeration value="RCV"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ParticipationInformationGenerator">
        <xs:annotation>
            <xs:documentation>abstDomain: V10251 (C-0-D10901-V10251-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="AUT"/>
            <xs:enumeration value="ENT"/>
            <xs:enumeration value="INF"/>
            <xs:enumeration value="WIT"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ParticipationInformationRecipient">
        <xs:annotation>
            <xs:documentation>specDomain: V10263 (C-0-D10901-V10263-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="IRCP"/>
            <xs:enumeration value="NOT"/>
            <xs:enumeration value="PRCP"/>
            <xs:enumeration value="REFB"/>
            <xs:enumeration value="REFT"/>
            <xs:enumeration value="TRC"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ParticipationPhysicalPerformer">
        <xs:annotation>
            <xs:documentation>specDomain: V10248 (C-0-D10901-V10248-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="PRF"/>
            <xs:enumeration value="DIST"/>
            <xs:enumeration value="PPRF"/>
            <xs:enumeration value="SPRF"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="ParticipationTargetDirect">
        <xs:annotation>
            <xs:documentation>specDomain: V10286 (C-0-D10901-V10286-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="ParticipationTargetDevice ParticipationTargetSubject">
            <xs:simpleType>
                <xs:restriction base="cs">
```

```
                <xs:enumeration value="DIR"/>
                <xs:enumeration value="BBY"/>
                <xs:enumeration value="CSM"/>
                <xs:enumeration value="DON"/>
                <xs:enumeration value="PRD"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="ParticipationTargetDevice">
    <xs:annotation>
        <xs:documentation>specDomain: V10298 (C-0-D10901-V10286-V10298-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="DEV"/>
        <xs:enumeration value="NRD"/>
        <xs:enumeration value="RDV"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ParticipationTargetSubject">
    <xs:annotation>
        <xs:documentation>specDomain: V19584 (C-0-D10901-V10286-V19584-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="SBJ"/>
        <xs:enumeration value="SPC"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ParticipationTargetLocation">
    <xs:annotation>
        <xs:documentation>specDomain: V10302 (C-0-D10901-V10302-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="LOC"/>
        <xs:enumeration value="DST"/>
        <xs:enumeration value="ELOC"/>
        <xs:enumeration value="ORG"/>
        <xs:enumeration value="RML"/>
        <xs:enumeration value="VIA"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="ParticipationVerifier">
    <xs:annotation>
        <xs:documentation>specDomain: V10259 (C-0-D10901-V10259-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="VRF"/>
        <xs:enumeration value="AUTHEN"/>
        <xs:enumeration value="LA"/>
```

```
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_EncounterParticipant">
    <xs:annotation>
        <xs:documentation>abstDomain: V19605 (C-0-D10901-V19605-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="ADM"/>
        <xs:enumeration value="ATND"/>
        <xs:enumeration value="CON"/>
        <xs:enumeration value="DIS"/>
        <xs:enumeration value="REF"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_EncounterPerformerParticipation">
    <xs:annotation>
        <xs:documentation>abstDomain: V16764 (C-0-D10901-V16764-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="PRF"/>
        <xs:enumeration value="CON"/>
        <xs:enumeration value="SPRF"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_InformationRecipient">
    <xs:annotation>
        <xs:documentation>abstDomain: V19366 (C-0-D10901-V19366-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="PRCP"/>
        <xs:enumeration value="TRC"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ParticipationAuthorPerformer">
    <xs:annotation>
        <xs:documentation>abstDomain: V19080 (C-0-D10901-V19080-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="PRF"/>
        <xs:enumeration value="AUT"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_ParticipationEntVrf">
    <xs:annotation>
        <xs:documentation>abstDomain: V19600 (C-0-D10901-V19600-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="VRF"/>
        <xs:enumeration value="ENT"/>
```

```
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_ParticipationPrfEntVrf">
        <xs:annotation>
            <xs:documentation>abstDomain: V19601 (C-0-D10901-V19601-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="PRF"/>
            <xs:enumeration value="VRF"/>
            <xs:enumeration value="ENT"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_ParticipationVrfRespSprfWit">
        <xs:annotation>
            <xs:documentation>abstDomain: V19083 (C-0-D10901-V19083-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="VRF"/>
            <xs:enumeration value="RESP"/>
            <xs:enumeration value="SPRF"/>
            <xs:enumeration value="WIT"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_ServiceEventPerformer">
        <xs:annotation>
            <xs:documentation>abstDomain: V19606 (C-0-D10901-V19606-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="PRF"/>
            <xs:enumeration value="PPRF"/>
            <xs:enumeration value="SPRF"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="RoleClass">
        <xs:annotation>
            <xs:documentation>vocSet: D11555 (C-0-D11555-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="RoleClassRoot"/>
    </xs:simpleType>
    <xs:simpleType name="RoleClassRoot">
        <xs:annotation>
            <xs:documentation>specDomain: V13940 (C-0-D11555-V13940-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="RoleClassAssociative RoleClassOntological RoleClassPartitive x_
DocumentEntrySubject x_DocumentSubject x_InformationRecipientRole x_RoleClassAccommodationRequestor x_
RoleClassCoverage x_RoleClassCoverageInvoice x_RoleClassCredentialedEntity x_RoleClassPayeePolicyRelationship
">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="ROL"/>
```

```
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="RoleClassAssociative">
    <xs:annotation>
        <xs:documentation>abstDomain: V19313 (C-0-D11555-V13940-V19313-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="RoleClassMutualRelationship RoleClassPassive cs"/>
</xs:simpleType>
<xs:simpleType name="RoleClassMutualRelationship">
    <xs:annotation>
        <xs:documentation>abstDomain: V19316 (C-0-D11555-V13940-V19313-V19316-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="RoleClassRelationshipFormal">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="CAREGIVER"/>
                <xs:enumeration value="PRS"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="RoleClassRelationshipFormal">
    <xs:annotation>
        <xs:documentation>abstDomain: V10416 (C-0-D11555-V13940-V19313-V19316-V10416-cpt)</
xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="LicensedEntityRole RoleClassAgent RoleClassEmployee
RoleClassInvestigationSubject">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="CIT"/>
                <xs:enumeration value="COVPTY"/>
                <xs:enumeration value="CRINV"/>
                <xs:enumeration value="CRSPNSR"/>
                <xs:enumeration value="GUAR"/>
                <xs:enumeration value="PAT"/>
                <xs:enumeration value="PAYEE"/>
                <xs:enumeration value="PAYOR"/>
                <xs:enumeration value="POLHOLD"/>
                <xs:enumeration value="QUAL"/>
                <xs:enumeration value="SPNSR"/>
                <xs:enumeration value="STD"/>
                <xs:enumeration value="UNDWRT"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
```

```
    <xs:simpleType name="RoleClassAgent">
        <xs:annotation>
            <xs:documentation>specDomain: V14006 (C-0-D11555-V13940-V19313-V19316-V10416-V14006-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="RoleClassAssignedEntity">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="AGNT"/>
                    <xs:enumeration value="GUARD"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="RoleClassAssignedEntity">
        <xs:annotation>
            <xs:documentation>specDomain: V11595 (C-0-D11555-V13940-V19313-V19316-V10416-V14006-V11595-
cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="RoleClassContact">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="ASSIGNED"/>
                    <xs:enumeration value="COMPAR"/>
                    <xs:enumeration value="SGNOFF"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="RoleClassContact">
        <xs:annotation>
            <xs:documentation>specDomain: V12205 (C-0-D11555-V13940-V19313-V19316-V10416-V14006-V11595-
V12205-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="CON"/>
            <xs:enumeration value="ECON"/>
            <xs:enumeration value="NOK"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="RoleClassEmployee">
        <xs:annotation>
            <xs:documentation>specDomain: V11569 (C-0-D11555-V13940-V19313-V19316-V10416-V11569-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="EMP"/>
            <xs:enumeration value="MIL"/>
        </xs:restriction>
    </xs:simpleType>
```

```
    <xs:simpleType name="RoleClassInvestigationSubject">
        <xs:annotation>
            <xs:documentation>specDomain: V19587 (C-0-D11555-V13940-V19313-V19316-V10416-V19587-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="INVSBJ"/>
            <xs:enumeration value="CASESBJ"/>
            <xs:enumeration value="RESBJ"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="RoleClassPassive">
        <xs:annotation>
            <xs:documentation>abstDomain: V19105 (C-0-D11555-V13940-V19313-V19105-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="RoleClassDistributedMaterial RoleClassManufacturedProduct RoleClassServiceDelive
ryLocation">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="ACCESS"/>
                    <xs:enumeration value="BIRTHPL"/>
                    <xs:enumeration value="EXPR"/>
                    <xs:enumeration value="HLD"/>
                    <xs:enumeration value="HLTHCHRT"/>
                    <xs:enumeration value="IDENT"/>
                    <xs:enumeration value="MNT"/>
                    <xs:enumeration value="OWN"/>
                    <xs:enumeration value="RGPR"/>
                    <xs:enumeration value="TERR"/>
                    <xs:enumeration value="WRTE"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="RoleClassDistributedMaterial">
        <xs:annotation>
            <xs:documentation>specDomain: V10418 (C-0-D11555-V13940-V19313-V19105-V10418-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="DST"/>
            <xs:enumeration value="RET"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="RoleClassManufacturedProduct">
        <xs:annotation>
            <xs:documentation>specDomain: V11580 (C-0-D11555-V13940-V19313-V19105-V11580-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
```

```
            <xs:enumeration value="MANU"/>
            <xs:enumeration value="THER"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="RoleClassServiceDeliveryLocation">
        <xs:annotation>
            <xs:documentation>specDomain: V16927 (C-0-D11555-V13940-V19313-V19105-V16927-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="SDLOC"/>
            <xs:enumeration value="DSDLOC"/>
            <xs:enumeration value="ISDLOC"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="RoleClassOntological">
        <xs:annotation>
            <xs:documentation>abstDomain: V10428 (C-0-D11555-V13940-V10428-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="RoleClassIsSpeciesEntity">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="INST"/>
                    <xs:enumeration value="SUBS"/>
                    <xs:enumeration value="SUBY"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="RoleClassIsSpeciesEntity">
        <xs:annotation>
            <xs:documentation>specDomain: V10441 (C-0-D11555-V13940-V10428-V10441-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="GEN"/>
            <xs:enumeration value="GRIC"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="RoleClassPartitive">
        <xs:annotation>
            <xs:documentation>abstDomain: V10429 (C-0-D11555-V13940-V10429-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="RoleClassIngredientEntity RoleClassLocatedEntity RoleClassSpecimen">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="CONT"/>
                    <xs:enumeration value="MBR"/>
                    <xs:enumeration value="PART"/>
                </xs:restriction>
```

```
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="RoleClassIngredientEntity">
        <xs:annotation>
            <xs:documentation>specDomain: V10430 (C-0-D11555-V13940-V10429-V10430-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="RoleClassInactiveIngredient">
            <xs:simpleType>
                <xs:restriction base="cs">
                    <xs:enumeration value="INGR"/>
                    <xs:enumeration value="ACTI"/>
                    <xs:enumeration value="ACTM"/>
                    <xs:enumeration value="ADTV"/>
                    <xs:enumeration value="BASE"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
    <xs:simpleType name="RoleClassInactiveIngredient">
        <xs:annotation>
            <xs:documentation>specDomain: V19089 (C-0-D11555-V13940-V10429-V10430-V19089-cpt)</
xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="IACT"/>
            <xs:enumeration value="COLR"/>
            <xs:enumeration value="FLVR"/>
            <xs:enumeration value="PRSV"/>
            <xs:enumeration value="STBL"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="RoleClassLocatedEntity">
        <xs:annotation>
            <xs:documentation>specDomain: V16815 (C-0-D11555-V13940-V10429-V16815-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="LOCE"/>
            <xs:enumeration value="STOR"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="RoleClassSpecimen">
        <xs:annotation>
            <xs:documentation>specDomain: V11591 (C-0-D11555-V13940-V10429-V11591-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="SPEC"/>
            <xs:enumeration value="ALQT"/>
            <xs:enumeration value="ISLT"/>
```

```
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_DocumentEntrySubject">
        <xs:annotation>
            <xs:documentation>abstDomain: V19367 (C-0-D11555-V13940-V19367-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="SPEC"/>
            <xs:enumeration value="PAT"/>
            <xs:enumeration value="PRS"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_DocumentSubject">
        <xs:annotation>
            <xs:documentation>abstDomain: V19368 (C-0-D11555-V13940-V19368-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="PAT"/>
            <xs:enumeration value="PRS"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_InformationRecipientRole">
        <xs:annotation>
            <xs:documentation>abstDomain: V16772 (C-0-D11555-V13940-V16772-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="ASSIGNED"/>
            <xs:enumeration value="HLTHCHRT"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_RoleClassAccommodationRequestor">
        <xs:annotation>
            <xs:documentation>abstDomain: V19382 (C-0-D11555-V13940-V19382-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="AGNT"/>
            <xs:enumeration value="PAT"/>
            <xs:enumeration value="PROV"/>
            <xs:enumeration value="PRS"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="x_RoleClassCoverage">
        <xs:annotation>
            <xs:documentation>abstDomain: V14008 (C-0-D11555-V13940-V14008-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="COVPTY"/>
            <xs:enumeration value="POLHOLD"/>
            <xs:enumeration value="SPNSR"/>
```

```
        <xs:enumeration value="UNDWRT"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_RoleClassCoverageInvoice">
    <xs:annotation>
        <xs:documentation>abstDomain: V14013 (C-0-D11555-V13940-V14013-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="PAYEE"/>
        <xs:enumeration value="PAYOR"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_RoleClassCredentialedEntity">
    <xs:annotation>
        <xs:documentation>abstDomain: V16930 (C-0-D11555-V13940-V16930-cpt)</xs:documentation>
    </xs:annotation>
    <xs:union memberTypes="LicensedEntityRole">
        <xs:simpleType>
            <xs:restriction base="cs">
                <xs:enumeration value="ASSIGNED"/>
                <xs:enumeration value="QUAL"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:union>
</xs:simpleType>
<xs:simpleType name="LicensedEntityRole">
    <xs:annotation>
        <xs:documentation>specDomain: V16773 (C-0-D11555-V13940-V16930-V16773-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="LIC"/>
        <xs:enumeration value="NOT"/>
        <xs:enumeration value="PROV"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="x_RoleClassPayeePolicyRelationship">
    <xs:annotation>
        <xs:documentation>abstDomain: V19395 (C-0-D11555-V13940-V19395-cpt)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="cs">
        <xs:enumeration value="COVPTY"/>
        <xs:enumeration value="GUAR"/>
        <xs:enumeration value="POLHOLD"/>
        <xs:enumeration value="PROV"/>
        <xs:enumeration value="PRS"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="RoleLinkType">
    <xs:annotation>
```

```
            <xs:documentation>vocSet: D11603 (C-0-D11603-cpt)</xs:documentation>
        </xs:annotation>
        <xs:union memberTypes="RelatedLinkType"/>
    </xs:simpleType>
    <xs:simpleType name="RelatedLinkType">
        <xs:annotation>
            <xs:documentation>specDomain: V19615 (C-0-D11603-V19615-cpt)</xs:documentation>
        </xs:annotation>
        <xs:restriction base="cs">
            <xs:enumeration value="REL"/>
            <xs:enumeration value="BACKUP"/>
            <xs:enumeration value="DIRAUTH"/>
            <xs:enumeration value="INDAUTH"/>
            <xs:enumeration value="PART"/>
            <xs:enumeration value="REPL"/>
        </xs:restriction>
    </xs:simpleType>
</xs:schema>
```

F.4 XML-схема повествовательного блока

```
<?xml version="1.0" encoding="ASCII"?>
<!-- $Id: NarrativeBlock.xsd,v 1.4 2005/04/17 03:13:19 lmckenzi Exp $ -->
<xs:schema xmlns="urn:hl7-org:v3" xmlns:xs="http://www.w3.org/2001/XMLSchema" targetNamespace="urn:hl7-org:v3"
elementFormDefault="qualified">
    <xs:complexType name="StrucDoc.Text" mixed="true">
        <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:element name="content" type="StrucDoc.Content"/>
            <xs:element name="linkHtml" type="StrucDoc.LinkHtml"/>
            <xs:element name="sub" type="StrucDoc.Sub"/>
            <xs:element name="sup" type="StrucDoc.Sup"/>
            <xs:element name="br" type="StrucDoc.Br"/>
            <xs:element name="footnote" type="StrucDoc.Footnote"/>
            <xs:element name="footnoteRef" type="StrucDoc.FootnoteRef"/>
            <xs:element name="renderMultiMedia" type="StrucDoc.RenderMultiMedia"/>
            <xs:element name="paragraph" type="StrucDoc.Paragraph"/>
            <xs:element name="list" type="StrucDoc.List"/>
            <xs:element name="table" type="StrucDoc.Table"/>
        </xs:choice>
        <xs:attribute name="ID" type="xs:ID"/>
        <xs:attribute name="language" type="xs:NMTOKEN"/>
        <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
        <xs:attribute name="mediaType" type="xs:string" fixed="text/x-hl7-text+xml"/>
    </xs:complexType>
    <xs:complexType name="StrucDoc.Title" mixed="true">
        <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:element name="content" type="StrucDoc.TitleContent"/>
            <xs:element name="sub" type="StrucDoc.Sub"/>
            <xs:element name="sup" type="StrucDoc.Sup"/>
            <xs:element name="br" type="StrucDoc.Br"/>
            <xs:element name="footnote" type="StrucDoc.TitleFootnote"/>
```

```
        <xs:element name="footnoteRef" type="StrucDoc.FootnoteRef"/>
    </xs:choice>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
    <xs:attribute name="mediaType" type="xs:string" fixed="text/x-hl7-title+xml"/>
</xs:complexType>
<!-- DELETE THIS, we don't need to define a global element for text <xs:element name="text" type="text"/> -->
<xs:complexType name="StrucDoc.Br"/>
<xs:complexType name="StrucDoc.Caption" mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="linkHtml" type="StrucDoc.LinkHtml"/>
        <xs:element name="sub" type="StrucDoc.Sub"/>
        <xs:element name="sup" type="StrucDoc.Sup"/>
        <xs:element name="footnote" type="StrucDoc.Footnote"/>
        <xs:element name="footnoteRef" type="StrucDoc.FootnoteRef"/>
    </xs:choice>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
</xs:complexType>
<xs:complexType name="StrucDoc.Col">
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
    <xs:attribute name="span" type="xs:string" default="1"/>
    <xs:attribute name="width" type="xs:string"/>
    <xs:attribute name="align">
        <xs:simpleType>
            <xs:restriction base="xs:NMTOKEN">
                <xs:enumeration value="left"/>
                <xs:enumeration value="center"/>
                <xs:enumeration value="right"/>
                <xs:enumeration value="justify"/>
                <xs:enumeration value="char"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="char" type="xs:string"/>
    <xs:attribute name="charoff" type="xs:string"/>
    <xs:attribute name="valign">
        <xs:simpleType>
            <xs:restriction base="xs:NMTOKEN">
                <xs:enumeration value="top"/>
                <xs:enumeration value="middle"/>
                <xs:enumeration value="bottom"/>
                <xs:enumeration value="baseline"/>
            </xs:restriction>
        </xs:simpleType>
```

```
        </xs:attribute>
    </xs:complexType>
    <xs:complexType name="StrucDoc.Colgroup">
        <xs:sequence minOccurs="0" maxOccurs="unbounded">
            <xs:element name="col" type="StrucDoc.Col"/>
        </xs:sequence>
        <xs:attribute name="ID" type="xs:ID"/>
        <xs:attribute name="language" type="xs:NMTOKEN"/>
        <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
        <xs:attribute name="span" type="xs:string" default="1"/>
        <xs:attribute name="width" type="xs:string"/>
        <xs:attribute name="align">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="left"/>
                    <xs:enumeration value="center"/>
                    <xs:enumeration value="right"/>
                    <xs:enumeration value="justify"/>
                    <xs:enumeration value="char"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="char" type="xs:string"/>
        <xs:attribute name="charoff" type="xs:string"/>
        <xs:attribute name="valign">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="top"/>
                    <xs:enumeration value="middle"/>
                    <xs:enumeration value="bottom"/>
                    <xs:enumeration value="baseline"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
    </xs:complexType>
    <xs:complexType name="StrucDoc.Content" mixed="true">
        <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:element name="content" type="StrucDoc.Content"/>
            <xs:element name="linkHtml" type="StrucDoc.LinkHtml"/>
            <xs:element name="sub" type="StrucDoc.Sub"/>
            <xs:element name="sup" type="StrucDoc.Sup"/>
            <xs:element name="br" type="StrucDoc.Br"/>
            <xs:element name="footnote" type="StrucDoc.Footnote"/>
            <xs:element name="footnoteRef" type="StrucDoc.FootnoteRef"/>
            <xs:element name="renderMultiMedia" type="StrucDoc.RenderMultiMedia"/>
        </xs:choice>
        <xs:attribute name="ID" type="xs:ID"/>
        <xs:attribute name="language" type="xs:NMTOKEN"/>
        <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
```

```
    <xs:attribute name="revised">
        <xs:simpleType>
            <xs:restriction base="xs:NMTOKEN">
                <xs:enumeration value="insert"/>
                <xs:enumeration value="delete"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
</xs:complexType>
<xs:complexType name="StrucDoc.TitleContent" mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="content" type="StrucDoc.TitleContent"/>
        <xs:element name="sub" type="StrucDoc.Sub"/>
        <xs:element name="sup" type="StrucDoc.Sup"/>
        <xs:element name="br" type="StrucDoc.Br"/>
        <xs:element name="footnote" type="StrucDoc.TitleFootnote"/>
        <xs:element name="footnoteRef" type="StrucDoc.FootnoteRef"/>
    </xs:choice>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
</xs:complexType>
<xs:complexType name="StrucDoc.Footnote" mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="content" type="StrucDoc.Content"/>
        <xs:element name="linkHtml" type="StrucDoc.LinkHtml"/>
        <xs:element name="sub" type="StrucDoc.Sub"/>
        <xs:element name="sup" type="StrucDoc.Sup"/>
        <xs:element name="br" type="StrucDoc.Br"/>
        <xs:element name="renderMultiMedia" type="StrucDoc.RenderMultiMedia"/>
        <xs:element name="paragraph" type="StrucDoc.Paragraph"/>
        <xs:element name="list" type="StrucDoc.List"/>
        <xs:element name="table" type="StrucDoc.Table"/>
    </xs:choice>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
</xs:complexType>
<xs:complexType name="StrucDoc.TitleFootnote" mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="content" type="StrucDoc.TitleContent"/>
        <xs:element name="sub" type="StrucDoc.Sub"/>
        <xs:element name="sup" type="StrucDoc.Sup"/>
        <xs:element name="br" type="StrucDoc.Br"/>
    </xs:choice>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
</xs:complexType>
```

```
<xs:complexType name="StrucDoc.FootnoteRef">
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
    <xs:attribute name="IDREF" type="xs:IDREF" use="required"/>
</xs:complexType>
<xs:complexType name="StrucDoc.Item" mixed="true">
    <xs:sequence>
        <xs:element name="caption" type="StrucDoc.Caption" minOccurs="0"/>
        <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:element name="content" type="StrucDoc.Content"/>
            <xs:element name="linkHtml" type="StrucDoc.LinkHtml"/>
            <xs:element name="sub" type="StrucDoc.Sub"/>
            <xs:element name="sup" type="StrucDoc.Sup"/>
            <xs:element name="br" type="StrucDoc.Br"/>
            <xs:element name="footnote" type="StrucDoc.Footnote"/>
            <xs:element name="footnoteRef" type="StrucDoc.FootnoteRef"/>
            <xs:element name="renderMultiMedia" type="StrucDoc.RenderMultiMedia"/>
            <xs:element name="paragraph" type="StrucDoc.Paragraph"/>
            <xs:element name="list" type="StrucDoc.List"/>
            <xs:element name="table" type="StrucDoc.Table"/>
        </xs:choice>
    </xs:sequence>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
</xs:complexType>
<xs:complexType name="StrucDoc.LinkHtml" mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="footnote" type="StrucDoc.Footnote"/>
        <xs:element name="footnoteRef" type="StrucDoc.FootnoteRef"/>
    </xs:choice>
    <xs:attribute name="name" type="xs:string"/>
    <xs:attribute name="href" type="xs:string"/>
    <xs:attribute name="rel" type="xs:string"/>
    <xs:attribute name="rev" type="xs:string"/>
    <xs:attribute name="title" type="xs:string"/>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
</xs:complexType>
<xs:complexType name="StrucDoc.List">
    <xs:sequence>
        <xs:element name="caption" type="StrucDoc.Caption" minOccurs="0"/>
        <xs:element name="item" type="StrucDoc.Item" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
```

```
    <xs:attribute name="listType" default="unordered">
        <xs:simpleType>
            <xs:restriction base="xs:NMTOKEN">
                <xs:enumeration value="ordered"/>
                <xs:enumeration value="unordered"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
</xs:complexType>
<xs:complexType name="StrucDoc.Paragraph" mixed="true">
    <xs:sequence>
        <xs:element name="caption" type="StrucDoc.Caption" minOccurs="0"/>
        <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:element name="content" type="StrucDoc.Content"/>
            <xs:element name="linkHtml" type="StrucDoc.LinkHtml"/>
            <xs:element name="sub" type="StrucDoc.Sub"/>
            <xs:element name="sup" type="StrucDoc.Sup"/>
            <xs:element name="br" type="StrucDoc.Br"/>
            <xs:element name="footnote" type="StrucDoc.Footnote"/>
            <xs:element name="footnoteRef" type="StrucDoc.FootnoteRef"/>
            <xs:element name="renderMultiMedia" type="StrucDoc.RenderMultiMedia"/>
        </xs:choice>
    </xs:sequence>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
</xs:complexType>
<xs:complexType name="StrucDoc.RenderMultiMedia">
    <xs:sequence>
        <xs:element name="caption" type="StrucDoc.Caption" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="referencedObject" type="xs:IDREFS" use="required"/>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
</xs:complexType>
<xs:complexType name="StrucDoc.Sub" mixed="true"/>
<xs:complexType name="StrucDoc.Sup" mixed="true"/>
<xs:complexType name="StrucDoc.Table">
    <xs:sequence>
        <xs:element name="caption" type="StrucDoc.Caption" minOccurs="0"/>
        <xs:choice>
            <xs:element name="col" type="StrucDoc.Col" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element name="colgroup" type="StrucDoc.Colgroup" minOccurs="0" maxOccurs="unbounded"/>
        </xs:choice>
        <xs:element name="thead" type="StrucDoc.Thead" minOccurs="0"/>
        <xs:element name="tfoot" type="StrucDoc.Tfoot" minOccurs="0"/>
        <xs:element name="tbody" type="StrucDoc.Tbody" maxOccurs="unbounded"/>
    </xs:sequence>
```

```
        <xs:attribute name="ID" type="xs:ID"/>
        <xs:attribute name="language" type="xs:NMTOKEN"/>
        <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
        <xs:attribute name="summary" type="xs:string"/>
        <xs:attribute name="width" type="xs:string"/>
        <xs:attribute name="border" type="xs:string"/>
        <xs:attribute name="frame">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="void"/>
                    <xs:enumeration value="above"/>
                    <xs:enumeration value="below"/>
                    <xs:enumeration value="hsides"/>
                    <xs:enumeration value="lhs"/>
                    <xs:enumeration value="rhs"/>
                    <xs:enumeration value="vsides"/>
                    <xs:enumeration value="box"/>
                    <xs:enumeration value="border"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="rules">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="none"/>
                    <xs:enumeration value="groups"/>
                    <xs:enumeration value="rows"/>
                    <xs:enumeration value="cols"/>
                    <xs:enumeration value="all"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="cellspacing" type="xs:string"/>
        <xs:attribute name="cellpadding" type="xs:string"/>
    </xs:complexType>
    <xs:complexType name="StrucDoc.Tbody">
        <xs:sequence maxOccurs="unbounded">
            <xs:element name="tr" type="StrucDoc.Tr"/>
        </xs:sequence>
        <xs:attribute name="ID" type="xs:ID"/>
        <xs:attribute name="language" type="xs:NMTOKEN"/>
        <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
        <xs:attribute name="align">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="left"/>
                    <xs:enumeration value="center"/>
                    <xs:enumeration value="right"/>
                    <xs:enumeration value="justify"/>
```

```
                <xs:enumeration value="char"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="char" type="xs:string"/>
    <xs:attribute name="charoff" type="xs:string"/>
    <xs:attribute name="valign">
        <xs:simpleType>
            <xs:restriction base="xs:NMTOKEN">
                <xs:enumeration value="top"/>
                <xs:enumeration value="middle"/>
                <xs:enumeration value="bottom"/>
                <xs:enumeration value="baseline"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
</xs:complexType>
<xs:complexType name="StrucDoc.Td" mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="content" type="StrucDoc.Content"/>
        <xs:element name="linkHtml" type="StrucDoc.LinkHtml"/>
        <xs:element name="sub" type="StrucDoc.Sub"/>
        <xs:element name="sup" type="StrucDoc.Sup"/>
        <xs:element name="br" type="StrucDoc.Br"/>
        <xs:element name="footnote" type="StrucDoc.Footnote"/>
        <xs:element name="footnoteRef" type="StrucDoc.FootnoteRef"/>
        <xs:element name="renderMultiMedia" type="StrucDoc.RenderMultiMedia"/>
        <xs:element name="paragraph" type="StrucDoc.Paragraph"/>
        <xs:element name="list" type="StrucDoc.List"/>
    </xs:choice>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
    <xs:attribute name="abbr" type="xs:string"/>
    <xs:attribute name="axis" type="xs:string"/>
    <xs:attribute name="headers" type="xs:IDREFS"/>
    <xs:attribute name="scope">
        <xs:simpleType>
            <xs:restriction base="xs:NMTOKEN">
                <xs:enumeration value="row"/>
                <xs:enumeration value="col"/>
                <xs:enumeration value="rowgroup"/>
                <xs:enumeration value="colgroup"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="rowspan" type="xs:string" default="1"/>
    <xs:attribute name="colspan" type="xs:string" default="1"/>
    <xs:attribute name="align">
```

```
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="left"/>
                    <xs:enumeration value="center"/>
                    <xs:enumeration value="right"/>
                    <xs:enumeration value="justify"/>
                    <xs:enumeration value="char"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="char" type="xs:string"/>
        <xs:attribute name="charoff" type="xs:string"/>
        <xs:attribute name="valign">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="top"/>
                    <xs:enumeration value="middle"/>
                    <xs:enumeration value="bottom"/>
                    <xs:enumeration value="baseline"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
    </xs:complexType>
    <xs:complexType name="StrucDoc.Tfoot">
        <xs:sequence maxOccurs="unbounded">
            <xs:element name="tr" type="StrucDoc.Tr"/>
        </xs:sequence>
        <xs:attribute name="ID" type="xs:ID"/>
        <xs:attribute name="language" type="xs:NMTOKEN"/>
        <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
        <xs:attribute name="align">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="left"/>
                    <xs:enumeration value="center"/>
                    <xs:enumeration value="right"/>
                    <xs:enumeration value="justify"/>
                    <xs:enumeration value="char"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="char" type="xs:string"/>
        <xs:attribute name="charoff" type="xs:string"/>
        <xs:attribute name="valign">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="top"/>
                    <xs:enumeration value="middle"/>
                    <xs:enumeration value="bottom"/>
```

```
                <xs:enumeration value="baseline"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
</xs:complexType>
<xs:complexType name="StrucDoc.Th" mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="content" type="StrucDoc.Content"/>
        <xs:element name="linkHtml" type="StrucDoc.LinkHtml"/>
        <xs:element name="sub" type="StrucDoc.Sub"/>
        <xs:element name="sup" type="StrucDoc.Sup"/>
        <xs:element name="br" type="StrucDoc.Br"/>
        <xs:element name="footnote" type="StrucDoc.Footnote"/>
        <xs:element name="footnoteRef" type="StrucDoc.FootnoteRef"/>
        <xs:element name="renderMultiMedia" type="StrucDoc.RenderMultiMedia"/>
    </xs:choice>
    <xs:attribute name="ID" type="xs:ID"/>
    <xs:attribute name="language" type="xs:NMTOKEN"/>
    <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
    <xs:attribute name="abbr" type="xs:string"/>
    <xs:attribute name="axis" type="xs:string"/>
    <xs:attribute name="headers" type="xs:IDREFS"/>
    <xs:attribute name="scope">
        <xs:simpleType>
            <xs:restriction base="xs:NMTOKEN">
                <xs:enumeration value="row"/>
                <xs:enumeration value="col"/>
                <xs:enumeration value="rowgroup"/>
                <xs:enumeration value="colgroup"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="rowspan" type="xs:string" default="1"/>
    <xs:attribute name="colspan" type="xs:string" default="1"/>
    <xs:attribute name="align">
        <xs:simpleType>
            <xs:restriction base="xs:NMTOKEN">
                <xs:enumeration value="left"/>
                <xs:enumeration value="center"/>
                <xs:enumeration value="right"/>
                <xs:enumeration value="justify"/>
                <xs:enumeration value="char"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="char" type="xs:string"/>
    <xs:attribute name="charoff" type="xs:string"/>
    <xs:attribute name="valign">
        <xs:simpleType>
```

```
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="top"/>
                    <xs:enumeration value="middle"/>
                    <xs:enumeration value="bottom"/>
                    <xs:enumeration value="baseline"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
    </xs:complexType>
    <xs:complexType name="StrucDoc.Thead">
        <xs:sequence maxOccurs="unbounded">
            <xs:element name="tr" type="StrucDoc.Tr"/>
        </xs:sequence>
        <xs:attribute name="ID" type="xs:ID"/>
        <xs:attribute name="language" type="xs:NMTOKEN"/>
        <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
        <xs:attribute name="align">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="left"/>
                    <xs:enumeration value="center"/>
                    <xs:enumeration value="right"/>
                    <xs:enumeration value="justify"/>
                    <xs:enumeration value="char"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="char" type="xs:string"/>
        <xs:attribute name="charoff" type="xs:string"/>
        <xs:attribute name="valign">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="top"/>
                    <xs:enumeration value="middle"/>
                    <xs:enumeration value="bottom"/>
                    <xs:enumeration value="baseline"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
    </xs:complexType>
    <xs:complexType name="StrucDoc.Tr">
        <xs:choice maxOccurs="unbounded">
            <xs:element name="th" type="StrucDoc.Th"/>
            <xs:element name="td" type="StrucDoc.Td"/>
        </xs:choice>
        <xs:attribute name="ID" type="xs:ID"/>
        <xs:attribute name="language" type="xs:NMTOKEN"/>
        <xs:attribute name="styleCode" type="xs:NMTOKENS"/>
        <xs:attribute name="align">
```

```
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="left"/>
                    <xs:enumeration value="center"/>
                    <xs:enumeration value="right"/>
                    <xs:enumeration value="justify"/>
                    <xs:enumeration value="char"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="char" type="xs:string"/>
        <xs:attribute name="charoff" type="xs:string"/>
        <xs:attribute name="valign">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKEN">
                    <xs:enumeration value="top"/>
                    <xs:enumeration value="middle"/>
                    <xs:enumeration value="bottom"/>
                    <xs:enumeration value="baseline"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
    </xs:complexType>
</xs:schema>
```

Приложение G
(справочное)

Руководство по стандартам HL7, версия 3

G.1 Предисловие

Рабочая группа HL7 является организацией, ориентированной на максимальное удовлетворение интересов ее членов. Поскольку многие члены Рабочей группы HL7, будь то производители или поставщики, реализующие спецификации, заинтересованы в ускоренном продвижении своих продуктов на рынке, очень важна как можно более гибкая политика.

Это означает, что Рабочая группа HL7 может поставить перед собой задачу составления спецификаций в зарождающейся сфере передачи сообщений, для которой существуют единичные фактические реализации. Спецификации, выпускаемые Рабочей группой HL7, предусматривают использование еще не существующих бизнес-процессов. Во многих случаях для принятия части требований необходим консенсус членов Рабочей группы HL7, но это пока распространяется не на все требования.

В некоторых случаях, жесткие сроки и загруженность работой различных Технических комитетов Рабочей группы HL7 приводят к тому, что лишь некоторые комитеты могут предложить спецификации, которые будут сочтены важными другими комитетами. Встречаются случаи, когда по итогам общего голосования частная спецификация может быть опубликована в голосуемом пакете в целях получения обратной связи от широкой аудитории, позволяющей определить практическое значение предложенного решения.

Управление деятельностью всех членов организации является сложной задачей, поскольку многие Технические комитеты работают параллельно. Когда выпуск спецификации на подходе, те спецификации, что требуют обсуждения в более широкой аудитории, объединяются и публикуются. Это делается для предоставления спецификаций не только для реализации, но и в некоторых случаях для заложения основ функциональности, которая будет внесена в последующие версии. Получение отзывов более широкой аудитории и поддержка тех, кто использует еще не утвержденные спецификации, повысят качество всех спецификаций, выпускаемых HL7.

Рабочая группа HL7 начинает использовать лучшие приемы успешных групп разработчиков, независимо от того, является ли продукция программой или спецификацией. Мы производим продукцию в управляемых выпусках, используя в процессе разработки модели управления сложностью и обеспечения согласованности; используем в процессе разработки автоматизацию для повышения производительности и качества; поощряем коммуникацию между организациями; начиная с ранних стадий разработки, получаем отзывы наших клиентов и реагируем на все их потребности.

G.1.1 О данном руководстве

Данное руководство предназначено для членов Рабочей группы Health Level Seven (HL7) в качестве дополнения к стандарту версии 3. Его предшественником был документ Message Development Framework (основы разработки сообщений), содержавший дополнительные материалы по дизайну и архитектуре стандарта версии 3. Однако этот документ не входил в текущие пакеты документов версии 3, представлявшихся на голосование, поэтому его использование не рекомендуется. В будущем он будет заменен новым документом, HL7 Development Framework (HDF, основы разработки стандартов HL7). Но он содержит изменения в нотации UML и в подходе, отличающемся от того, что принят в настоящем голосуемом пакете, к тому же пока находится в стадии проекта. Объяснение значений терминов, встречающихся в документе, можно найти в Глоссарии.

Руководство по версии 3 состоит из 2 разделов: «Основополагающие компоненты» и «Компоненты сообщений». Раздел «Основополагающие компоненты» содержит главы об информационных моделях, словаре, типах данных, и спецификации технологии реализации (ITS), используемых в спецификациях HL7 версии 3. Раздел «Компоненты сообщений» предоставляет собой руководство по артефактам стандарту Версии 3, например, сценарии, инициирующие события, взаимодействия и т. д., помогающее читателям лучше понимать, как читать и использовать описания этих артефактов. Для понимания спецификаций версии 3 читателям данного Руководства необходимо иметь общее представление об обоих компонентах.

G.2 Основополагающие компоненты стандартов HL7

G.2.1 Введение

Выпуск стандарта версии 3 содержит несколько документов. Некоторые из этих документов содержат спецификации версии 3, в то время как другие ключевую информацию по разработке и транспорту сообщений. Основополагающие компоненты относятся к последней категории. Их состав показан на рисунке G.1.

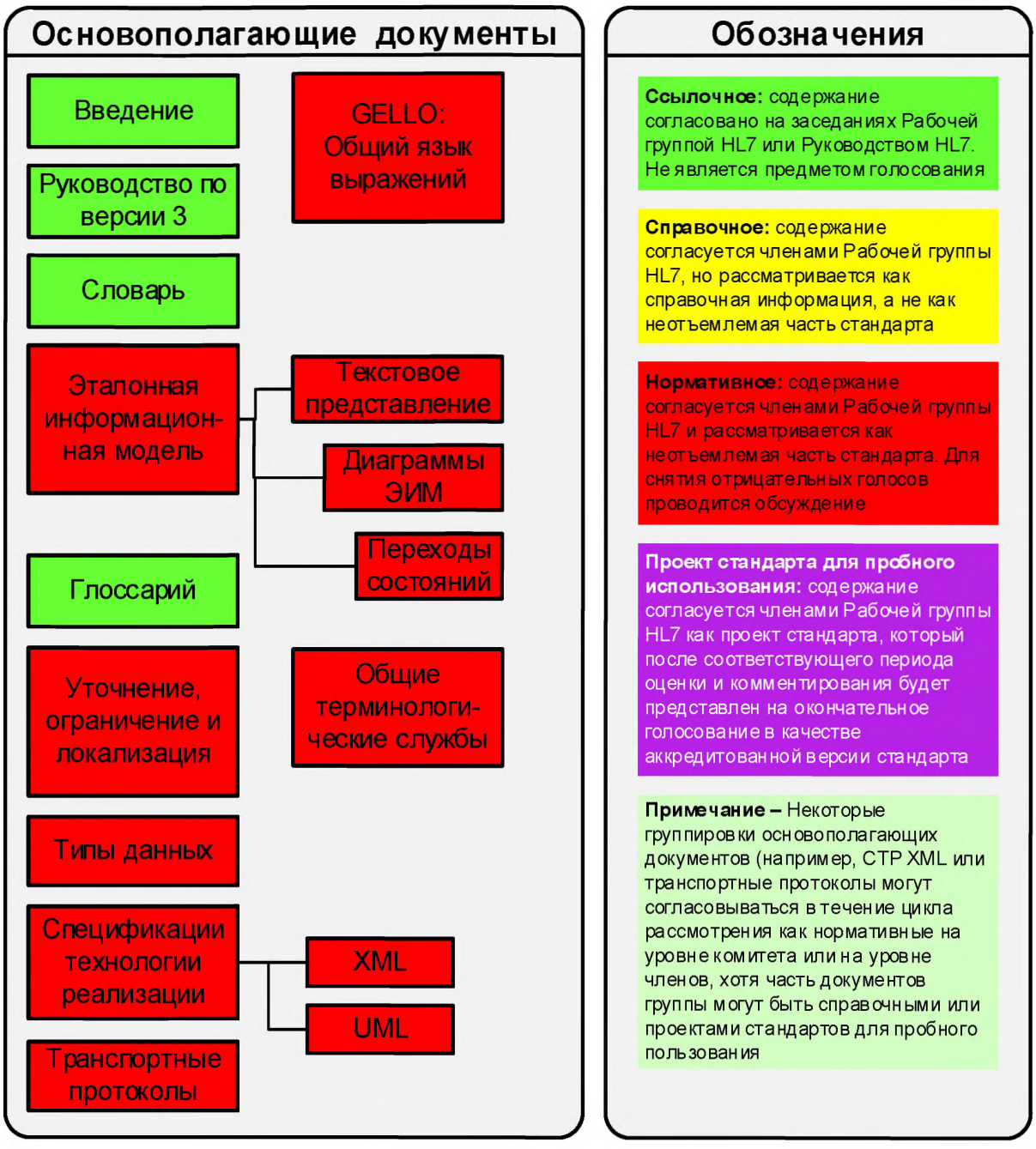


Рисунок G.1 — Основополагающие компоненты

Общее понимание основополагающих компонентов, обсуждаемых в настоящем разделе, необходимо для лучшего восприятия компонентов сообщений, описываемых в разделе «Компоненты сообщений». К основополагающим компонентам относятся:

- информационная модель: описывает типы информационных моделей стандарта HL7 версии 3, включая эталонную информационную модель RIM (Reference Information Model);
- словарь: обсуждается использование контролируемого словаря в спецификациях стандарта HL7 версии 3, а также принципы работы Технического комитета HL7 Vocabulary и методы использования и управления словарем;
- спецификация технологии реализации ITS (Implementation Technology Specification): описывает документ «Спецификация технологии реализации», а также его функцию;
- типы данных: описывает основополагающие принципы типов данных версии 3;

- типы общих элементов сообщений CMET (Common Message Element Types): описывает назначение, определения и использование CMET;

- примеры: приведен пример сообщения версии 2.4 и его эквивалент в стандарте версии 3 с использованием XML ITS (включая примечания к отображению).

G.2.2 Информационная модель

Информационная модель представляет собой структурированное описание информации в специфичном домене. Она описывает требуемые классы информационных объектов и их свойства, включая атрибуты, отношения, ограничения и состояния. В стандарте HL7 границы рассматриваемого домена изменяются от домена системы здравоохранения в целом, до конкретного контекста обменов информации, выполняемых с определенной целью. В стандарте HL7 определены различные типы информационных моделей, представляющих различные интересующие контексты.

G.2.2.1 Компоненты информационной модели

Информационная модель состоит из следующих компонентов:

- классы, их атрибуты и отношения между классами;
- типы данных всех атрибутов и словарные домены для кодированных атрибутов;
- модели перехода состояний для некоторых классов.

Информационные модели HL7 построены на Унифицированном языке моделирования UML (Unified Modeling Language) и могут быть представлены графически в виде UML-диаграмм. Язык UML представляет собой язык моделирования, объединяющий методы объектно-ориентированного моделирования Гради Буча (Grady Booch), Джеймса Рамбо (James Rumbaugh), Ивара Якобсона (Ivar Jacobson) и других авторов. Язык UML является преимущественно графическим языком с широкими возможностями и служит для представления объектно-ориентированных понятий. Более подробную информацию о языке UML можно получить на сайте или в книге «UML Distilled» Мартина Фаулера (Martin Fowler), ISBN 0-201-32563-2.

G.2.2.2 Типы информационных моделей

В процессе информационного моделирования HL7 выделяются три взаимосвязанных типа информационных моделей. Все они используют одну и ту же нотацию и имеют одинаковую базовую структуру. Модели различаются между собой по информационному содержанию, ограничениям и области применения. Определены следующие типы информационных моделей:

- эталонная информационная модель RIM (Reference Information Model) используется для представления информации, предназначенной для совместной работы в Рабочей группе HL7. Эта информационная модель охватывает рассматриваемую предметную область в целом. RIM является согласованной, совместно используемой моделью, которая служит источником структуры всех сообщений, описываемых в стандарте HL7. Как таковая, она позволяет описать согласованные данные и понятия, многократно используемые в разных информационных структурах, в том числе в сообщениях. Модель RIM развивается с помощью процесса, основанного на сотрудничестве и консенсусе, в который вовлечены Технические комитеты и Специальные группы. В этом процессе, известном как гармонизация модели, используются стандартные процедуры достижения консенсуса, включая оценку, дебаты, улучшения и согласованную доводку предложений о внесении изменений, представленных Техническими комитетами. Согласованные изменения вносятся в модель RIM для выпуска новой версии. Модель RIM намеренно является абстрактной, позволяющей представить все богатство разделов информации, совместно используемой в сфере здравоохранения. Принципы, положенные в основу этой абстракции, подробно обсуждаются во введении к описанию модели RIM;

- доменная информационная модель сообщений D-MIM (Domain Message Information Model) представляет собой уточненное подмножество модели RIM, включающее комплекс клонов классов, атрибутов и отношений, используемых для описания структуры сообщений в определенном домене (интересующей предметной области сферы здравоохранения). Модель D-MIM используется в качестве общей базовой модели, на основе которой создаются все модели R-MIM данной предметной области. Правила процесса уточнения, с помощью которого создаются модели D-MIM и R-MIM, детально обсуждаются в спецификации уточнения и локализации;

- уточненная информационная модель сообщений R-MIM (Refined Message Information Model) представляет собой подмножество модели D-MIM, используемое для представления информационной структуры сообщения или группы сообщений, к которому добавлены аннотации и уточнения, специфичные для сообщения. Содержание модели R-MIM извлекается из модели D-MIM, разработанной для предметной области, в которой используется R-MIM. Модель R-MIM может включать в себя клоны отдельных классов, которым присвоены псевдонимы, подобранные для отражения специфики конструируемых сообщений. Модель R-MIM описывает состав информации одной или нескольких абстрактных структур сообщений, называемых также Иерархическими определениями сообщений HMD (Hierarchal Message Definitions).

G.2.2.3 Статические структуры: классы и отношения

G.2.2.3.1 Классы

Класс представляет собой абстракцию сущностей или понятий, являющихся предметами интереса в заданном домене приложений. Все сущности или понятия, отнесенные к конкретному классу, обладают одними и тем же свойствами, удовлетворяющими одним и тем же правилам. Классы могут представлять людей, места, роли, сущности и события, о которых хранится некоторая информация. Классы имеют имя, описание, а также набор атрибутов, отношений и состояний.

Экземпляры классов называются объектами. В то время как классы представляют собой категории понятий, людей или сущностей, объекты представляют собой отдельные сущности этих категорий. Однако существует различие между объектами и представляемыми ими сущностями. Объекты содержат всю релевантную информацию о сущностях, известную информационной системе, но не являются самими сущностями. Например, конкретный человек Джон Доу может быть представлен в информационной системе объектом. Этот объект содержит демографические и медицинские данные Джона Доу, но все же отличается от самого Джона Доу.

G.2.2.3 2 Отношения между классами

Классы могут быть различными способами связаны с другими классами. Выделяют отношения двух типов: генерализацию и ассоциацию. Ниже они будут описаны подробнее.

Генерализации

Отношение генерализации представляет собой связь между классами, а не объектами. Оно представляет собой ассоциацию между двумя классами (родительским классом и дочерним классом), в которой дочерний класс является производным от родительского (т. е. родительский класс обобщает дочерний, а дочерний является специализацией родительского). Дочерний класс наследует все свойства родительского класса, включая атрибуты, отношения и состояния. Экземпляры дочернего класса являются также экземплярами родительского класса.

В отношении генерализации дочерний класс наследует все свойства родительского класса, включая атрибуты, отношения и состояния. Кроме того, дочерний класс может содержать дополнительные свойства, специфичные для этого класса.

Каждый дочерний класс может в свои очередь иметь собственные дочерние классы. Таким образом, понятия родительского/дочернего класса и генерализации/специализации являются родственными. Класс может одновременно являться и дочерним классом для своего родительского класса, и родительским — для своих дочерних классов. Совокупность родительских и дочерних классов с общим корневым родительским классом называют иерархией генерализации.

Обычно генерализация имеет несколько специализаций. Однако в модели должны быть представлены не все концептуальные специализации. Только те понятия, которые обладают особыми свойствами (например, атрибутами, отношениями, состояниями) моделируются ка отдельные классы. Если в модели все специализации класса полностью представлены дочерними классами, то родительский класс может быть «абстрактным». Абстрактный класс не имеет собственных экземпляров помимо экземпляров дочерних классов.

Ассоциации

Ассоциация описывает отношение между объектами. Объекты могут быть экземплярами двух различных классов или одного и того же класса (рефлексивная ассоциация). Ассоциации, как и классы, имеют экземпляры. Экземпляр ассоциации представляет собой связь между объектами и описывается ассоциацией, соединяющей классы.

В информационных моделях HL7 ассоциации имеют, по крайней мере, два конца. Каждый конец экземпляра ассоциации соединяется с одним и только одним объектом. Однако объект может быть связан с более чем одним объектом того же класса с помощью той же самой ассоциации. В этом случае существует несколько экземпляров ассоциации, каждый из которых соединяет ровно два объекта. Количество экземпляров ассоциации, связанных с одним объектом, регулируется кратностью ассоциации.

Кратность ассоциации определяет минимальное и максимальное количество объектов каждого класса, участвующих в ассоциации. Кратность задается парой порядковых чисел (неотрицательных целых) min..max. Нижняя граница кратности, min — неотрицательное число, обычно ноль или единица. Верхняя же граница кратности, max, является целым числом, большим или равным min, обычно единица, или не ограничена, что обозначается звездочкой (“*”)[65]. Возможные значения кратности приведены в таблице G.1.

Таблица G.1 — Возможные значения кратности

Кратность	Описание
0..1	минимальное значение 0, максимальное — 1
0..n	минимальное значение 0, максимальное — целое число n, где n > 1
0..*	минимальное значение 0, максимальное не ограничено
1	сокращенная запись «1..1» — минимальное и максимальное значение равны 1

[65] В других моделях буквы «n» или «m» в нижнем регистре иногда используются для обозначения целочисленных переменных. Такая нотация не определена в руководстве UML Notation Guide (руководство по условным обозначениям в языке UML). Проблема использования букв вместо символа «*» состоит в том, что повторное использование буквы может привести к неправильному представлению одного и того же целого числа в разных местах. Далее, «n» обозначает некоторую специфицированную границу, а символ «*» — неограниченную кратность, никак не указывая какую-либо зависимость между такими кратностями.

Окончание таблицы G.1

Кратность	Описание
1..n	минимальное значение 1, максимальное — целое число n, где n > 1
1..*	минимальное значение 1, максимальное не ограничено
n1..n2	минимальное значение — целое число n1, максимальное — целое число n2, где n1 > 1, а n2 > n1
n..*	минимальное значение — целое число n, где n > 1, максимальное – не ограничено

G.2.2.4 Атрибуты

Атрибуты класса являются ключевыми компонентами информационной модели. Они являются источником всего информационного содержания стандарта HL7. Большинство атрибутов являются описательными, и отражают особенности классов, важные для взаимодействия между информационными системами здравоохранения. Помимо описательных атрибутов, в информационной модели существует три специальных вида атрибутов: атрибуты идентификации, атрибуты классификации-и атрибуты состояний. Ниже они обсуждаются более детально.

Атрибуты идентификации

Атрибуты идентификации могут использоваться для идентификации экземпляра класса. Иногда для этого может понадобиться более одного атрибута. Атрибуты идентификации всегда имеют значения, являющиеся уникальными для всех экземпляров данного класса. Поскольку уникальность является статической характеристикой, значения этих атрибутов никогда не изменяются. Атрибутам идентификации присвоен тип данных «множество идентификаторов экземпляров» (SETII) типа данных. Обычно они именуются «id», что позволяет указать несколько идентификаторов.

В модели RIM примерами атрибутов идентификации являются Entity.id и Act.id, уникально идентифицирующие соответственно отдельную сущность (Entity) или действие (Act). В каждом случае атрибуты идентификации представляют множество экземпляров идентификаторов. Это означает, что у разных экземпляров классов Entity и Act могут быть уникальные идентификаторы. Идентификаторы экземпляров класса Entity могут содержать серийные номера устройств, номера социального страхования, номера водительских прав. Идентификаторы экземпляров класса Act могут содержать номера заказов, присваиваемых заказчиком, номера заказов, присваиваемых исполнителем, и т. д.

Атрибуты классификации

Атрибуты классификации представляют собой важный аспект основополагающих классов модели RIM (сущность Entity, роль Role и действие Act). Данный вид атрибутов именуется «classCode». С помощью атрибутов классификации информационная модель может быть сделана значительно более гибкой и расширяемой. Словарные домены атрибутов классификации содержат коды всех специализаций базового класса. Например, словарный домен атрибута Entity.classCode включает в себя коды, обозначающие живого субъекта, организацию, место и материал. Кроме того, словарный домен может содержать элементы, которые в модели явно не представлены классами. Например, group (группа) является допустимым значением атрибута Entity.classCode (или специализации класса Entity), но не отражается в модели в виде класса. Это и обеспечивает гибкость и способность к расширению.

Структурные атрибуты

Структурные атрибуты — атрибуты, чьи кодированные значения необходимы для полной интерпретации классифицируемого ими класса. К ним относится небольшое число обязательных атрибутов, включающих классифицирующие атрибуты: classCode, описанный в предыдущем параграфе, а также moodCode, typeCode и determinerCode. Все четыре никогда не встречаются в одном классе. Например, у классов Act и Entity нет атрибута determinerCode. Как и атрибут classCode, каждый структурный атрибут имеет ограниченный словарь значений, управляемый HL7. Например, для атрибута наклонения (mood) в классе действия Act исользуется словарь actMood.

Атрибуты состояния

Атрибут состояния используются в существенных классах (в классах, которые Технический комитет определил как основные для комплекса сообщений). Он содержит значение, кратко характеризующее текущее (именованное) состояние класса. Существенный класс должен содержать лишь один атрибут состояния. Атрибуту состояния должен быть присвоен тип данных «множество кодовых значений», что позволяет указывать несколько флагов состояний. Атрибуты состояния именуются «status_cd» и связаны с описанными в HL7 словарями предметных областей, соответствующих конечному автомату, определенному для существенного класса. Например, домен атрибута Act.status_cd содержит значения active (активный), suspended (приостановленный), cancelled (отмененный), completed (завершенный) и aborted (прекращенный).

G.2.2.5 Ограничения

Ограничения сужают область возможных значений атрибута. К ограничениям относятся: ограничения словаря предметной области (например, значение атрибута должно быть кодом LOINC), ограничения диапазона значений (например, атрибут должен содержать значения с плавающей точкой от 0 до 1) и т. д. Хотя термин «ограничение» имеет негативный оттенок, цель определения ограничений в большей степени является описание правильного использования класса или атрибута.

Ограничения могут быть указаны в моделях RIM, D-MIM, R-MIM или в иерархических описаниях сообщений (HMD). Будучи указанным в модели RIM, ограничение относится к атрибуту во всех сообщениях, содержащих этот атрибут. Будучи указанным в D-MIM или R-MIM, оно специфично для всех сообщений, произведенных из этих моделей. Кроме того, ограничения могут быть указаны в иерархическом описании сообщения HMD. В этом случае они могут быть сделаны специфичными для отдельных структур сообщений. Ограничения, определенные на более высоком уровне (например, в модели RIM), могут быть дополнены ограничениями на более низком уровне (например, в модели D-MIM или в HMD). Однако подчиненные ограничения должны удовлетворять ограничениям более высокого уровня. Ограничения более высокого уровня не могут быть отменены на более низком уровне. (Подробнее ограничения обсуждаются в нормативной части раздела «Уточнение, ограничения и локализация».)

G.2.2.6 Динамическое поведение: состояния и переходы

Поведение класса описывается диаграммой перехода состояний, ассоциированной с этим классом в информационной модели. Диаграммы перехода состояний, показывающие все возможные состояния класса, создаются для классов, являющихся главными субъектами взаимодействия. Подобные классы называются существенными классами. Иногда взаимодействия происходят из-за изменений состояния существенного класса. Например, в качестве существенного класса может выступать класс действия Act. Словарь предметной области атрибута Act.status_cd описывает определенные состояния этого класса: active (активный), suspended (приостановленный), cancelled (отмененный), completed (завершенный) и aborted (прекращенный). На диаграмме перехода состояния класса изображаются в виде прямоугольников, в центре которых указывается имя состояния. Изменение состояния называется переходами состояний и изображаются на диаграмме линией со стрелкой, показывающей направление перехода. Примером перехода между состояниями может служить изменение состояния active класса Act на состояние complete. Изменение состояния (переход) связано с инициирующим событием, которое и вызывает переход. В приведенном примере инициирующим событием может выступать выполнение заказа. При этом заказ является особым видом класса Act. Переход заказа из активного состояния (active) в выполненное (complete) происходит после выполнения заказа. На рисунке G.2 изображены состояния, инициирующее событие и переход состояния.

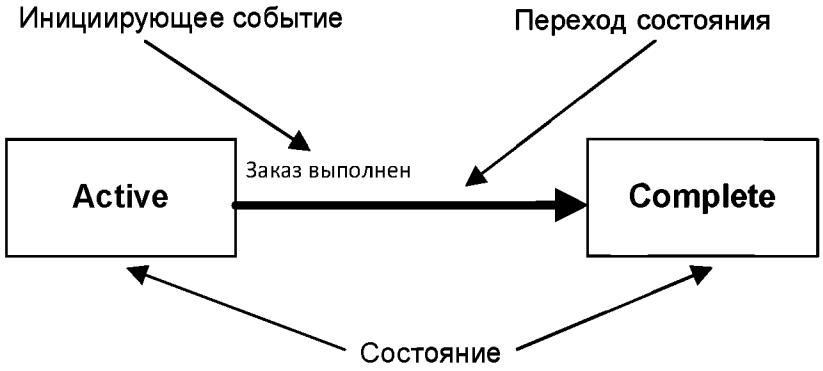


Рисунок G.2 — Переход состояния

G.2.3 Словарь

В рамках стандарта HL7 словарь предметной области представляет собой совокупность всех понятий, которые могут использоваться в качестве значений экземпляров кодируемых полей или атрибутов. Например, класс Living_subject (живой субъект) содержит кодируемый атрибут административного пола administrative_gender_cd. Если атрибут administrative_gender_cd станет частью иерархического описания сообщения HMD, а затем экземпляр сообщения создастся как часть реализованного интерфейса, то можно будет ожидать, что атрибут administrative_gender_cd будет иметь значение кода, обозначающего либо мужской, либо женский пол. В данном примере «мужской пол» и «женский пол» являются понятиями, и может существовать несколько схем кодирования, охватывающих эти понятия.

Необходимо отметить, что словарь предметной области состоит из множества понятий, а не из отдельных слов или кодов. В различных вариантах реализации интерфейса одно и то же понятие может быть представлено, используя разные системы кодирования. Таким образом, у каждого понятия в словаре есть отношение «один ко многим» с кодами, которые могли бы использоваться для представления понятия в экземпляре сообщения.

Система кодирования в основном является схемой представления понятий с использованием (обычно) коротких идентификаторов, обозначающих понятия, являющиеся элементами системы. Схема кодирования определяет множество уникальных кодов понятий.

G.2.3.1 Словари предметной области и атрибуты модели RIM

В модели RIM каждый кодируемый атрибут (например, имеющий тип данных CD, CE, CR, CS или CV) ассоциируется с одним и только одним словарным доменом. Например, словарным доменом атрибута administrative_

gender_cd является домен AdmistrativeGender. Некоторые словарные домены ассоциируются с несколькими атрибутами модели RIM. Например, словарный домен PhysicalQuantity ассоциируется с атрибутами diet_carbohydrate_qty и diet_energy_qty. Таблица словарного домена может быть как таблицей, определенной HL7, так и внешней схемой кодирования, признаваемой Рабочей группой HL7 (например, LOINC, SNOMED) или их сочетанием. Она может также содержать коды, определенные на местном уровне.

Таблицы словарных доменов, разработанные Рабочей группой HL7 для кодируемых атрибутов классов, содержатся в хранилище, которое ведется Рабочей группой HL7. Для распечатки содержания словарного домена из него можно извлечь различные представления (view). Представления задаются в табличной форме и содержат список значений доменного словаря, а также таблицы доменов HL7 и список кодируемых атрибутов. Внешние словарные домены, признаваемой Рабочей группой HL7, перечислены в списке внешних доменов. Эти таблицы и имена атрибутов содержат ссылки на модель RIM.

Таблица значений словарного домена, определенного Рабочей группой HL7, содержит мнемонический код, идентификатор понятия, отображаемое имя и определение/описание каждого значения. В этой таблице показана также информация обо всех иерархических отношениях между значениями домена.

В таблицах доменов HL7 и таблице кодируемых атрибутов поименованы кодируемые атрибуты модели RIM, значения которых берутся из данного словарного домена.

Таблица внешних доменов содержит идентификатор понятия, определяющее выражение, сокращенное название системы кодирования, описание и ссылку на источник таблицы внешнего домена.

G.2.3.2 Квалификаторы словарных доменов

Кодируемые поля содержат две части информации, относящейся к словарю: имя словарного домена и квалификатор расширения. Квалификатор расширения может принимать одно двух значений: CNE (coded no extensions — кодируемое без расширений) и CWE (coded with extensions — кодируемое с расширениями).

Для каждого кодируемого атрибута модели RIM в ее описательной части указаны имя словарного домена и квалификатор расширения. Эта спецификация содержится в первой строке описания кодируемого атрибута класса модели RIM. Она форматируется в соответствии со следующим примером записи информации об атрибуте Entity. class_cd:

Домен: «EntityClass» (CNE)

Имя словарного домена (EntityClass) является гиперссылкой на строку, описывающую этот домен в таблице значений словарных доменов. Тем доменам, значения которых еще не разработаны, доменное имя присваивается, но таблица значений не содержит никаких данных.

Значение CWE квалификатора расширения означает, что для удовлетворения требованиям конкретной реализации набор допустимых кодов может быть расширен. Когда кодируемый атрибут передается в сообщении, то при отсутствия необходимого понятия в стандартном словарном домене вместо стандартного кода может быть передан местный код понятия или его текстовое описание.

Значение CNE квалификатора расширения означает, что набор допустимых кодов фиксирован и не может быть расширен. Понятие, принадлежащее конкретному домену, должно быть указано в передаваемом сообщении в виде значения кодируемого поля. Если же этому полю нельзя присвоить значение кода понятия, указанного в данном словарном домене, то оно не может быть включено в сообщение. Если в сообщении поле с квалификатором CNE является обязательным, а значение поля невозможно взять из словарного домена, то сообщение не может быть создано.

Квалификатор Realm (сфера действия) позволяет задать словарный домен кодируемого атрибута в зависимости от географических, организационных и политических условий, в которых используется стандарт HL7. Например, данный квалификатор позволяет хранить в домене Gender (пол) различные наборы значений в зависимости от того, где используются сообщения, соответствующие стандарту HL7: в Японии или США.

Все квалификаторы доменов определены в домене VocabularyDomainQualifier.

G.2.3.3 Использование словарных ограничений

Ограничения кодируемых атрибутов классов модели RIM состоят в требования выбора их значений из соответствующих доменных словарей.

Указание словарного домена в модели RIM всегда относится к полному словарному домену. Таким образом, на уровне модели RIM не существует специализации, основанной на использовании квалификатора сферы действия Realm или контекста и потребностей специфичного сообщения. Поскольку для удовлетворения особенностей конкретного контекста сообщений атрибуты модели RIM специализируются, то домен значений атрибута может быть сужен (ограничен) для отражения этих особенностей.

Домен, ограниченный с учетом квалификатора Realm и системы кодирования, называется набором значений. Ограничение словарного домена может быть применено на любом уровне ниже модели RIM. Оно представляет собой выражение, описывающее, каким образом данный набор значений был получен из домена, определенного в модели RIM. Ограничение словарного домена в моделях R-MIM, D-MIM и т. д. содержит имя набора значений и его список квалификаторов домена. Выражения, описывающие ограничения доменов, хранятся только в таблице описания набора значений.

При создании ограничений словарных доменов общее правило состоит в том, что при специализации, предназначенной для решения конкретной задачи, область применения домена может быть сокращена, но ее семантика никогда не может быть расширена. Из этого общего правила вытекают следующие следствия:

- словарный домен кодируемого элемента или атрибута, используемого на любом уровне специализации ниже модели RIM, должен являться подмножеством словарного домена, определенного для этого атрибута в модели RIM. Отметим, что у атрибутов, имеющих квалификатор расширения CWE, местные коды являются разрешенными расширениями предметной области, но при этом не допускается расширение семантики области применения атрибута;

- поскольку квалификатор расширения CNE может быть применен на всех уровнях спецификации атрибутов (RIM, D-MIM, R-MIM, CMET, HMD), то квалификатор CNE должен распространяться на все дальнейшие ограничения данного домена. Например, если кодируемый атрибут имеет квалификатор CNE в модели RIM, то в моделях D-MIM, R-MIM и спецификациях CMET, HMD для домена этого атрибута также должен быть указан квалификатор CNE. Если в модели RIM словарный домен имеет квалификатор расширения CWE (кодирование с расширениями), то при последующих ограничениях домена этого атрибута можно указать как квалификатор CNE, так и квалификатор CWE.

G.2.4 Спецификации технологии реализации

Спецификация технологии реализации (ITS) определяет способ представления объектов модели RIM в передаваемых сообщениях. Она затрагивает шестой и пятый уровень Модели взаимосвязи открытых систем ИСО. Первой спецификацией ITS, принятой Рабочей группой HL7, была технология на основе языка XML. В данном руководстве приводится краткая информация о спецификации XML ITS. Спецификация UML ITS была разработана недавно и является частью текущих пакетов документов, поставленных на голосование.

Стандарт HL7 определяет сообщения на абстрактном уровне. В названии стандарта HL7 цифра «7» соответствует седьмому, прикладному уровню Модели взаимосвязи открытых систем ИСО. На этом уровне основное внимание уделяется семантике содержания сообщений, а не тому, как они представлены или как закодированы для передачи.

Модель абстрактного сообщения HL7 версии 2 представляла собой описание сегментов и полей. Она описывала конкретную схему символьного представления экземпляров абстрактных сообщений — так называемое «кодирование с вертикальной чертой». Информация из домена Health (здоровье), описанная на семантическом седьмом уровне, представлялась в версии 2 в виде сегментов и полей, состоящих из ASCII-символов со множеством символов вертикальной черты «|». Совместимое с данным подходом, новое XML-кодирование в версии 2 заключается в основном в непосредственной замене кодирования с вертикальной чертой.

Модель абстрактных сообщений версии 3 основана на модели RIM. Определения сообщений в стандарте HL7 версии 3 можно рассматривать как передачу графов объектов RIM от отправителя к получателю. Спецификация ITS может наилучшим образом описать эти сообщения, используя соответствующие представлений объектов, атрибутов и типов данных.

Общедоступный стандарт языка XML определяет формат передачи сообщений HL7. Он описывает способ представления XML-документов в виде потока 8-битовых байтов, определяет правила соответствия открывающихся и закрывающихся тегов. Это соответствует пятому уровню Модели взаимосвязи открытых систем ИСО. Объектная модель документа XML (XML DOM) описывает абстрактное дерево синтаксического разбора XML-документов и соответствует шестому уровню этой модели.

Спецификация ITS необходима для описания формата передачи сообщений, соответствующих стандарту HL7. Из всего семейства стандартов языка XML для этих целей были выбраны рекомендации XML Schema (XML-схема). С помощью ограничений, наложенных в XML-схемах, можно указать, что приемлемо в XML-документах. С помощью стандартных средств обработки XML-документов разработанная схема HL7-сообщения может использоваться для проверки того, соответствует ли конкретное сообщение этой схеме. Самая обширная часть спецификации ITS связана с типами данных, в которой фрагменты XML-схем приведены для каждого из 42-х типов данных, поддерживаемых стандартом HL7. Кроме того, в ITS описаны следующие объекты:

- имена классов;
- ассоциации классов;
- имена атрибутов.

В рабочей реализации схемы используются следующие практические соображения:

- использование подкаталога фрагментов схемы для классов модели RIM позволяет без труда делать ссылки на релевантные классы из схемы конкретного сообщения;

- аналогично, использование подкаталога типов данных позволяет без труда включать в схему соответствующие фрагменты схем;

- использование подкаталога словаря позволяет включать все домены, полностью определенные в самом стандарте HL7, в качестве части другого подкаталога схемы.

Документ «Структуры XML» описывает особенности использования языка XML в спецификации ITS.

Во всех схемах, определенных в стандарте HL7, указано одно и то же пространство имен XML. Хотя рассматривались и другие варианты, было решено, что использование единого пространства имен предоставляет больше преимуществ.

Совокупность XML-экземпляров, соответствующих каждому типу сообщения, описанному в спецификации HMD, имеет структуру, содержащую элементы, соответствующие клонам классов, а также элементы и атрибуты, соответствующие атрибутам клонов классов. Структурные атрибуты клона класса представляются в виде XML-атрибутов, а остальные атрибуты классов, определенные в стандарте HL7, представлены в XML-экземпляре

дочерними элементами. Имена элементов, представляющих клоны классов, определяются именами ассоциаций.

Некоторые описания типов сообщений ссылаются на типы общих элементов сообщений CMET. В данной спецификации описано также, каким образом кодируются CMET, вложенные в экземпляр сообщения.

Все АКД-документы наследуются от единственного типа сообщения, однако для кодирования HL7-сообщения требуется больше информации.

Обертки (wrappers) предназначены для предоставления дополнительной общей информации, требуемой как при передаче сообщений (Transmission Wrapper — обертка передачи сообщений), так и при их семантической интерпретации (Control Act Wrapper — обертка управляющего действия). Взаимодействие (interaction) определяет все сообщение, включающая его тип, а также ассоциированные с ним обертки передачи сообщений и управляющих действий. В данной спецификации описано представление оберток в XML-экземпляре сообщения.

Каждому атрибуту, определенному в стандарте HL7, присвоен определенный тип данных. XML-реализация этих типов данных приведена в отдельном голосуемом документе, а именно, XML ITS datatypes (типы данных ITS XML).

1 На рисунке показаны шаги использования ITS при передаче информационных объектов из базы данных системы-отправителя в базу данных системы-получателя:

2 Приложение, отправляющее информацию («отправитель»), хранит ее в своем собственном формате базы данных.

3 Отправитель логически представляет необходимую информацию в виде графа объектов модели RIM.

4 Используя формат сообщений, описанный в спецификации HMD, и алгоритм, описанный в ITS, отправитель представляет объекты модели RIM в виде XML-документа, например, с помощью построения дерева объектной модели документа (DOM).

5 Отправитель создает XML-сообщение с помощью сериализации дерева DOM.

6 Отправитель передает XML-сообщение принимающему приложению («получателю») по протоколу TCP/IP, по электронной почте, или с помощью иного уровня транспорта сообщений.

7 Получатель распаковывает XML-сообщение, полученное через транспортный уровень.

8 Получатель удаляет обертки сообщения, предусмотренные в версии 3, и с помощью стандартного синтаксического разборщика преобразует полученный результат в дерево DOM.

9 Затем получатель интерпретирует дерево DOM, «обращая» преобразование, описанное в ITS, и при необходимости строит граф полученных объектов модели RIM.

В заключение получатель сохраняет эти данные в своем собственном формате базы данных.

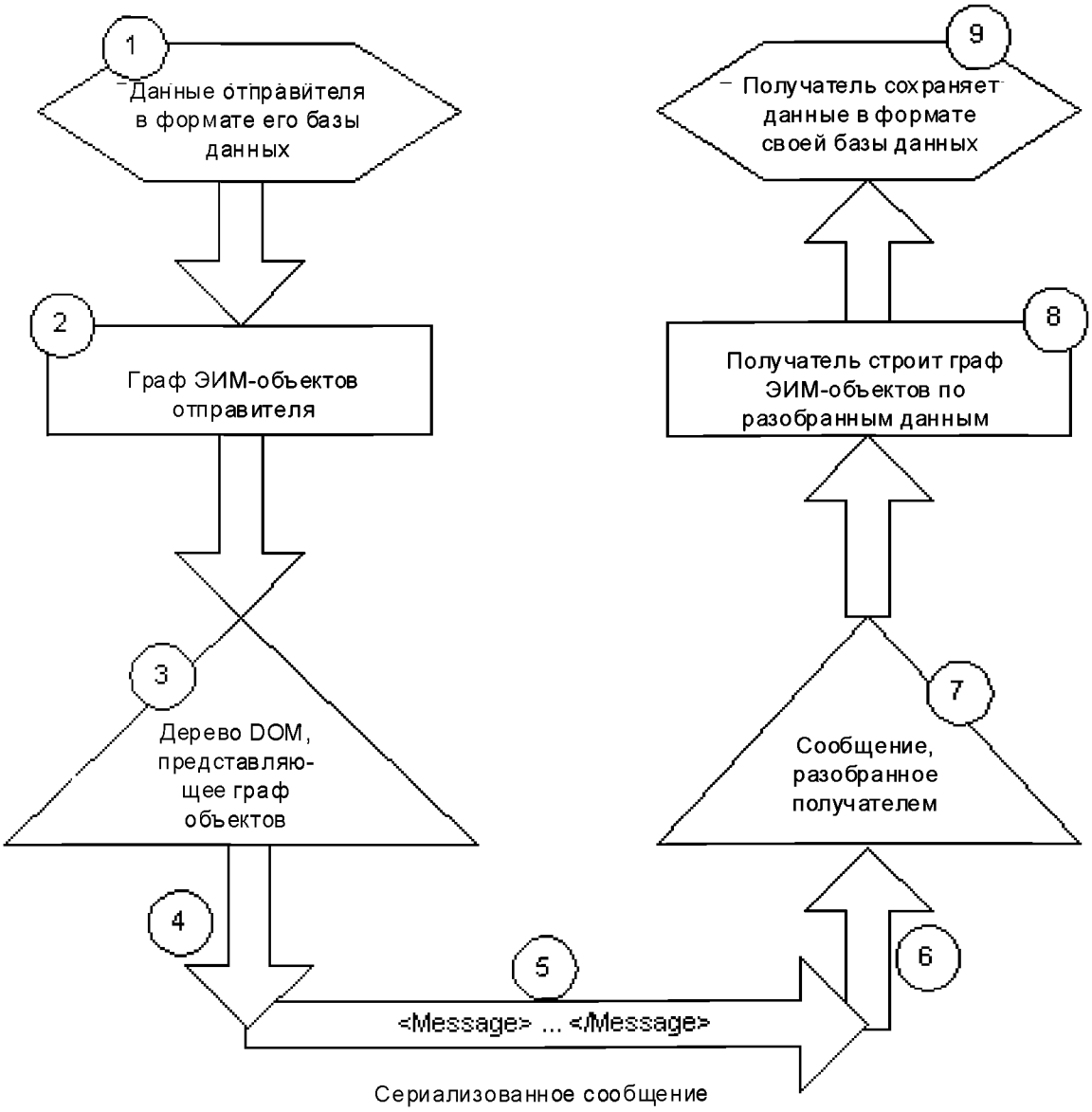


Рисунок G.3 — Использования ITS при передаче информационных объектов

G.2.5 Типы данных

Типы данных являются основными «строительными блоками» атрибутов. Они определяют структурный формат данных атрибута и множество допустимых значений, которые могут быть присвоены атрибуту. Некоторые типы данных имеют очень узкое внутреннее семантическое содержание, и семантический контекст такого типа данных переносится соответствующим ему атрибутом. Однако в HL7 определены также достаточно широкие типы данных наподобие PNMP (person name part — часть фамилии, имени или отчества лица), который содержит всю структуру и семантику, необходимую для представления фамилии, имени, отчества лица. Каждому атрибуту в модели RIM соответствует только один тип данных, а каждому типу данных соответствует ноль или более атрибутов.

Типы данных описываются в двух документах из комплекса документов, составляющего стандарт HL7 версии 3. Документ Data Types, Unabridged (полное описание типов данных) содержит подробное, строгое описание типов данных версии 3 и предназначен для тех, кто желает детально изучить типы данных. Менее формальное описание типов данных включено в спецификацию ITS. Глядя правде в глаза, можно утверждать, что никакая отдельная спецификация ITS не может включать в себя все, что описано в спецификации Data Types. Спецификация XML ITS не является исключением из этого правила, но, тем не менее, рекомендуется читателям стандарта HL7 как лучшее руководство для ознакомления с поддерживаемыми типами данных и особенностям их реализации. В таблице G.2 приведены основные категории типов данных.

Таблица G.1 — Категории типов данных

Категория типов данных	Описание	Примеры
Базовые типы данных	К базовым типам относится 31 тип из 42 типов, определенных в стандарте HL7	Текст, коды, идентификаторы, имена, адреса, количества
Параметризованные коллекции	Типы данных, которые могут содержать несколько значений. Их не являются полными типами данных, поскольку с каждой коллекцией может быть ассоциирован любой тип данных из тех, что указан в других описанных здесь группах	Sequence, BAG и SET
Расширения параметризованного типа	Возможность расширять существующие типы данных с помощью формализованного языка расширений	Не поддерживается в текущей версии XML ITS
Спецификации времени	Все требования к указаниям моментов времени	IVL, интервалы времени

Основной особенностью типов данных является возможность приведения конкретного типа данных к более простому. Например, тип кодируемых данных CV может быть приведен к типу данных CS, который содержит лишь атрибуты Code (код) и Display Name (отображаемое имя). Спецификация XML ITS представляет такие приведения типов, всегда показывая, когда более простой тип данных является ограничением более сложного типа. Инструментальное средство на основе Visio сначала может предложить более сложный тип данных, определенный для атрибута, но затем позволяет выбрать любой из более ограниченных типов, что при возможности облегчает выбор более простого типа данных.

Специфичная проблема интерпретации спецификации типа данных была обнаружена для типа данных SET (множество). Спецификация определяет SET<1..1> как единственный экземпляр атрибута в составе множества, а не единственное множество, однако во многих случаях интерпретация осуществлялась неверно.

G.2.6 Типы общих элементов сообщений

Типы общих элементов сообщений CMET являются рабочими инструментами, создаваемыми конкретным комитетом для представления общих, полезных и повторно используемых понятий. Они обычно «потребляются» или используются как создавшим их комитетом, так и другими комитетами. Поскольку они предназначены для общего использования в сообщениях, сконструированных всеми комитетами, то они предлагаются, рассматриваются и управляются специальной группой CMET комитета MnM. Специальная группа CMET гармонизирует все типы CMET и управляет работой над ними.

Типы CMET можно рассматривать как фрагмент типа сообщения, повторно используемый в других типах сообщений. Любой тип сообщения, включая другие типы CMET, может ссылаться на тип CMET. Например, несколько комитетов могут потребовать использовать общее понятие, описывающее физическое лицо в роли пациента. Для представления этого понятия в виде типа сообщения, клонирующего роль, выполняемую лицом, со всеми соответствующими атрибутами, может быть определен тип CMET. Затем этот тип CMET используется всеми заинтересованными комитетами для единообразного представления данного понятия.

Тип CMET производится от одной модели D-MIM, определенной конструирующим его комитетом. Его содержание представляет собой непосредственное подмножество клонов классов и атрибутов, определенных в этой модели D-MIM, и не включает в себя содержание из других моделей D-MIM. (Оно не «охватывает» несколько моделей D-MIM.)

G.2.6.1 Семантическая классификация

Все типы CMET могут быть разделены на две категории: атрибутивность (Attribution) и выбор генерализации-специализации (Gen-Spec Selectivity). Назначение типа CMET можно изобразить, представляя его точкой в пространстве. Как правило, выбор генерализации-специализации соответствует уровню спецификации CMET в иерархическом описании сообщения HMD, а атрибутивность — спецификации типа сообщения в HMD.

Ось 1. Атрибутивность

Ось атрибутивности типу сообщения CMET и всегда содержит, по меньшей мере, три варианта:

- универсальный (universal) — данный вариант включает все атрибуты и ассоциации, представленные в модели R-MIM. Любые необязательные и не требуемые атрибуты или ассоциации могут присутствовать или отсутствовать в зависимости от ограничений кратности;
- детальный (detailed) — является строгим подмножеством универсального варианта и представляет собой именованный вариант, который может ограничивать универсальный вариант. Любые необязательные и не требуемые атрибуты или ассоциации могут присутствовать или отсутствовать в зависимости от ограничений кратности;
- идентифицированный (identified) — является строгим подмножеством универсального и детального варианта. Может включать только обязательные и требуемые атрибуты. Другие варианты не могут быть заменены во время исполнения.

В рамках данной категории могут быть определены и идентифицированы другие варианты, являющиеся строгими подмножествами универсального варианта. Эти варианты занимают промежуточное место между детальными и идентифицированными вариантами, накладывая ограничения на один или несколько атрибутов/ассоциаций универсального варианта.

Ось 2. Выбор генерализации-специализации

На данной оси указаны иерархии генерализации-специализации, допустимые в типе сообщения. Это соответствует вариантам уровня в иерархическом описании сообщения HMD. Всегда существует хотя бы один вариант, а именно, полностью общий (fully general). Следующие варианты могут присутствовать или отсутствовать:

- общий (general) — данный вариант содержит все допустимые специализации любой иерархии генерализации-специализации, представленной в модели R-MIM. Данный вариант используется в том случае, когда разработчик модели R-MIM хочет, чтобы любая из перечисленных специализаций могла быть выбрана во время исполнения.
- специфичный (specific) — вариант, оставляющий из всех доступных специализаций типа сообщения только избранные. Данный вариант используется в том случае, когда разработчик модели R-MIM хочет ограничить допустимый выбор типов сообщений, осуществляемый во время исполнения.

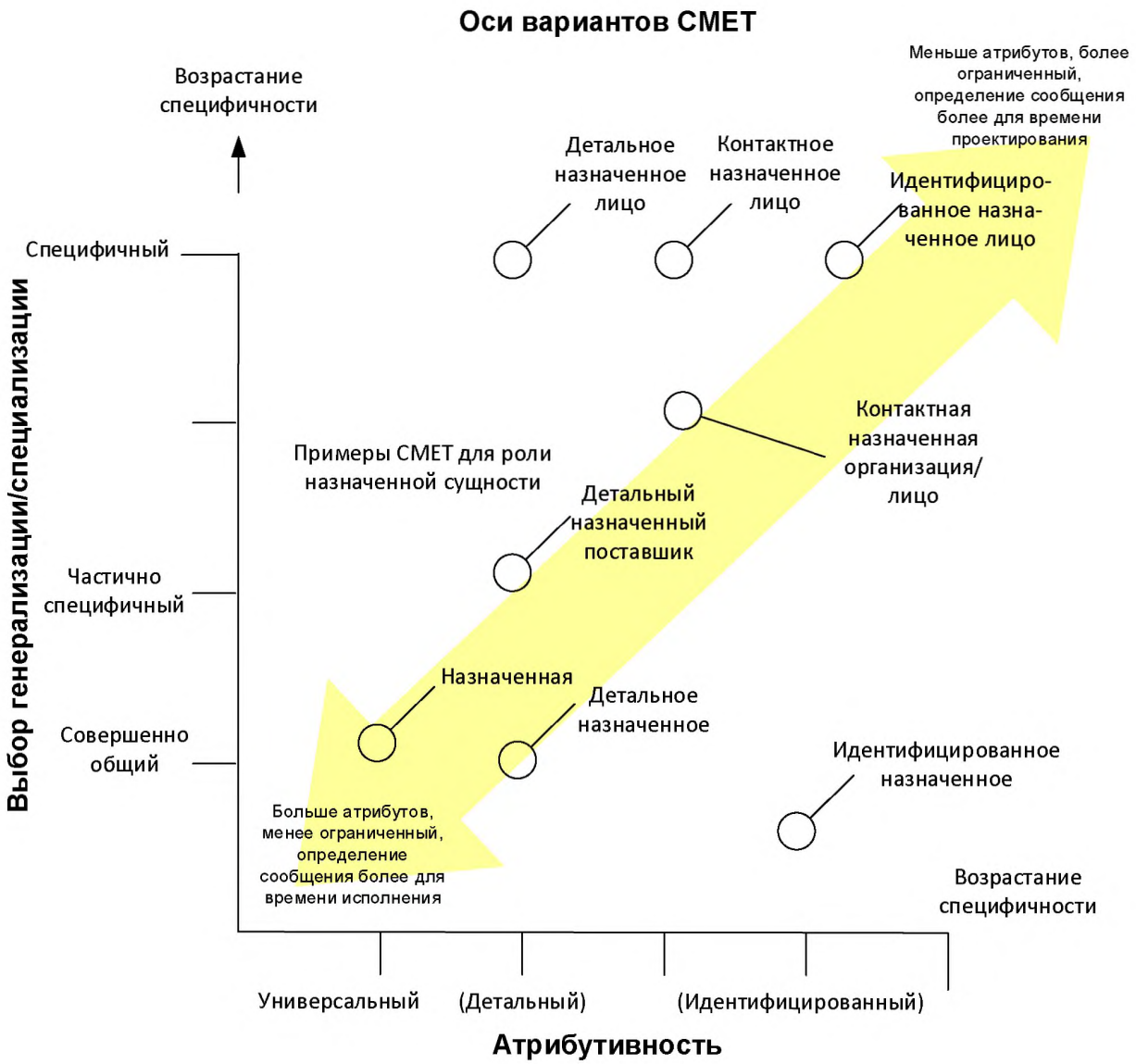


Рисунок G.4 — Варианты CMET

G.2.6.2 Описание и использование типов CMET

Типы CMET предназначены для представления общих, повторно используемых шаблонов. Однако при этом не предполагается, что они могут использоваться в роли механизма упрощения диаграмм, отображающих модели D-MIM или R-MIM. Это означает, что тип CMET должен быть представлен в виде полностью развернутой диаграммы экземпляра в той модели D-MIM, где он определен. Для обозначения корневого класса CMET на диаграмме размещается точка входа.

Отсюда следует, что тип CMET несет только справочную информацию и не предназначен для переноса информации об изменении в сообщении. Проще говоря, тип CMET должен использоваться для облегчения идентификации центрального класса, являющегося целью операция изменения. Сам по себе тип CMET не должен переносить обновляемые значения.

Точка входа

Точка входа в типе CMET представляет собой одну точку, в которой тип CMET «прикрепляется» к сообщению, которое содержит его или ссылается на него. Визуально точка входа отображается таким же образом, как и другие точки входа в диаграмме модели D-MIM. Точка входа определяет класс модели RIM, называемый корневым классом, с которого начинается иерархическое описание сообщения HMD для типа CMET. При использовании типа CMET идентификатор точки входа размещается в сообщении в этой точке.

Вид

Вид типа CMET — тип корневого класса, вместе с любыми релевантными внешними атрибутами, например, class_code. В настоящее время видами CMET являются Act (действие), Role (роль), и Entity (сущность).

Типы ролей

В случае, если видом типа CMET является Role, он может «участвовать» (participate), «исполнять или участвовать» (play or participate) или «контролировать или участвовать» (scope or participate). Это означает, что тип CMET вида Role может быть использован как цель класса Participation, как цель исполняющей сущности или класса Participation, либо как цель контролирующей сущности или класса Participation. Отметим, что тип CMET данного вида по-прежнему может содержать лишь одну точку входа, и не может одновременно являться целью и класса Participation, и исполняющей сущности; должно быть или-или.

Точка выхода

Тип CMET не содержит каких-либо точек выхода. Это означает, что тип CMET должен быть «завершающей» структурой типа сообщения, т. е. нельзя покинуть CMET и продолжить обход других частей модели R-MIM для их извлечения в иерархическое описание сообщения HMD.

Иерархии генерализации-специализации (выбор)

С помощью типа CMET можно осуществить выбор между экземплярами нескольких специализаций иерархии «Генерализация-специализация». Подобная структура называется структурой выбора. Структуры выбора подразделяются на следующие две категории:

- выбор между классами различного типа в окончательном типе сообщения откладывается до момента создания экземпляра сообщения. В этом случае экземпляр сообщения должен позволять выбирать лишь один элемент сообщения. При этом модуль формирования экземпляров сообщений просто выбирает соответствующее содержание сообщения во время исполнения;
- выбор ограничивается в модели R-MIM или в иерархическом описании сообщения HMD путем понижения до единственного типа класса. Выбора исключается из окончательного типа сообщения и не доступен модулю формирования экземпляров сообщений. Данный тип ограничения возникает в результате выбора производных типов сообщений, использующих тип CMET, во время проектирования.

G.2.6.3 Артефакты спецификации технологии реализации

С помощью методологии стандарта HL7 осуществляется конструирование иерархического описания сообщения HMD и определений типов сообщений, которые затем реализуются в соответствии со спецификацией ITS. В общем случае конструируется XML-схема описания типов сообщений CMET. Затем данная схема включается в XML-схему любого типа сообщения, использующего данный тип CMET. Поэтому, поскольку по типу CMET создается схема типа сообщения, то он используется для формирования отдельных фрагментов сообщения, используемых в более крупной конструкции.

G.2.6.4 Визуальное и табличное представление типа CMET

Типы CMET обычно представлены визуальной диаграммой, понижаемой до уровня иерархического описания сообщения HMD. Затем они более точно определяются в производной табличной форме, как и другие типы сообщений. Этот поход иллюстрируется в следующих подразделах.

Диаграмма модели D-MIM

Типы CMET получаются в результате итерационного уточнения модели предметной области. Верхний уровень, модель всей предметной области, визуально представляется в виде диаграммы модели D-MIM, содержащей несколько точек входа. Каждая точка входа идентифицируется именем и идентификатором объекта. Таким образом, каждая точка входа определяет общую базу, или корневой класс моделей R-MIM и иерархических описаний сообщения HMD. С их помощью можно сконструировать любой тип сообщения, включая тип CMET. На этом уровне в качестве объектов выступают диаграммы экземпляров модели R-MIM, представляющие абстрактные семейства типов сообщений, полученные из данной модели D-MIM.

Точки входа типов CMET включаются в диаграмму модели D-MIM, чтобы указать их место в модели предметной области.

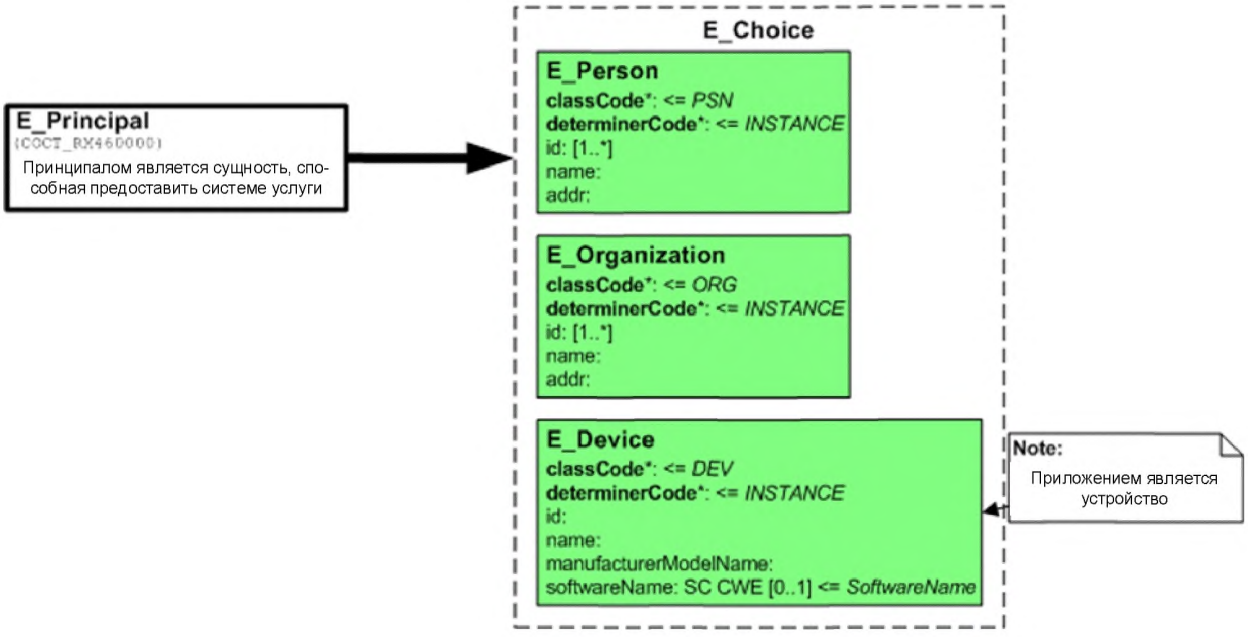


Рисунок G.5 — Пример точки входа типа CMET на диаграмме D-MIM

Диаграмма модели R-MIM

Одна точка входа в модели D-MIM может породить несколько диаграмм модели R-MIM. Каждая диаграмма модели R-MIM представляет семейство иерархических описаний сообщений HMD, описывающих конкретное множество повторно используемых понятий.

Диаграмма иерархического описания сообщения HMD

Из каждой диаграммы модели R-MIM можно произвести одну или несколько диаграмм иерархических описаний сообщений HMD, представляющих собой диаграммы экземпляров HMD. Диаграммы иерархических описаний сообщений HMD изображают идентификаторы точек входа, соответствующих HMD, конструируемые по шаблону «HDrrhh00».

Отметим, что диаграмма модели R-MIM, построенная для типа CMET, показывает весь набор ограничений, применяемых к произведенному иерархическому описанию сообщения HMD и к типу сообщения. В этом отношении она фактически является диаграммой экземпляров объектов иерархического описания сообщения HMD.

На приведенной ниже диаграмме показаны две диаграммы модели R-MIM или иерархического описания сообщения HMD, произведенные от общей точки входа. В целях описания конкретной функции каждая из этих диаграмм сокращает исходную диаграмму. Эти сокращения включают в себя исключения атрибутов (справа) и исключение одного элемента из пары предпочтений (в обоих случаях), позволяющие создать тип CMET для счета юридического или физического лица.

На первой диаграмме (рисунок G.6) показано, что объект R_ResponsibleParty выполняет роль агента, в которой исполняющими и контролирующими сущностями являются физическое лицо или организация.

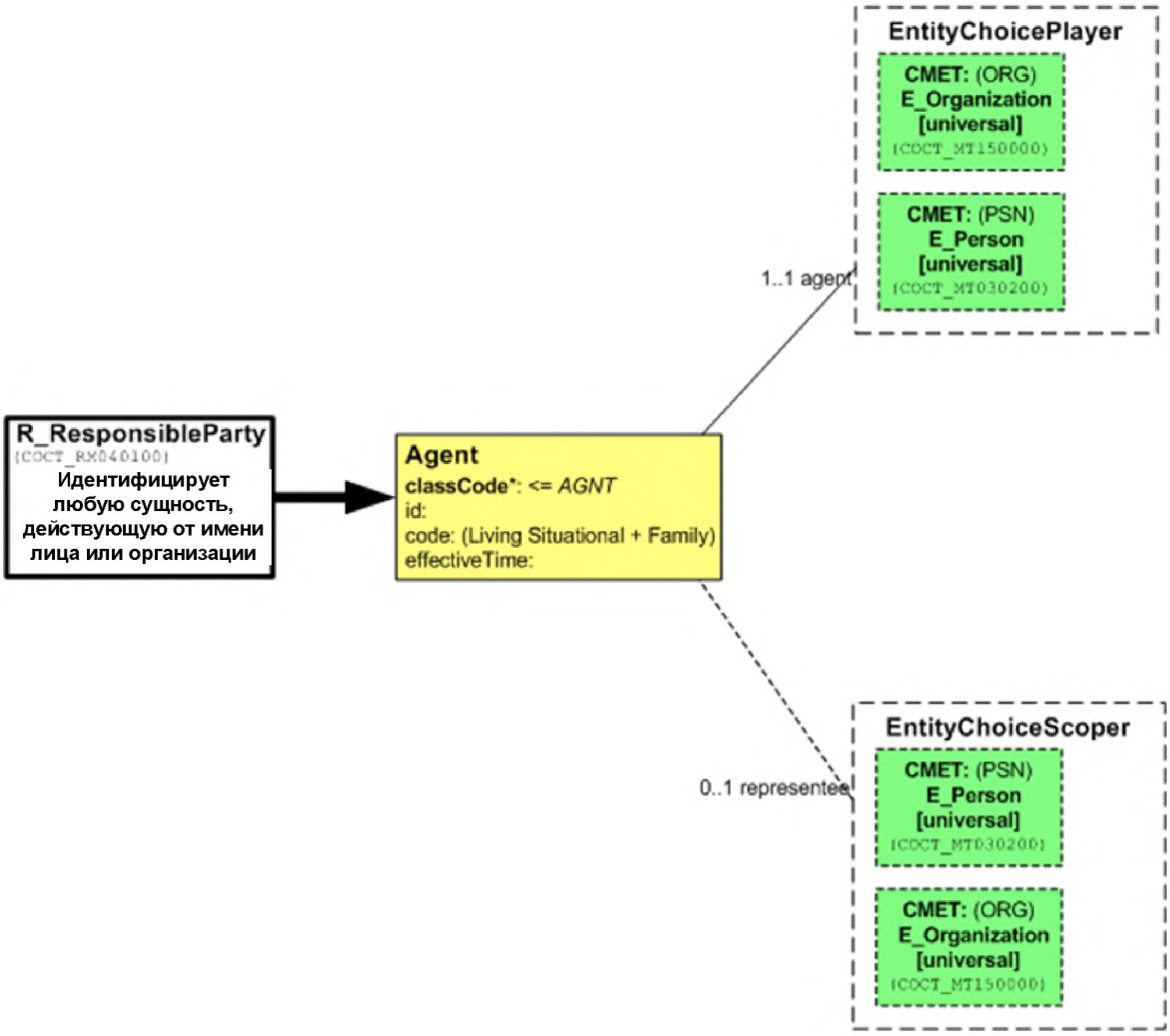


Рисунок G.6 — Первая диаграмма модели R-MIM

На второй диаграмме (рисунок G.7) показано, что исполняющей сущностью является устройство (device).

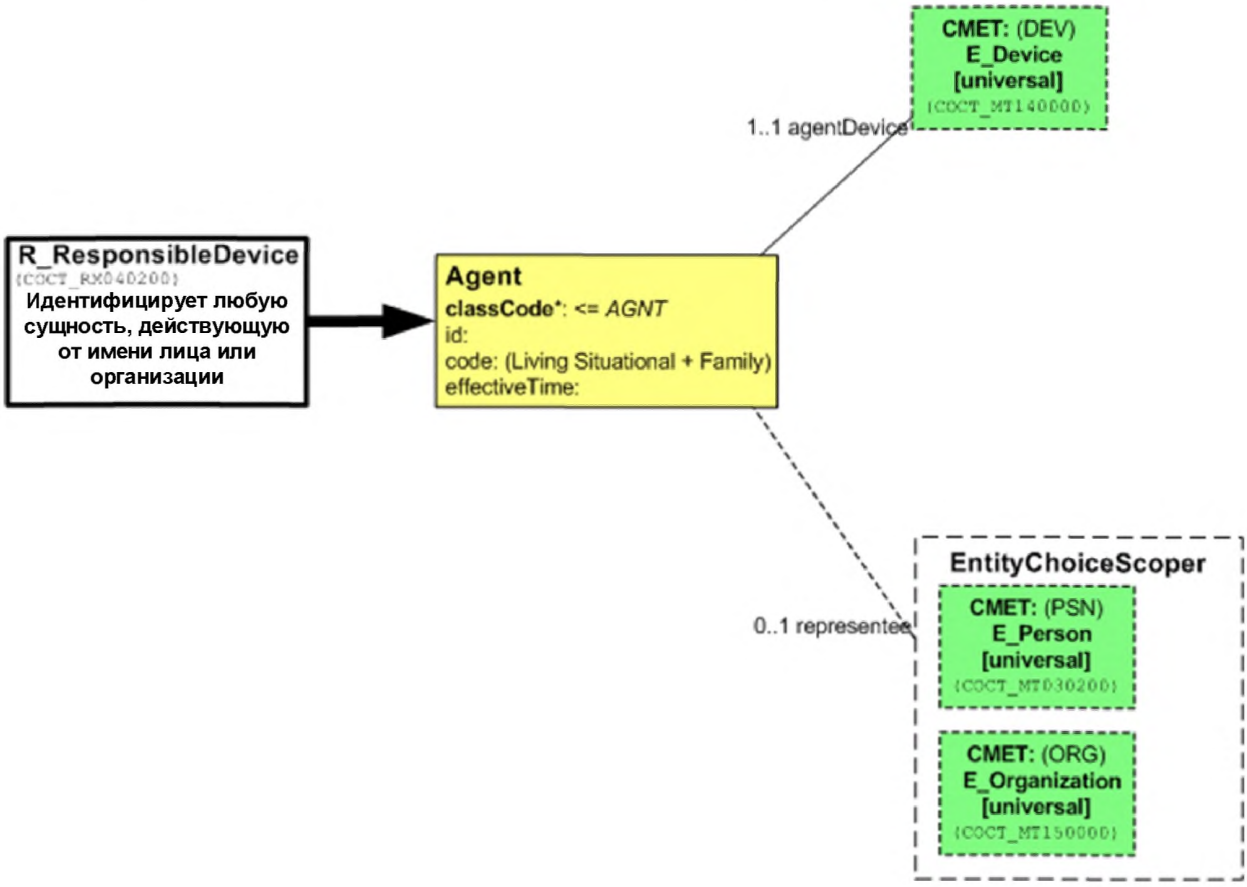


Рисунок G.7 — Вторая диаграмма модели R-MIM

Ссылка на тип CMET

Когда другая диаграмма ссылается на тип CMET или использует его непосредственно, то этот тип показан в специальной нотации, в виде прямоугольника с пунктирными сторонами, содержащем имя типа CMET, его идентификатор объекта, его код класса и уровень атрибутивности. Учтите, что также его выделяют цветом, соответствующим его корневому классу.

На рисунке G.8 представлена диаграмма, на которой показан один из производных типов CMET (HMD/MT), используемый в другой модели R-MIM.

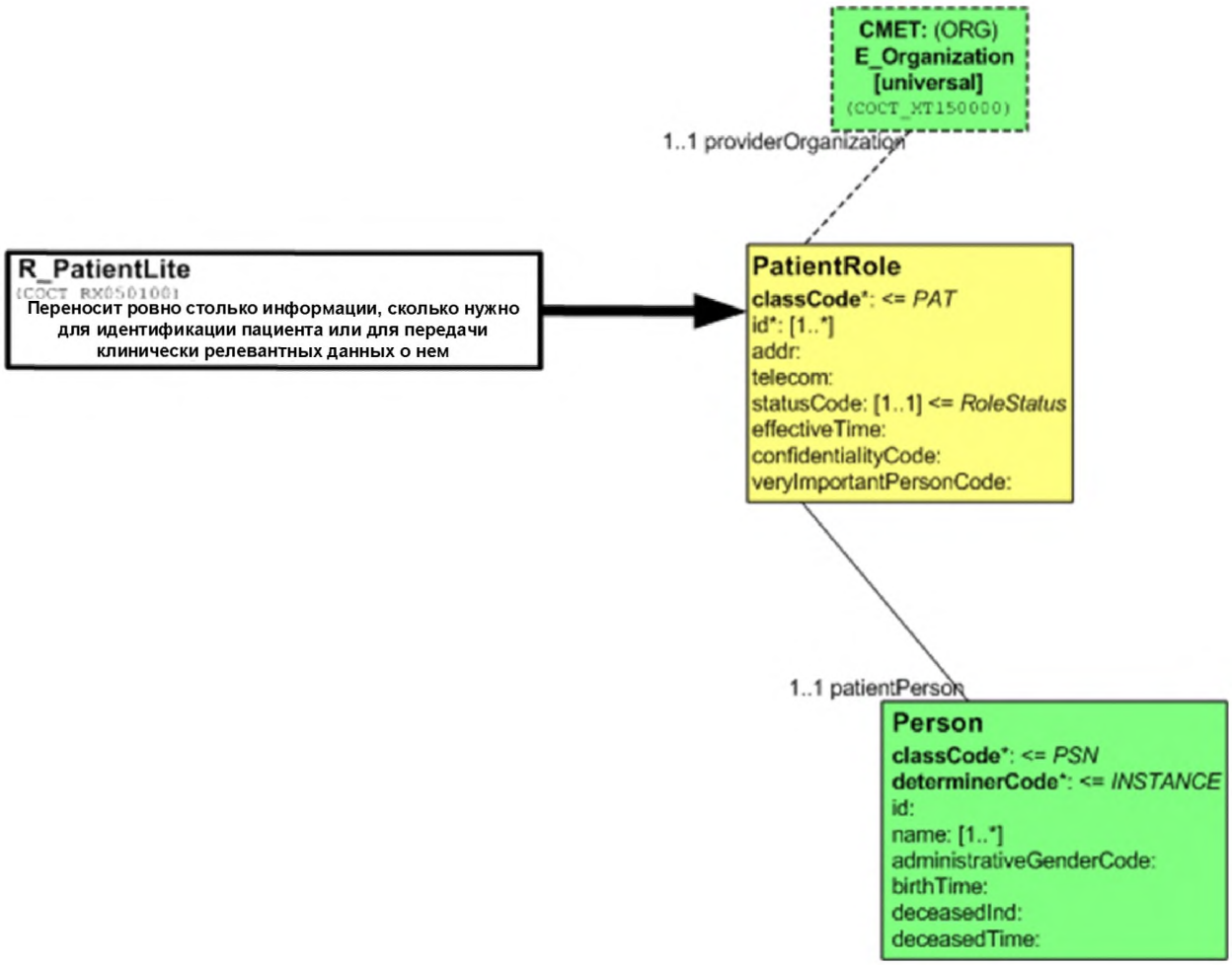


Рисунок G.8 — Использование типа CMET в другой модели R-MIM

Табличное представление иерархических описаний сообщений HMD

Тип CMET, описанный диаграммой уровня иерархического описания сообщения HMD, может быть полностью представлен в форме таблицы HMD. В таблице G.3 приведено простое табличное представление COCT_HD950100 типа CMET, показанного на рисунке G.8. Отметим, что это лишь фрагмент HMD.

Таблица G.3 — Фрагмент иерархического описания сообщения HMD COCT_HD950100

PRPA_HD101001 **New Person** (новое лицо)	Данное иерархическое описание HMD определяет сообщение о добавлении новой записи в регистр физических лиц. Данная модель R-MIM определяет сообщение о добавлении новой записи в регистр физических лиц. Произведена от модели R-MIM PRPA_RM101001
IdentifiedPerson (идентифицированное лицо)	Первичный ключ записи интересующего лица в регистре физических лиц
classCode [1..1] (M) Role(CS) {CNE:IDENT, fixed value="IDENT"}	Структурный атрибут; обозначает «идентифицированную» сущность
id[1..*] Role(SET<II>)	Один или несколько идентификаторов, присвоенных интересующему лицу в данном регистре физических лиц. Поскольку это обязательный атрибут, то нет смысла иметь в регистре идентифицированных лиц запись без идентификатора лица

Окончание таблицы G.3

addr[0..*] Role(BAG<AD>)	Почтовый адрес или адреса интересующего лица в контексте данного регистра физических лиц
telecom[0..*] Role(BAG<TEL>)	Телекоммуникационный адрес (адреса) для взаимодействия с интересующим лицом в контексте данного регистра физических лиц

G.2.6.5 Примеры

Примеры XML-схемы для типа CMET R_Patient можно найти в подразделе 2.7 Руководства по версии 3.

G.2.7 Уточнение

Уточнение представляет собой фундаментальный подход ко многим аспектам разработки стандарта HL7, версия 3. В настоящем руководстве описаны следующие ключевые области применения уточнения:

1 Уточнение модели RIM, приводящее к появлению моделей D-MIM и R-MIM.

2 Уточнение модели R-MIM для разработки иерархического описания сообщений HMD.

Третьим основным аспектом детализации являются ограничения, которые лежат в основе спецификации уточнения, ограничения и локализации. Ограничения видимости (appearance constraints) изменяют статус модели или элемента сообщения. Элементы (описанные как необязательные) можно сделать обязательными или исключить из сообщения. Ограничения кратности (cardinality constraints) изменяют минимальное и максимальное число повторений атрибута класса или ассоциации. Ограничения типов (type constraints) относятся к замене типа данных, определенного для атрибута, на более ограниченный тип данных. При этом могут применяться лишь специальные замены, описанные в спецификации ITS XML, или на более общем уровне в спецификации абстрактных типов данных. Данная задача может быть упрощена за счет использования инструментов на основе Visio, показывающие для каждого атрибута, какие ограничения могут быть для него применены. Совокупность ограничений сообщения называется профилем ограничений (constraint profile). В стандарте HL7, версия 3 профили ограничений могут быть получены из утвержденной спецификации сообщения версии 3 (V3 message specification). Соответствие (conformance) представляет собой процесс взаимодействия спонсоров информационных систем, например, производителей, с организацией пользователя для объяснения, что их продукты удовлетворяют спецификациям стандарта HL7. В последующих версиях стандарта версии 3 требования соответствия будут явно отнесены к релевантным ролям приложений, определенных в стандарте версии 3, применительно к конкретному сообщению. Этот процесс называется функциональным соответствием. Кроме того, техническое соответствие должно быть показано для описания сообщений версии 3, а также одного или нескольких профилей данного сообщения. Локализация (localization) представляет собой процесс, сочетающий ограничения и расширения существующих артефактов стандарта версии 3. Локализацию могут выполнять многие организации, но стандартом жестко определено, когда и при каких условиях она возможна. Более подробную информацию можно получить в подразделе «Локализация» раздела «Уточнение, ограничение и локализация».

G.2.8 Примеры

В данной подразделе приведен пример сообщения стандарта версии 2.4, а также эквивалентное ему сообщение версии 3, представленное с использованием спецификации XML ITS с аннотациями. Содержание подраздела основано на переработанной версии спецификации XML ITS. Однако генератор XML-схемы для данной версии спецификации ITS все еще не очень устойчив. Поэтому некоторые части приведенного ниже примера могут быть впоследствии изменены, поскольку их интерпретация в спецификации ITS может иметь некоторые изъяны. Комитет Publishing Committee продолжит вносить изменения в это подраздел с примерами и, возможно, выпустит к ней поправки в процессе утверждения данной спецификации. К тем местам подраздела, где могут возникать подобные проблемы, будут приведены комментарии.

Ниже показаны сообщение версии 2.4 ORU^R01 для передачи концентрации глюкозы в сыворотке крови и эквивалентное ему сообщение версии 3. Различные документы XML-схем не приведены. Некоторые важные преобразования и производное содержание указаны в комментариях к XML-сообщению стандарта версии 3.

G.2.8.1 Сообщение HL7 версии 2.4

```
MSH|^~\&|GHH LAB|ELAB-3|GHH OE|BLDG4|200202150930||ORU^R01
  |CNTRL-3456|P|2.4<cr>
PID|||555-44-4444||EVERYWOMAN^EVE^E^^^^L|JONES
  |196203520|F|||153 FERNWOOD DR.^^STATESVILLE^OH^35292||
  (206)3345232|(206)752-121||||
  AC555444444||67-A4335^OH^20030520<cr>
OBR|1|845439^GHH OE|1045813^GHH LAB|1554-5^GLUCOSE|||200202150730|||||||||
  555-55-5555^PRIMARY^PATRICIA P^^^^MD^^LEVEL SEVEN HEALTHCARE, INC.
  |||||||||F||||||444-44-4444^HIPPOCRATES^HOWARD H^^^^MD<cr>
OBX|1|SN|1554-5^GLUCOSE^POST 12H CFST:MCNC:PT:SER/PLAS:QN||^182|mg/dl|
  70_105|H|||F<cr>
```

G.2.8.2 Корневой элемент

Корневой элемент XML-экземпляра содержит информацию, необходимую для проверки правильности сообщения. Корневым элементом является Message, определенный схемой MCCI_MT000101.xsd. Текущая версия спецификации ITS (3-е издание) использует для всех экземпляров сообщений единственное пространство имен стандарта версии 3.

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!–Example copyright 2002 by Health Level Seven, Inc. -->
<Message xmlns="urn:hl7-org:v3"
  xmlns:xsi="http://www.w3.org/2002/XMLSchema-instance"
  xsi:schemaLocation="urn:hl7-org:v3/MCCI_MT000101
    MCCI_MT000101.xsd»>
```

Отправитель и получатель описаны в содержании корневого элемента в терминах приложений и организаций, осуществляющих обмен данными. Для обеспечения глобальной уникальности идентификаторов используются объектные идентификаторы (ОИД). В приведенном примере все ОИД одинаковые, а в реальном сообщении они должны быть уникальными.

Отметим, что в корневом элементе определяется уникальный идентификатор обмена сообщениями (message's interaction identifier), а данном примере, POLB_IN004410, идентифицирующий тип сообщения, инициирующее событие и роли приложения.

Элемент executedByRcvApp описывает приложение-получателя и организацию, элементы executedByRespondToOrg и executedBySendApp идентифицируют лицо, отправившее сообщений, приложение и организацию. Получателем является система GHH_OE, обменивающихся данными Bldg4. Отправителем служит система GHH_LAB, имеющая местонахождение E-LAB3.

Корневой элемент является оберткой полезного содержания, представляющего собой событие управления сообщением. Оно включено в элемент has_payload_ControlActEvent.

```
  ...
  <id root="2.16.840.1.113883.1122" extension="CNTRL-3456"/>
  <!– ИД сообщения, [msh.10] -->
  <creation_time value="2002-08-16T14:30:35.16-06:00"/>
  <!– [msh.7] -->
  <version_id>3.0</version_id>
  <interaction_id root="2.16.840.1.113883"
    extension="POLB_IN004410"/>
  <!—ИД взаимодействия = Observation Event Complete,
    уведомление (POLB_IN004410) источник=ORU^R01-->
        уведомление (POLB_IN004410) источник=ORU^R01-->
  <!– [msh.9] -->
  <processingCode code="P"/>
  <!– код обработки, [msh.11] -->
  <processingModeCode code="I"/>
  <!– код обработки, [msh.11] -->
  <acceptAckCode code="ER"/>
  <!– [msh.15] -->
  <!– только ошибки –>
  <applicationAckCode code="ER"/>
  <!– [msh.16] -->
   <communicationFunctionRsp>
    <!-- полагаем контакт отправителя = «respond_to» -->
    <type_cd code="RSP"/>
    <telecom use="WP" url="555-555-5555"/>
    <servedBy>
      <nm xsi:type="dt:PN">
```

```
        <dt:family>Гиппократ</dt:family>
        <dt:given>Гарольд</dt:given>
        <dt:given>X</dt:given>
        <dt:suffix qualifier="AC">MD</dt:suffix>
      </nm>
      <telecom use="WP" url="555-555-5555"/>
    </servedBy>
  </executedByRespondToOrg>
  <executedBySendApp>
    <type_cd code="SND"/>
    <telecom value=»127.127.127.255»/>
    <servedBy>
      <!-- приложение-отправитель, [msh.3] -->
      <id extension="GHH LAB" root="2.16.840.1.113883.1122"/>
      <nm use="L">
        <given>Имя компонента</given>
      </nm>
      <telecom value="555-555-2005" use="H"/>
      <agencyFor>
        <!—учреждение-отправитель [msh.4]  -->
        <representedOrganization>
          <id nullFlavor="OTH"/>
        </representedOrganization>
      </agencyFor>
      <presence>
        <location>
          <id root="2.16.840.1.113883.1122"
            extension="ELAB-3"/>
          <nm xsi:type="dt:TN">GHH Lab</nm>
        </location>
      </presence>
    </servedBy>
  </executedBySendApp>
  <executedByRcvApp>
    <type_cd code="RCV"/>
    <telecom value="127.127.127.0"/>
    <servedBy>
      <!-- Приложение-получатель, [msh.5] -->
      <id root=»2.16.840.1.113883.1122»
        extension="GHH OE"/>
      <nm use="L">
        <given>Имя компонента</given>
      </nm>
      <telecom value="555-555-2005" use="H"/>
      <agencyFor>
        <representedOrganization>
          <id root="2.16.840.1.113883.19.3.1001"/>
          <nm xsi:type="TN">Амбулаторная клиника GHH</nm>
        </representedOrganization>
```

```
      </agencyFor>
      <presence>
        <location>
          <id root="2.16.840.1.113883.1122"
            extension="BLDG4"/>
          <nm xsi:type="TN">Амбулаторная клиника GHH</nm>
        </location>
      </presence>
    </servedBy>
  </executedByRcvApp>
  <has_payload_ControlActEvent xsi:type="MCAI_HD700200">
    ...
  </has_payload_ControlActEvent>
</Message>
```

G.2.8.3 Управляющее действие

Управляющее действие сообщения является еще одной оберткой фактического сообщения. Она имеет атрибут class_cd, равный «CEVN», и указывает, что ответ на это сообщение не ожидается. В тегах элемента verifier указан врач-лаборант или иное лицо, ответственное за данный результат исследования, а элемент interactionTarget (связь действий) связывает эту обертку события управления с целевым сообщением типа POLB_MT004401.

```
<has_payload_ControlActEvent xsi:type="MCAI_HD700200">
  <!-- Обертка события управления взаимодействием сообщений,
    подставляется отправителем -->
  <!—событие управления действием  -->
  <response_cd code="N"/>
  <verifier>
    <participant_COCT_MT090100>
      <i
+d root="2.16.840.1.113883.1122"
        extension="444-444-4444"/>
    </participant_COCT_MT090100>
  </verifier>
    <interactionTarget xsi:type="POLB_MT004101">
    ...
  </interactionTarget>
</has_payload_ControlActEvent>
```

G.2.8.4 Тело сообщения

«Содержание предметной области» начинается собственным корневым элементом, а именно, interactionTarget. В его внутренних элементах указаны вид исследования, код статуса status_cd и результаты. Фактическое значение результата показано в элементе value. Значение элемента interpretation_cd показывает, что значение value интерпретировалось как «высокое» (H), в то время как в элементе referenceRange предоставлен диапазон нормы для данного конкретного исследования.

Части сообщения patient (пациент) и provider («ответственный поставщик») обсуждаются в последующих разделах, а вся остальная информация приведена ниже.

```
<interactionTarget xsi:type="POLB_MT004101">
  <ObservationEvent>
    <!--- Идентификатор — номер заказа у исполнителя ... -->
    <id root="2.16.840.1.113883.1122"
      extension="1045813"
      assigningAuthorityName="GHH LAB"/>
```

```
    <cd code="1554-5" codeSystemName="LN"
      displayName="GLUCOSE^POST 12H
        CFST:MCNC:PT:SER/PLAS:QN"/>
    <status_cd code="completed"/>
    <!-- время взятия образца -->
    <effective_time>
      <dt:center value="2002-02-15T07:30:00"/>
    </effective_time>
    <!-- время фактического выполнения лабораторного анализа -->
    <activity_time>
      <dt:center value="2002-02-15T08:30:00"/>
    </activity_time>
    <priority_cd code="R"/>
    <value xsi:type="dt:PQ" value="182" unit="mg/dL"/>
    <interpretation_cd code="H"/>
    <participant>
      ...
    </participant>
    <patient>
      <!-- PID -->
      <patient>
        ...
      </patient>
    </patient>
    <inFulfillmentOf>
      <!-- placer order -->
      <priorObservationOrder>
        ...
      </priorObservationOrder>
    </inFulfillmentOf>
    <referenceRange>
      <referenceObservationEventCriterion>
        <value xsi:type="dt:IVL_PQ">
          <dt:low value="70" unit="mg/dL"/>
          <dt:high value="105" unit="mg/dL"/>
        </value>
      </referenceObservationEventCriterion>
    </referenceRange>
  </ObservationEvent>
</interactionTarget>
```

G.2.8.5 Исходный заказ

Ссылка на исходный заказ, по которому было выполнено данное лабораторное исследование, указано в отношении действий типа FLFS, имеющим имя priorObservationOrder, которое идентифицирует заказ номером, присвоенным заказчиком. Этот номер должен использоваться получателем для соотнесения результатов с заказом. В сообщение включены идентификатор, имя и фамилия заказавшего врача (д-р Patricia Primery). Отметим, что здесь присутствуют два уровня представления информации: уровень врача и уровень физического лица. Идентификатор относится к уровню врача, а имя и фамилия — к уровню физического лица.

```
<inFulfillmentOf>
  <!-- идентификация заказа у заказчика -->
  <priorObservationOrder>
```

```
    <id root="2.16.840.1.113883.1122"
      extension="845439"
      assigningAuthorityName="GHH OE"/>
    <cd code="1554-5" codeSystemName="LN"
      displayName="Глюкоза в сыворотке крови"/>
    <participant>
      <type_cd code="RESP"/>
      <assignedEntity>
        <id root="2.16.840.1.113883.1122"
          extension="555-555-5555"/>
        <assignee_Person>
          <nm use="L" xsi:type="dt:PN">
            <dt:family>Primary</dt:family>
            <dt:given>Patricia</dt:given>
            <dt:given>P</dt:given>
            <dt:suffix qualifier="AC">
              MD</dt:suffix>
          </nm>
        </assignee_Person>
      </assignedEntity>
    </participant>
  </priorObservationOrder>
</inFulfillmentOf>
```

G.2.8.6 Врач, выполнивший заказ

В сообщение включены идентификатор, имя и фамилия врача, выполнившего заказ (Mr. Harold H Hippocrates). Как и в случае заказавшего врача, здесь присутствуют два уровня представления информации: уровень врача и уровень физического лица. Идентификатор относится к уровню врача, а имя и фамилия — к уровню физического лица.

```
<participant>
  <!-- врач, подписавший результат -->
  <type_cd code=»AUT»/>
  <!-- время подписи -->
  <time value=»2002-08-16T09:30:00»/>
  <mode_cd code=»WRITTEN»/>
  <!-- бумажный документ подписан -->
  <signature_cd code="S"/>
  <assignedEntity>
    <id root="2.16.840.1.113883.1122"
      extension="444-444-4444"/>
    <assignee_Person>
      <nm use="L" xsi:type="dt:PN">
        <dt:family>Hippocrates</dt:family>
        <dt:given>Harold</dt:given>
        <dt:given>H</dt:given>
        <dt:suffix qualifier="AC">MD</dt:suffix>
      </nm>
    </assignee_Person>
  </assignedEntity>
</participant>
```

G.2.8.7 Пациент

Пациент, Eve E. Everywoman, также представлен двумя уровнями — уровнем пациента и уровнем физического лица. Внутри элемента patient (пациент) находится немного элементов, поскольку он является частью сильно связанного сообщения (например, не предусмотрен элемент адреса). С другой стороны, необходимо указывать идентификатор пациента id, а также имя и фамилию, что позволяет обеспечить хотя бы простейшую форму проверки ошибок. Для каждого уровня существует свой идентификатор — заранее присвоенный идентификатор пациента на уровне пациента и номер водительских прав на уровне физического лица.

```
<patient>
  <patient>
    <id root="2.16.840.1.113883.1122"
      extension="375913"/>
    <patient_Person>
      <!-- Водительские права в штате Огайо -->
      <pat:id root="2.16.840.1.113883.1122"
        extension="444-22-2222"
        validTime="-2003-05-20"
        assigningAuthorityName="OH"/>
      <pat:nm use="L" xsi:type="PN">
        <dt:family>Everywoman</dt:family>
        <dt:given>Eve</dt:given>
        <dt:given>E</dt:given>
      </pat:nm>
    </patient_Person>
  </patient>
</patient>
```

G.2.8.8 Пример полной версия сообщения observation.xml

```
<Message xmlns="urn:hl7-org:v3"
  xmlns:xsi="http://www.w3.org/2002/XMLSchema-instance"
  xsi:schemaLocation="urn:hl7-org:v3/MCCI_MT000101
    MCCI_MT000101.xsd">
  <id root="2.16.840.1.113883.1122" extension="CNTRL-3456"/>
  <!-- ИД сообщения, [msh.10] -->
  <creation_time value="2002-08-16T14:30:35.16-06:00"/>
  <!-- [msh.7] -->
  <version_id>3.0</version_id>
  <interaction_id root="2.16.840.1.113883"
    extension="POLB_IN004410"/>
  <!—ИД взаимодействия = Observation Event Complete,
    уведомление (POLB_IN004410) источник=ORU^R01-->
        уведомление (POLB_IN004410) источник=ORU^R01-->
  <processing_cd code="P"/>
  <!-- код обработки, [msh.11] -->
  <accept_ack_cd code="ER"/>
  <!-- только ошибки -->
  <application_ack_cd code="ER"/>
  <executedByRespondToOrg>
    <!-- полагаем контакт отправителя = “respond_to” -->
    <type_cd code="RSP"/>
```

```
    <telecom use="WP" url="555-555-5555"/>
    <servedBy>
      <nm xsi:type="dt:PN">
        <dt:family>Hippocrates</dt:family>
        <dt:given>Harold</dt:given>
        <dt:given>H</dt:given>
        <dt:suffix qualifier="AC">MD</dt:suffix>
      </nm>
      <telecom use="WP" url="555-555-5555"/>
    </servedBy>
  </executedByRespondToOrg>
  <executedBySendApp>
    <type_cd code="SND"/>
    <telecom value=»127.127.127.255»/>
    <servedBy>
      <!-- приложение-отправитель, [msh.3] -->
      <id extension="GHH LAB" root="2.16.840.1.113883.1122"/>
      <nm use="L">
        <given>Имя компонента</given>
      </nm>
      <telecom value="555-555-2005" use="H"/>
      <agencyFor>
        <!—учреждение-отправитель [msh.4]  -->
        <representedOrganization>
          <id nullFlavor="OTH"/>
        </representedOrganization>
      </agencyFor>
      <presence>
        <location>
          <id root="2.16.840.1.113883.1122"
            extension="ELAB-3"/>
          <nm xsi:type="dt:TN">GHH Lab</nm>
        </location>
      </presence>
    </servedBy>
  </executedBySendApp>
  <executedByRcvApp>
    <type_cd code="RCV"/>
    <telecom value="127.127.127.0"/>
    <servedBy>
      <!-- Приложение-получатель, [msh.5] -->
      <id root=»2.16.840.1.113883.1122»
        extension="GHH OE"/>
      <nm use="L">
        <given>Имя компонента</given>
      </nm>
      <telecom value="555-555-2005" use="H"/>
      <agencyFor>
        <representedOrganization>
```

```
        <id root="2.16.840.1.113883.19.3.1001"/>
        <nm xsi:type="TN">Амбулаторная клиника GHH</nm>
      </representedOrganization>
    </agencyFor>
    <presence>
      <location>
        <id root="2.16.840.1.113883.1122"
          extension="BLDG4"/>
        <nm xsi:type="TN">Амбулаторная клиника GHH</nm>
      </location>
    </presence>
  </servedBy>
</executedByRcvApp>
<has_payload_ControlActEvent xsi:type="MCAI_HD700200">
<!--  Обертка события управления взаимодействием сообщений,
  подставляется отправителем -->
  <response_cd code=»N»/>
  <verifier>
    <participant_COCT_MT090100>
      <id root="2.16.840.1.113883.1122"
        extension="444-444-4444"/>
    </participant_COCT_MT090100>
  </verifier>
  <interactionTarget xsi:type="POLB_MT004101">
    <ObservationEvent>
  <!-- Идентификатор — номер заказа у исполнителя -->
      <id root="2.16.840.1.113883.1122"
        extension="1045813"
        assigningAuthorityName="GHH LAB"/>
      <cd code="1554-5" codeSystemName="LN"
        displayName="GLUCOSE^POST 12H
          CFST:MCNC:PT:SER/PLAS:QN"/>
      <status_cd code="completed"/>
      <!-- время взятия образца -->
      <effective_time>
        <dt:center value="2002-02-15T07:30:00"/>
      </effective_time>
      <!-- время фактического выполнения лабораторного анализа -->
      <activity_time>
        <dt:center value="2002-02-15T08:30:00"/>
      </activity_time>
      <priority_cd code="R"/>
      <value xsi:type="dt:PQ" value="182" unit="mg/dL"/>
      <interpretation_cd code="H"/>
      <participant>
        <!-- врач, подписавший результат -->
        <type_cd code=»AUT»/>
        <!-- время подписи -->
        <time value=»2002-08-16T09:30:00»/>
```

```
    <mode_cd code=»WRITTEN»/>
    <!-- бумажный документ подписан -->
    <signature_cd code="S"/>
    <assignedEntity>
      <id root="2.16.840.1.113883.1122"
        extension="444-444-4444"/>
      <assignee_Person>
        <nm use="L" xsi:type="dt:PN">
          <dt:family>Hippocrates</dt:family>
          <dt:given>Harold</dt:given>
          <dt:given>H</dt:given>
          <dt:suffix qualifier="AC">MD</dt:suffix>
        </nm>
      </assignee_Person>
    </assignedEntity>
  </participant>
  <patient>
    <!-- Идентификатор пациента -->
    <patient>
      <id root="2.16.840.1.113883.1122"
        extension=»375913»/>
      <patient_Person>
        <!-- Водительские права в штате Огайо -->
        <pat:id root="2.16.840.1.113883.1122"
          extension="444-22-2222"
          validTime="-2003-05-20"
          assigningAuthorityName="OH"/>
        <pat:nm use="L" xsi:type="PN">
          <dt:family>Everywoman</dt:family>
          <dt:given>Eve</dt:given>
          <dt:given>E</dt:given>
        </pat:nm>
      </patient_Person>
    </patient>
  </patient>
  <inFulfillmentOf>
    <!-- идентификация заказа у заказчика -->
    <priorObservationOrder>
      <id root="2.16.840.1.113883.1122"
        extension="845439"
        assigningAuthorityName="GHH OE"/>
      <cd code="1554-5" codeSystemName="LN"
        displayName="Глюкоза в сыворотке крови"/>
      <participant>
        <type_cd code="RESP"/>
        <assignedEntity>
          <id root="2.16.840.1.113883.1122"
            extension="555-555-5555"/>
          <assignee_Person>
```

```
                <nm use="L" xsi:type="dt:PN">
                  <dt:family>Primary</dt:family>
                  <dt:given>Patricia</dt:given>
                  <dt:given>P</dt:given>
                  <dt:suffix qualifier="AC">
                    MD</dt:suffix>
                </nm>
              </assignee_Person>
            </assignedEntity>
          </participant>
        </priorObservationOrder>
      </inFulfillmentOf>
      <referenceRange>
        <referenceObservationEventCriterion>
          <value xsi:type="dt:IVL_PQ">
            <dt:low value="70" unit="mg/dL"/>
            <dt:high value="105" unit="mg/dL"/>
          </value>
        </referenceObservationEventCriterion>
      </referenceRange>
    </ObservationEvent>
  </interactionTarget>
 </has_payload_ControlActEvent>
</Message>
```

G.3 Компоненты сообщений HL7

G.3.1 Навигация по спецификации сообщений

Данный раздел Руководства помогает читателям стандарта HL7, версия 3, разобраться в структуре сообщений и типах содержания, представленных техническими комитетами. Эта информация приводится в описаниях каждой главы, посвященной отдельной предметной области.

Стандарт версии 3 предлагает новый подход к разработке структуры сообщений. В целях достижения понимания нового подхода в главах предметных областей раздела сначала приводится высокоуровневый обзор, а затем рассматриваются все более конкретные детали описания сообщений (включая сценарии, роли приложений, инициирующие события, модели D-MIM, модели R-MIM, иерархические описания сообщений HMD, типы сообщений и взаимодействия). Ниже приведена структура оглавления каждой предметной области:

Предисловие
i Обращение к читателям
ii Благодарности
iii Изменения по сравнению с предыдущим изданием
iv Известные проблемы и планируемые изменения
1 Обзор
1.1 Введение и область применения
1.2 Навигация по элементам конструкции сообщений
1.3 Информационные модели DMIM
1.4 Сценарии
2 Раздел человеческих ресурсов
2.1 Сценарии
2.2 Роли приложений
2.3 Инициирующие события
2.4 Модели R-MIM
2.5 Иерархические описания сообщений
2.6 Взаимодействия

Каждая глава начинается с обзора предметной области и области применения описанных в ней сообщений. Поскольку «общая картина» понимания отдельной предметной области критична для понимания определяемых сообщений, каждая глава предметной области содержит раздел «Сценарии». Этот раздел содержит текстовые

описания передаваемых сообщений и соответствующие им диаграммы, а также описания типов приложений сферы здравоохранения, взаимодействующих с помощью сообщений, описанных в этой главе. Кроме того, в нем иллюстрируется общий поток сообщений, которым обмениваются эти приложения, включая порядок сообщений в этом потоке.

В описанном ниже разделе навигации по элементам конструкции сообщений приводится краткий обзор типов сообщений, а также отношения между иерархическими описаниями сообщений HMD и моделями R-MIM, включая гиперссылки на фактические описания приведенных в главе артефактов. При навигации используется представление «в форме дерева», позволяющее пользователю развертывать или скрывать иерархию элементов. Если левым символом строки является знак «+», то при выборе данной строки отображаются все ее дочерние строки. Если же левым символом является знак «–», то при выборе этой строки указанные под ней зависимые, дочерние строки скрываются. Если левым символом строки является точка, то у этой строки нет дочерних строк, которые можно было бы показать.

Содержание описания каждой предметной области, измененного после января 2004 г., было приведено в соответствие со следующей структурой. Вначале приводится информация, применимая ко всей предметной области. Она описана в последующих двух абзацах. За ней приводятся описания сообщений, сгруппированные по «тематическим разделам». Фактически каждый такой раздел охватывает все определения в конкретной подобласти. Названия тем и их количество определяется соответствующим комитетом Рабочей группы HL7.

В самом начале описания каждой предметной области представлена ее информационная модель (D-MIM) в форме графического и текстового описания границ предметной области. Модель D-MIM является первой из информационных моделей, представленных в каждой главе предметной области. В модели D-MIM приводится графическое представление только тех классов, атрибутов и ассоциаций, которые требуются для сообщений, описанных для данной предметной области. Модель D-MIM является подмножеством модели RIM, но использует специальные обозначения для представления таких объектов, как типы CMET и различные виды ассоциаций. Как только классы, атрибуты и ассоциации конкретной предметной области выделены с помощью модели D-MIM, следующим шагом является извлечение из модели D-MIM совокупностей классов, атрибутов и ассоциаций, требуемых для иерархического описания сообщения HMD или нескольких HMD, производимых из одного и того же корневого класса. Результаты извлечения называются моделями R-MIM (Refined Message Information Models — уточненные информационные модели сообщений). Они, являются подмножеством модели D-MIM и следуют тем же соглашениям, что и модель D-MIM предметной области.

Вторым элементом предметной области является один или несколько сценариев уровня предметной области. Каждый сценарий включает в себя графическое представление, называемое диаграммой взаимодействия. На данной диаграмме приложения, предназначенные для сферы здравоохранения, представлены в виде исполняемых ими ролей, например, «Хранилище клинических документов» или «Заказчик». Эти общие роли известны как роли приложения и изображаются на диаграмме вертикальными линиями. Каждый поток информации между двумя ролями приложений представляет собой взаимодействие и изображается на диаграмме сценариев в виде горизонтальной линии со стрелкой. Взаимодействия описывают назначение потока информации. Примерами взаимодействий, описанных в стандарте HL7, версия 3, могут служить транзакции Create Patient Billing Account (создать лицевой счет пациента) и Delete Patient Billing Account (удалить лицевой счет пациента). После представления общей картины обмена данными в разделе «Сценарии» приводится текстовое описание того, как эти взаимодействия могут происходить в реальной жизни. Затем приводится описание каждой роли приложения. Сценарии могут быть представлены как для всей предметной области, так и для отдельной темы, выделенной в данной предметной области.

Весь оставшийся материал сгруппирован по именам тем. Некоторые комитеты определяют большое количество имен тем, другие — лишь несколько. Решение о выборе количества имен тем зависит только от комитета. В содержании каждой темы спецификация сообщения группируется под шестью заголовками. Каждый из них описан в следующих абзацах.

Сценарий. Для каждой темы приведен один или несколько сценариев, покрывающих следующие за ними спецификации.

Роли приложений. Приводятся роли приложений, применимые к данной теме. Из описания ролей приложения читатель может понять назначение потоков информации между двумя приложениями, используемыми в сфере здравоохранения, а также узнать роли, выполняемые этими приложениями в данном обмене.

Инициирующие события. Дают ответ на вопрос, что именно служит инициатором информационного обмена между двумя приложениями. В данной части спецификации тематики приводится список (описание) инициирующих событий, описанных для данной предметной области.

Конечный результат, типы сообщений, производится путем пересечения специфицированных взаимодействий, ролей приложений и инициирующих событий. В заключительных частях каждой главы предметной области с помощью ряда информационных моделей описывается процесс получения результирующих типов сообщений.

Данные, представленные в модели R-MIM, упорядочены в табличном формате, известном как Иерархическое описание сообщения HMD (Hierarchical Message Description). Каждое описание HMD создает один базовый шаблон сообщения, по которому конструируются конкретные типы сообщений. Тип сообщения представляет уникальный набор ограничений, применяемых к конкретному иерархическому описанию сообщения HMD. Типы сообщений представляются как в форме сетки, так и в табличной форме, а также в виде листов Excel. Кроме того,

наряду с примером XML-документа приводится ссылка на XML-схему, используемую для проверки соответствия сообщения данному типу. Каждая из глав предметной области завершается разделом, группирующим данные сообщений по типу взаимодействия, ролям приложений-отправителей и получателей, инициирующим событиям и типам сообщений.

В заключительной части спецификации предметной области приводится указатель типов общих элементов сообщений CMET, произведенных от модели D-MIM предметной области, а также типов CMET, упоминаемых в каждом иерархическом описании сообщения HMD, определенном в модели D-MIM. Этот указатель содержит ссылки на содержание предметной области типов общих элементов сообщений.

Для удобного перехода между документами используется указатель взаимодействий. В этом указатели перечислены все взаимодействия, используемые каждой ролью приложения, инициирующим событием и типом сообщения.

Глоссарий предметной области содержит как термины, определенные для данной предметной области, так и термины, общие для всех предметных областей.

G.3.1.1 Система идентификации артефактов

В стандартах HL7, версия 3 компоненты, образующие документацию, называются «артефактами». К ним относятся сценарии, роли приложений, инициирующие события, модели D-MIM, модели R-MIM, иерархические описания сообщений HMD, типы сообщений и взаимодействия. Каждый артефакт представляется Техническим комитетом и получает уникальный идентификатор, представляющий собой сочетание кодов подраздела, предметной области и артефакта с шестизначным числом, не имеющим смысловой нагрузки.

Коды подразделов и предметных областей приведены в таблице G.4.

Таблица G.4 — Коды подразделов и предметных областей

Предметные области информации о здоровье и клинической информации
Подраздел: Операции (PO)
Предметная область: Лаборатория (POLB)
Предметная область: Лекарственные назначения (PORX)
Подраздел: Медицинские карты (RC)
Предметная область: Электронные медицинские карты (RCMR)
Предметные области административной информации
Подраздел: Оказание медицинской помощи (PR)
Предметная область: Учет движения пациентов (PRPA)
Предметная область: Планирование приема пациентов (PRSC)
Предметная область: Управление персоналом (PRPM)
Подраздел: Финансовый (FI)
Предметная область: Требование оплаты и возмещение затрат (FICR)
Предметная область: Бухгалтерия и счета на оплату лечения (FIAB)
Спецификация инфраструктуры
Подраздел: Управление сообщениями (MC)
Предметная область: Инфраструктура управления сообщениями (MCCI)
Предметная область: Инфраструктура действий сообщений (MCAI)
Подраздел: Нормативно-справочная информация (MF)
Предметная область: Инфраструктура управления нормативно-справочными файлами (MFMI)
Подраздел: Запрос (QU)
Предметная область: Инфраструктура запросов (QUQI)
Подраздел: Общее содержание (CO)
Предметная область: Общие элементы сообщений (COCT)
Предметная область: Общее содержание сообщений (COMT)

Предметные области информации о здоровье и клинической информации, а также предметные области административной информации могут содержать специфическое содержание, относящееся к подразделам запросов и нормативно-справочных файлов, а также информацию, общую для всех предметных областей.

Артефактам присваиваются коды, перечисленные в таблице G.5.

Таблица G.5 — Коды артефактов

Артефакт	Код
Роль приложения	AR
D-MIM (доменная информационная модель)	DM
HMD (иерархическое описание сообщения)	HD
Взаимодействие	IN
Тип сообщения	MT
R-MIM (уточненная информационная модель сообщения)	RM
Сценарий	ST
Текст сценария	SN
Инициирующее событие	TE

Текущая версия стандарта HL7 3x поддерживает только одну сферу действия, идентифицируемую как «Универсальная сфера» (код «UV»).

Присвоение номера версии осуществляется, как только артефакт успешно прошел голосование на уровне членов. В настоящее время всем артефактам присвоена версия 00, указывающая, что они подлежат голосованию.

Роль приложения, представленная в предметной области «Оказание медицинской помощи», имеет следующий уникальный идентификатор объекта:

PRPA_AR00001UV00

где PR = подраздел «Оказание медицинской помощи», PA = предметная область «Учет движения пациентов», AR = артефакт «Роль приложения», 00001 = 6-значное незначащее число, присвоенное техническим комитетом для обеспечения уникальности, UV = сфера действия (единственным текущим значением является «UV» — универсальная сфера), 00 = текущий номер версии.

После изучения документов, поставленных на голосование, вы заметите, что большинство идентификаторов артефактов являются гиперссылками, связанными с соответствующими артефактами.

Остальные подразделы данного раздела руководства разъясняют каждый из этих артефактов.

G.3.2 Описание артефактов сообщений версии 3

Точное описание артефактов Версии 3 достигается с помощью определения для каждого артефакта «структурированного сортируемого имени». Такое уникальное структурированное имя обычно используется для идентификации связей данного артефакта с другими артефактами, позволяющих правильно упорядочить его в базе публикаций, являющейся как хранилищем артефактов, так и основой для логически упорядоченного представления спецификаций и голосуемых материалов.

Некоторые типы артефактов имеют также «Имя заголовка», которое является «дружественным для пользователей», отображается в оглавлении и используется в качестве заголовка артефакта. Заголовок не обязательно уникален, но, тем не менее, может служить удобным средством распознавания объекта человеком. В том случае, если для артефакта определен заголовок, то структурированное сортируемое имя отображается под заголовком для ссылки. Если же заголовок для артефакта не определен, то в оглавлении и строке заголовка отображается структурированное сортируемое имя.

Отметим, что в отличие от заголовка и структурированного сортируемого имени код артефакта является лучшим ключом поиска артефакта, поскольку он не меняется со временем и гарантированно уникален.

В таблице G.6 приведен алгоритм формирования структурированного сортируемого имени для каждого типа артефактов.

Таблица G.6 — Алгоритм формирования структурированного сортируемого имени

Объект	Алгоритм формирования структурированного сортируемого имени
Сценарий	[Базовый класс] [Свободный текст]
Текст сценария	[Базовый класс] [Свободный текст]
Роль приложения — не скрытая	[Базовый класс] [Свободный текст]
Роль приложения — скрытая	[Базовый класс] [Наклонение] [Способность] [Стереотип] [Окружение]
Роль приложения — запрос	Запрос [Базовый класс] [Наклонение] [Квалификатор] Инициатор запроса или [Базовый класс] [Наклонение] [Квалификатор] Исполнитель запроса
Инициирующее событие	[Базовый класс] [Наклонение] [Переход состояния] [Квалификатор] [Действие]
Инициирующее событие — запрос	[Базовый класс] [Наклонение] [Квалификатор] Запрос или [Базовый класс] [Наклонение] [Квалификатор] Ответ на запрос
Модель D-MIM	[Имя предметной области] Модель предметной области
Модель R-MIM	[Базовый класс] [Наклонение]
Иерархическое описание сообщения HMD	[Базовый класс] [Наклонение] [Переход состояния]
Тип сообщения	[Базовый класс] [Наклонение] [Переход состояния] [Окружение]
Взаимодействие	[Базовый класс] [Наклонение] [Переход состояния] [Действие] [Окружение]
Взаимодействие — запрос	[Базовый класс] [Наклонение] [Квалификаторы] Запрос или [Базовый класс] [Наклонение] [Квалификаторы] Ответ на запрос

Ниже описаны допустимые значения и порядок сортировки каждого компонента структурированного сортируемого имени.

Базовый класс (Base Class)

Определение: задается техническим комитетом в качестве первичного ключа сортировки для классификации содержание предметной области.

Порядок сортировки: задается техническими комитетами в зависимости от предметной области.

Свободный текст (Free Text)

Определение: структурированное сортируемое имя, присваиваемое некоторым артефактам, является обязательным только для артефактов, связанным с переходами состояний, например, для инициирующих событий. С помощью компонента «Свободный текст» любой технический комитет может присвоить артефакту произвольное текстовое имя. Однако и в этом случае структурированное сортируемое имя должно быть уникальным. В настоящее время некоторые технические комитеты не следуют этим правилам при описании инициирующих событий (что требует наличия всех компонентов, кроме какого-то одного). В результате структурированные сортируемые имена инициирующих событий не настолько удобочитаемы, как могло бы быть.

Порядок сортировки: упорядочивание по алфавиту.

Квалификаторы (Qualifiers)

Определение: уникально определяют запрос и основывается на параметрах запроса и структуре ответа на запрос.

Порядок сортировки: упорядочивание по алфавиту.

Окружение (Environment)

Определение: в зависимости от предметной области, техническим комитетом может быть задано любое значение компонента Environment, подходящее данной предметной области.

Порядок сортировки: упорядочивание по алфавиту.

Наклонение (Mood)

Определение: данный элемент является дополнительной характеристикой базового класса, используемой для базового класса Act (действие). Он ссылается на значение атрибута mood (наклонение) базового класса (см. описание модели RIM). Основными значениями являются Event (событие), Order (заказ), Intent (намерение) и Proposal (предложение).

Таблица G.7 — Последовательность сортировки и допустимые значения наклонения

Последовательность сортировки и допустимые значения
1 Proposal (предложение)
2 Order (заказ)
3 Intent (намерение)
4 Event (событие)

Действие (Action)

Определение. действие является квалификатором взаимодействий. Отражает специфичный тип коммуникации, осуществляемой при конкретном взаимодействии.

Таблица G.8 — Последовательность сортировки и допустимые значения квалификаторов взаимодействий

Последовательность сортировки и допустимые значения
1 Notification (уведомление)
2 Fulfilment Request (требование выполнения)
3 Confirmation (подтверждение)
4 Rejection (отклонение)

Стереотипы ролей приложения (Application Role Stereotypes)

Определение: термин «стереотип», широко используемый в UML-приложениях, в стандарте версии 3 используется пока что лишь в ролях приложений. В основном стереотипы используются для обозначения связанной группы элементов, относящихся к одной категории. В настоящее время выделены шесть стереотипов ролей приложений, основанных на способности одного приложения взаимодействовать с другим. На базовом уровне приложения просто отправляют и получают уведомления о возникновении событий (стереотипы Informer и Tracker). На следующем уровне приложения ожидают действий и ответа от приложения, получившего уведомление (стереотипы Placer и Fulfiller). На верхнем уровне приложения способны обрабатывать подтверждения запросов (стереотипы Confirmer и Confirmation Receiver). Использование этих стандартизированных имен стереотипов предназначено в основном для содействия в описании и использовании ролей приложений. Стереотипы не формируют исключительный набор имен ролей приложений.

Таблица G.9 — Последовательность сортировки и допустимые значения стереотипов ролей приложений

Последовательность сортировки и допустимые значения
1 Placer (инициатор): приложение, способное уведомить другое приложение о возникновении важного события, и ожидающее от получателя ответных действий.
2 Fulfiller (исполнитель): приложение, способное получать запрос от приложения со стереотипом Placer.
3 Confirmer (инициатор подтверждения): приложение, способное подтвердить запрос, полученный от приложения со стереотипом Fulfiller
4 Confirmation Receiver (получатель подтверждения): роль, реализуемая инициатором, указывающая, какие типы подтверждений она принимает.
5 Informer (информатор): приложение, способное уведомить другое приложение о возникновении важного события (изменение состояния существенного класса), но не ожидающее ответных действий от получателя. Используется в паре с ролью, имеющей стереотип «Tracker».
6 Tracker (контролер): приложение, способное получить информацию о возникновении важного события (изменение состояния существенного класса). Отправитель не ожидает от контролера выполнения каких-либо действий

Переходы состояний (State Transition)

Определение: допустимые значения данного компонента унаследованы от моделей перехода состояний, являющихся часть модели RIM и определенных в процессе гармонизации, принятом в Рабочей группе HL7. Эти значения описаны в той части стандарта, которая посвящена модели RIM.

Таблица G.10 — Последовательность сортировки и допустимые значения переходов состояний

Последовательность сортировки и допустимые значения
1 New (новый)
2 Cancel (отмена)
3 Hold (отложенное)
4 Release (освобождено)
5 Activate (активировано)
6 Suspend (приостановлено)
7 Resume (возобновлено)
8 Abort (прервано)
9 Complete (выполнено)
10 Reactivate (активировано заново)
11 Nullify (аннулировано)
12 Revise (пересмотрено)
13 Replace (заменено)

Способности, связанные с классами действий Act в наклонении event (событие)

Определение: описывает функции, которые приложении способно выполнить в терминах перехода состояний, если артефакт содержит атрибут наклонения mood с кодом event (событие), определенный в компоненте [Наклонение] структурированного сортируемого имени.

Таблица G.11 — Последовательность сортировки и допустимые значения способностей, связанных с классами действий Act в наклонении event (событие)

Последовательность сортировки и допустимые значения
1 Completion (завершение)
2 Existence (наличие)
3 Creator (создатель)
4 Cancellation (отмена)
5 Holder (откладывание)
6 Revision (пересмотр)
7 Replacement (замена)
8 Suspension (приостановка)
9 Abortion (прекращение)
10 Reactivation (повторное активирование)
11 Nullification (аннулирование)
12 Comprehensive (исчерпывающий)
13 Global (глобальный)

Способности, связанные с классами действий Act в наклонении, отличающемся от event (событие)

Определение: описывает функции, которые приложении способно выполнить в терминах перехода состояний, если артефакт содержит атрибут наклонения mood с кодом, отличающимся от event (событие), определенный в компоненте [Наклонение] структурированного сортируемого имени.

Таблица G.12 — Последовательность сортировки и допустимые значения способностей, связанных с классами действий Act в наклонении, отличающемся от event

Последовательность сортировки и допустимые значения
1 Existence (наличие)
2 Completion (завершение)
3 Creator (создатель)
4 Cancellation (отмена)
5 Holder (откладывание)
6 Revision (пересмотр)
7 Replacement (замена)
8 Suspension (приостановка)
9 Abortion (прекращение)
10 Reactivation (повторное активирование)
11 Nullification (аннулирование)
12 Comprehensive (исчерпывающий)
13 Global (глобальный)

Любые компоненты структурированного сортируемого имени, не соответствующие описанным выше требованиям или включающие недопустимые компоненты, сортируются по алфавиту.

G.3.2. Сценарий и текст сценария

Понятие сценария позаимствовано из индустрии кино и анимации, и полезно при разработке сообщений стандарта HL7 по тем же причинам, что зарекомендовали себя в этой индустрии, а именно:

- сценарий разделяет текст на последовательность «кадров» или событий в хронологическом порядке;
- каждый кадр представляет собой распознаваемый, значимый момент в последовательности событий, о котором читатель должен знать для понимания всей последовательности и результата;
- каждый кадр показывает основных участников сценария и их взаимодействие с другими участниками;
- в совокупности последовательность кадров предоставляет связное описание полного процесса или действия.

Сценарий состоит из краткого описания его цели и диаграммы взаимодействия, показывающей последовательность взаимодействий между ролями приложений. Текст сценария представляет собой описание события реальной жизни, предусматривающее контекст, необходимый для разработки конкретного взаимодействия, описанного в сценарии. Процесс создания сценариев закладывает фундамент для описания сообщений стандарта HL7 и их содержания.

G.3.2.1 Формат сценария

В стандарте версии 3 сценарии состоят из следующих разделов:

1 Имя — короткая, описательная фраза.

2 Идентификатор артефакта — код из 16 символов, в котором в качестве кода артефакта указан ST;

3 Цель — краткое описание, задающее основной набор действий, представляемых сценарием.

4 Пример, показывающий описанные выше элементы сценариев, приведен на рисунке G.9.

2.1 Storyboards

Go To Top

6 Storyboards (Sorted by Title)
- Add New Person(PRPA_ST101001)
- Add New Person Request(PRPA_ST101201)
- Nullify Person(PRPA_ST101999)
- Person Registry Find Associated Identifiers Query(QUPA_ST101003)
- Person Registry Find Candidates Query(QUPA_ST101002)
- Person Registry Get Demographics Query(QUPA_ST101001)
- Resolve Duplicate Person Registrations(PRPA_ST101004)
- Revise Person Information(PRPA_ST101002)
- Revise Person Information Request(PRPA_ST101202)

6 Storyboards (Sorted by Structured Sort Name)
- Person Notification - Activate(PRPA_ST101001)
- Person Notification - Edit(PRPA_ST101002)
- Person Notification - Nullify(PRPA_ST101999)
- Person Registry Find Candidates Query(QUPA_ST101002)
- Person Registry Get Associated Ids Query(QUPA_ST101003)
- Person Registry Get Demographics Query(QUPA_ST101001)
- Person Request Add(PRPA_ST101201)
- Person Request Revise(PRPA_ST101202)
- Person Resolve Duplicates(PRPA_ST101004)

6 Storyboards (Sorted by Display Order)
- Add New Person(PRPA_ST101001)
- Revise Person Information(PRPA_ST101002)
- Nullify Person(PRPA_ST101999)
- Person Registry Find Candidates Query(QUPA_ST101002)
- Person Registry Find Associated Identifiers Query(QUPA_ST101003)
- Person Registry Get Demographics Query(QUPA_ST101001)
- Add New Person Request(PRPA_ST101201)
- Revise Person Information Request(PRPA_ST101202)
- Resolve Duplicate Person Registrations(PRPA_ST101004)

Рисунок G.9 — Пример элементов сценариев

5 Диаграмма взаимодействия — отражает взаимодействия между ролями приложений. Обычно эти взаимодействия изображаются в виде диаграмм последовательности. Пример диаграммы взаимодействия приведен на рисунке G.10. Отметим, что прямоугольники вверху диаграммы и связанные с ними вертикальные линии представляют роли приложений типы систем, отправляющих и принимающих сообщения). Горизонтальные стрелки, начинающиеся в верхнем левом углу диаграммы и спускающиеся вниз к нижнему правому углу, показывают взаимодействия между различными ролями приложений. Имена и коды артефактов этих взаимодействий отмечаются на линиях стрелок.

Управление клиническими документами
Диаграмма взаимодействия в сценарии
(включены только взаимодействия, описанные в сценарии)

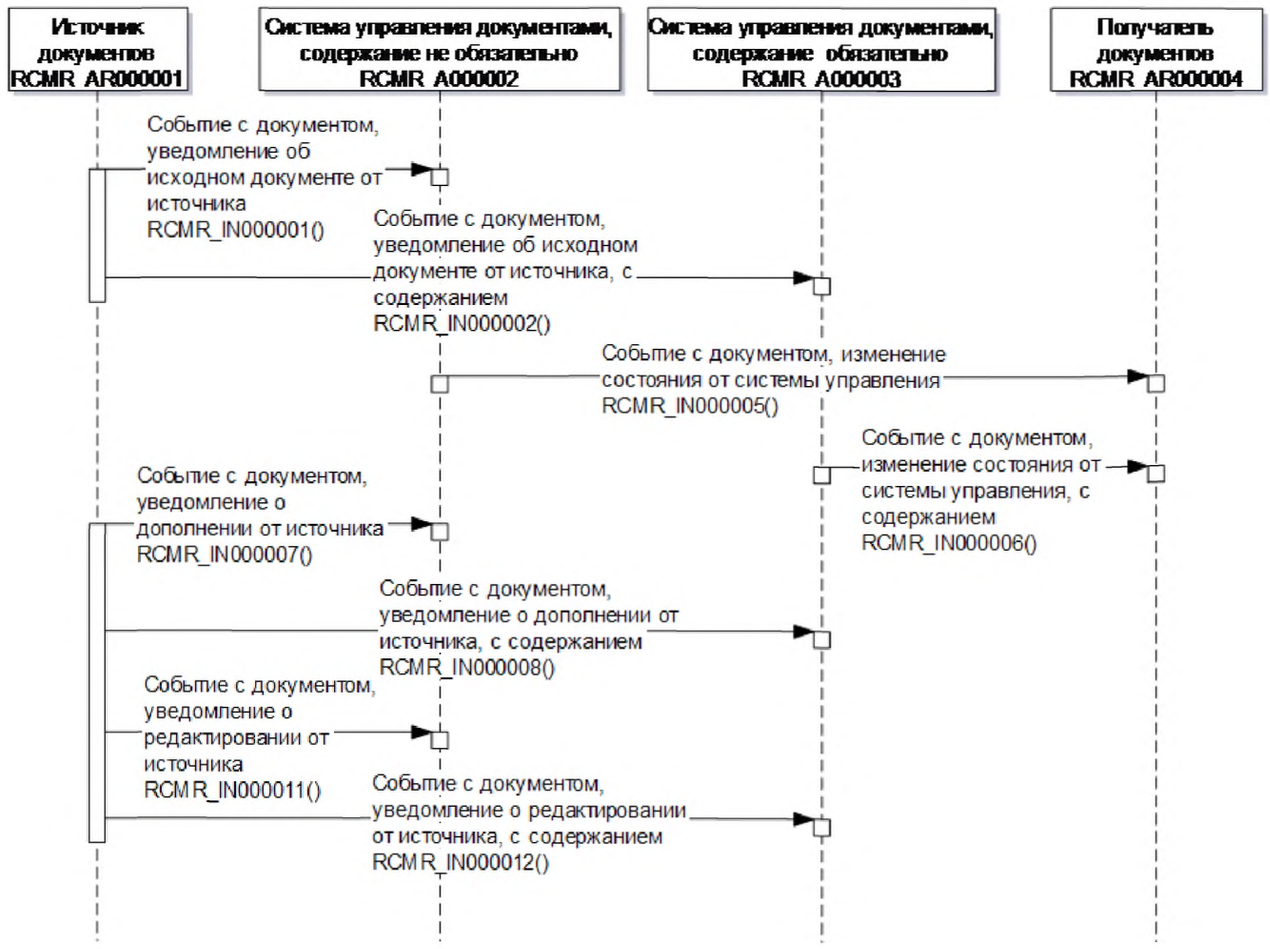


Рисунок G.10 — Пример диаграммы взаимодействия

6 Список взаимодействий — следует непосредственно за диаграммой взаимодействия.

Рисунок G.11 — Пример списка взаимодействий

7 Один или несколько текстов сценария — следом за списком взаимодействий идет вербальное описание каждого элемента из списка взаимодействий. Отметим, что повествование включает название и номер объекта, и служит примером представления взаимодействия с помощью естественного языка. Пример повествования приведен на рисунке G.12.

2.1.4.1 Patient Activate (PRPA_SN201001)

Mr. Adam Everyman's physician, Dr. Patricia Primary, called the Good Health Hospital to schedule an inpatient visit for Mr. Everyman for lung surgery. The clerk located the existing entry for Mr. Everyman in the GHH Person Registry and added added Mr. Everyman to the GHH Patient Registry along with the mailing address and telephone number to use for communicating patient information to him [Trigger Event Patient Activate Notification] and scheduled the admission for two weeks from that day.

2.1.4.2 PatientLivingSubject Activate (PRPA_SN201101)

Mr. Adam Everyman's physician, Dr. Patricia Primary, called the Good Health Hospital to schedule an inpatient visit for Mr. Everyman for lung surgery. The clerk added Mr. Everyman's name, demographics, address and telephone number to the GHH Patient Registry along with the mailing address and telephone number to use for communicating patient information to him [Trigger Event PatientLivingSubject Activate Notification] and scheduled the admission for two weeks from that day.

Рисунок G.12 — Пример текстов сценариев

G.3.3 Роли приложений

Роли приложений представляют собой совокупность коммуникационных функций, которые могут быть выполнены приложением. Например, они описывают компоненты или субкомпоненты системы, отправляющие и/или принимающие взаимодействия.

G.3.3.1 Нормативный статус

Несмотря на то, что роли приложения документированы в нормативных спецификациях предметных областей, Технический руководящий комитет Рабочей группы HL7 определил в выпуске 1 версии 3, что роли приложений должны быть справочными, а не нормативными. Причина этого в том, что прежде чем определять содержание и детализацию ролей приложения, Рабочая группа HL7 хотела бы ознакомиться с рядом реализаций версии 3. Роли приложений станут нормативными не раньше, чем в выпуске 2. Несмотря на это ограничение, спецификация соответствия указывает, что в будущем роли приложений будут взяты за основу требований соответствия.

G.3.3.2 Ответственность

При описании взаимодействия одна роль приложения отвечает за отправку (инициирование) этого взаимодействия, а другая роль отвечает за получение этого взаимодействия и выполнение необходимых ответных действий. Ответственность за отправку и получение взаимодействия может быть возложена на одну и ту же роль приложения.

G.3.3.3 Документирование

Роли приложения, показанные на диаграмме взаимодействия, описаны в разделе «Сценарий». Прямоугольники в верхней части диаграммы и соответствующие вертикальные линии представляют роли приложения. Каждая роль приложения, показанная на диаграмме взаимодействия, упомянута в главе предметной области. Отметим, что, как и многие другие объекты, роли приложений имеют имя, идентификатор артефакта и описание. Имя также присвоено структурированное имя, используемое при логическом упорядочивании ролей приложений в конкретной предметной области.

Пример списка ролей приложений приведен на рисунке G.13.

2.2 Application Roles

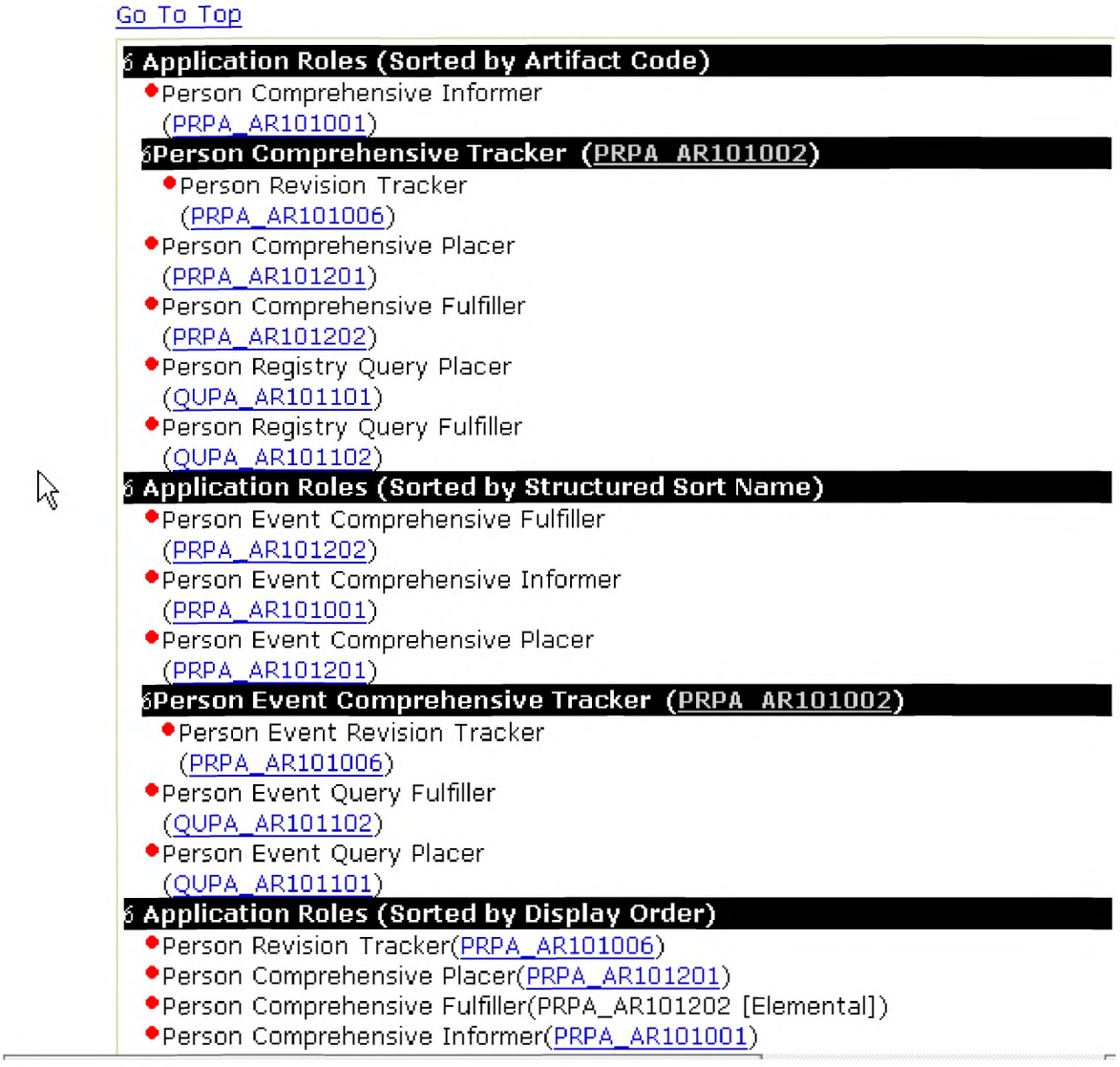


Рисунок G.13 — Пример списка ролей приложений

G.3.4 Инициирующие события

Инициирующее событие представляет собой явно описанное множество условий, при выполнении которых инициируется передача информации между компонентами системы (ролями приложений). К ним относятся реальные события, например, завершение подготовки заказа на лабораторные анализы или на лекарства. Возникновение инициирующего события должно систематически распознаваться автоматизированной системой.

В рамках стандартов версии 3 инициирующим событиям присваиваются имя, идентификатор артефакта, описание и тип. Для логического упорядочения инициирующих событий в заданной предметной области Технический комитет присваивает им структурированные имена. Требуемое структурированное имя инициирующего события должно содержать компоненты наклонения (Mood), перехода состояний (State Transition) и тип (Type), однако учтите, что еще не все комитеты следуют этим рекомендациям.

Большинству инициирующих событий в стандарте HL7, версия 3 присваивается определенный тип, относящийся к одной из следующих категорий:

события, основанные на взаимодействии (Interaction Based): инициирующие события могут быть основаны на другом взаимодействии. Например, ответ на запрос (являющийся взаимодействием) служит примером такого типа взаимодействий;

события, основанные на переходе состояний (State-Transition Based): такие события возникают при переходе состояний, описанном для конкретного взаимодействия в Модели перехода состояний. Например, событие отмены документа может рассматриваться как инициирующее событие, основанное на переходе состояний;

события, основанные на запросе пользователя (User Request Based): инициирующие события могут быть основаны на пользовательском запросе. Например, инициирующее событие, требующее, чтобы некоторая система каждые 12 часов отправляла накопленные данные контролирующей системе, может рассматриваться как событие, основанное на запросе пользователей.

Большинство инициирующих событий основано на переходе состояний, и обнаруживаются при чтении информации динамической модели сообщения, предназначенной для поддержки конкретного взаимодействия, осуществляемого с помощью сообщений. Некоторые инициирующие события могут зависеть от нескольких одновременно происходящих переходов состояний. В некоторых случаях инициирующие события не могут быть отнесены ни к одной из трех указанных выше категорий. Тогда типу события присваивается значение Unspecified (не указан). Когда тип инициирующего события указан, то он определяет ответственность участников взаимодействий, инициированных данным событием.

На рисунке G.14 показано, как в рамках спецификации HL7, версия 3 документируются и представляются инициирующие события:

При определении пользовательских уточнений инициирующих событий технические комитеты поощряют создание «поясняющих» моделей R-MIM, содержащих только атрибуты и отношения, уместные для данного сообщения. Подобные описательные модели R-MIM представлены в соответствующих главах в виде диаграмм Visio, но поскольку в них не указаны все атрибуты и отношения, их нельзя непосредственно использовать для описания сообщений. Для каждой поясняющей модели R-MIM комитет должен создать соответствующую «полную» модель R-MIM, в которой атрибуты не удалены и не запрещены и в которой учтены все необходимые ограничения.

Для инициирующих событий разработчики могут проектировать также собственные управляющие действия Control Act, используемые по местным соглашениям. Однако для обеспечения совместимости со стандартом HL7, необходимой отправителям, придерживающихся официального стандарта, в таких собственных управляющих действиях не должны удаляться или запрещаться атрибуты или отношения.

3.3 Trigger Events

Go To Top

6 Trigger Events (Sorted by Title)
- Duplicate Patient Records Resolved(PRPA_TE201004)
- New Patient Added(PRPA_TE201001)
- Patient Information Revised(PRPA_TE201002)
- Patient Nullified(PRPA_TE201999)

6 Trigger Events (Sorted by Structured Sort Name)
- Patientlivingsubject Event Activate Notification(PRPA_TE201001)
- Patientlivingsubject Event Nullify Notification(PRPA_TE201999)
- Patientlivingsubject Event Obsolete Notification(PRPA_TE201004)
- Patientlivingsubject Event Revise Notification(PRPA_TE201002)

6 Trigger Events (Sorted by Display Order)
- New Patient Added(PRPA_TE201001)
- Patient Information Revised(PRPA_TE201002)
- Patient Nullified(PRPA_TE201999)
- Duplicate Patient Records Resolved(PRPA_TE201004)

Рисунок G.14 — Пример списка инициирующих событий

G.3.5 Информационные модели предметных областей

Информационная модель предметной области D-MIM (Domain Message Information Model) представляет собой подмножество модели RIM, включающее полностью расширенный набор классов, атрибутов и отношений, используемых при создании сообщений, используемых в конкретной предметной области. Например, совокупность классов, используемая в предметной области Medical Records/Structured Documents (медицинские карты/структурированные документы), существенно отличается от той, что используется в предметной области Patient Administration (движение пациентов). Модели D-MIM этих двух предметных областей будут значительно отличаться, хотя обе они произведены от одной модели RIM.

Как и другие модели, входящие в документы стандарта Версии 3, информационная модель предметной области представлена в виде диаграммы, на которой показаны отношения между классами. При этом на диаграммах

используются соглашения и обозначения, разработанные в рамках стандарта HL7 для представления специальных семантических конструкций, содержащихся в критичных, «основополагающих» классах модели RIM. Несмотря на то, что модели D-MIM и R-MIM могут быть, как и модель RIM, представлены в UML-нотации, та нотация, что предложена Рабочей группой HL7, предоставляет более подробную информацию о представляемых специфичных ограничениях и клонах классов. Соглашение по составлению диаграмм, принятые Рабочей группой HL7, предусматривает упрощение некоторых соглашений об отношениях, в результате чего диаграммы становятся меньше и лаконичнее, а также предоставляют более подробную визуальную информацию. Ключом к пониманию модели D-MIM является ознакомление с соглашениями и обозначениями, принятыми при построении диаграмм. Соглашения, используемые в моделях D-MIM, распространяются также на модели R-MIM и типы CMET (они обсуждаются в отдельных разделах настоящего руководства). Остальная часть настоящего подраздела посвящена соглашениям и обозначениям, используемым на диаграммах модели D-MIM. Поскольку обычно диаграмма модели D-MIM не помещается на одной странице, рисунки, приведенные в данном подразделе, представляют примеры отдельных частей диаграммы модели D-MIM, соответствующих рассматриваемым соглашениям.

G.3.5.1 Точки входа

Каждая диаграмма модели D-MIM имеет по меньшей мере одну точку входа (Entry Point). Точки входа обозначают класс или классы, которые служат началом сообщений, конструируемых в данной предметной области. Поскольку модель D-MIM охватывает всю предметную область, может существовать несколько точек входа, по одной для каждой модели R-MIM, входящей в эту предметную область. Точки входа обозначаются на диаграмме модели D-MIM не закрашенным прямоугольником с жирными черными границами. Из точки входа выходит жирная черная стрелка, указывающая на класс, рассматриваемый корневым, или центральным, для одного или нескольких иерархических описаний сообщений HMD. Каждая точка входа содержит имя, идентификатор артефакта модели R-MIM, порождаемой этой точкой, а также краткое описание. На рисунке G.15 показан пример обозначения точки входа:

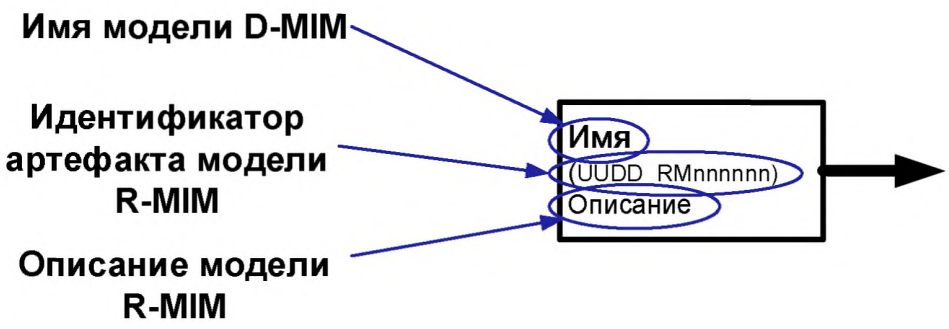


Рисунок G.15 — Точка входа модели D-MIM

G.3.5.2 Цветовое обозначение классов

Как и в модели RIM, «строительными блоками» модели D-MIM являются классы, определенные в модели RIM, и их атрибуты. Однако в модели D-MIM один и тот же класс может встречаться на диаграмме несколько раз с различными ограничениями (описанными ниже) или ассоциациями. Подобные многократные экземпляры одного и того же класса принято называть «клонами». Например, модель D-MIM может включать множество клонов класса Participation (участие), одни из которых служат для обозначения участия «авторов», другие — для обозначения «субъектов», и т. д. Некоторые классы могут вовсе не встречаться на диаграмме. В модель D-MIM включаются только те классы, атрибуты и отношения, которые необходимы для создания сообщений в данной предметной области.

На диаграмме модели D-MIM классы обозначаются прямоугольниками (подобно тому, как они представлены на диаграмме модели RIM). Имя клона класса (имя, относящееся к конкретному экземпляру класса) располагается в верхней левой части прямоугольника. Имена клонов присваиваются в соответствии с формальным алгоритмом, обсуждаемым в следующих подразделах. Цвет прямоугольника, обозначающего класс, соответствует области модели RIM, которой принадлежит этот класс. Классы, являющиеся клонами класса Act (действие), обозначаются красным цветом, клоны класса Entity (сущность) — зеленым, а клоны класса Role (роль) — желтым цветом.

В тех случаях, когда в модель D-MIM включены основные классы модели RIM (Act (действие), Entity (сущность) или Role (роль)), в качестве первого атрибута клона выступает атрибут class_cd основного класса вместе со своим значением.

Не основные классы модели RIM, входящие в модель D_MIM, обозначаются темно-синими прямоугольниками. Как и в других классах, имя клона размещается в верхней левой части прямоугольника. Физическое имя класса (как оно используется в модели RIM) отображается в круглых скобках под именем клона. На рисунке G.16 приведен пример отображения классов на диаграмме модели D-MIM.

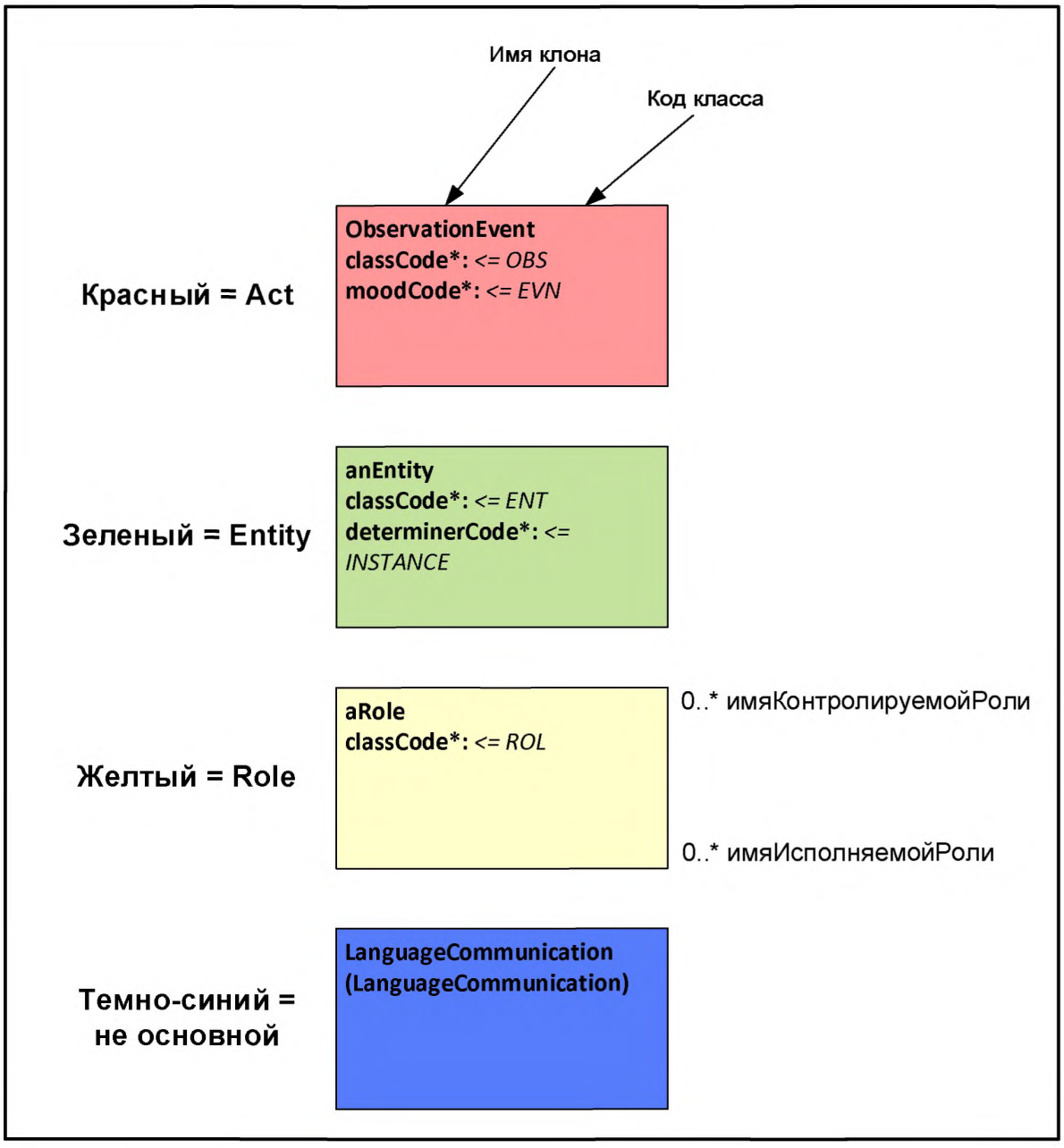


Рисунок G.16 — Цветовые обозначения классов

G.3.5.3 Формальные имена клонов классов и ассоциаций

Стандарт HL7 позволяет переименовывать копии существующих классов в определенном контексте. Данный процесс носит название «клонирование» (cloning) и его результатом является «клон» (clone) класса. Рабочей группой HL7 определен формальный алгоритм именования клонов и ассоциаций, появляющихся в модели D-MIM или модели R-MIM. Он гарантирует, что похожие понятия будут иметь одинаковые или похожие имена независимо от того, где они используются в спецификациях стандарта HL7. Имена клонов классов Entity (сущность), Role (роль) и Act (действие) основаны на значении атрибута class_cd соответствующего класса. Кроме этого, имена клонов классов Entity и Act зависят также от значений атрибутов mood_cd или determiner_cd. Например, клон класса Act со значениями атрибутов class_cd = OBS (Observation) и mood_cd = PRMS (Promise) получит название ObservationPromise.

Аналогично, значения атрибута type_cd используются для именования клонов классов Participation (участие), Act_relationship (отношение действий) и Role_link (роль связи). Так, клон класса Act_relationship со значением атри-

бута type_cd = COMP (имеет компонент) получит название Component. Ассоциации в модели D-MIM именуются в соответствии с теми же алгоритмами, но помимо этого в имени отражается направление движения. Таким образом, отношение класса Entity, контролирующего роль Patient (пациент), получит название healthCareProvider, при этом движение от класса Entity к классу PatientRole будет именоваться patient.

Каждый ассоциативный класс имеет четыре имени (входящее имя источника отношения; входящее имя приемника отношения; исходящее имя источника отношения и исходящее имя приемника отношения). Имена ассоциаций должны быть уникальными для элемента, из которого они исходят, но в модели D-MIM или в модели R-MIM они не обязаны должны быть уникальными. При необходимости уникальность имен достигается за счет нумерации имен ассоциации, например, имя, за которым следует целое число, «dataenterer02».

Если конкретное имя клона должно встречаться в модели дважды, то к имени второго вхождения добавляется суффикс «A» и так далее. Технические комитеты наделены правом заменить этот суффикс на более содержательный термин, а также присоединять добавочные суффиксы к любому формальному имени, используемому в модели.

Требуется, чтобы технические комитеты определяли «бизнес-имя» для каждого клона класса Act, Role и Entity. Данные имена должны быть уникальными в рамках модели D-MIM или модели R-MIM. Присвоенное бизнес-имя будет отображаться в диаграммах R-MIM, построенных с помощью Visio, а также использоваться в качестве имени элемента в XML ITS.

G.3.5.4 Контролирующие сущности и исполняемые роли: штриховые и пунктирные линии

Взглянув на клоны классов Role (желтые прямоугольники) и клоны классов Entity (зеленые прямоугольники) на диаграмме модели D-MIM, можно увидеть соединяющие их сплошные и пунктирные линии. Каждый клон класса Role, окрашенный в желтый цвет, может быть связан максимум с двумя клонами классов Entity, окрашенными в зеленый цвет. Эти линии представляют отношения между клонами классов Role и Entity. Один из клонов класса Entity является исполнителем роли (например, пациент), а другой представляет сущность, контролирующую роль (например, организацию здравоохранения). Сплошная линия, проведенная к роли, идентифицирует клон класса Entity, описывающий исполнителя (например, класс Person представляет лицо, выступающее в роли пациента), а штриховая линия идентифицирует клон класса Entity, описывающий контролирующую сущность (например, класс Organization представляет организацию, назначившую роль пациенту). Посмотрите на отношения, показанные на диаграмме модели D-MIM для классов Patient, Person и Organization (рисунок G.17). Атрибут class_cd в классе Patient, окрашенном в желтый цвет, обозначающий роль, имеет значение PAT, указывающее, что исполняемой ролью является пациент. Мы Атрибут class_cd в классе Person имеет значение PSN, означающее физическое лицо, и отношение, проведенное от него к классу Patient, указывает, что это лицо, исполняющее роль пациента. При этом атрибут id класса Patient будет содержать номер медицинской карты пациента. Сущность, присвоившая пациенту этот номер (а также определившая, что лицо является пациентом), представлена классом Organization (организация), и отношение, проведенное от него к классу Patient, указывает, что это организация, контролирующая роль пациента. Класс Organization является клоном класса Entity, у которого атрибут class_cd имеет значение ORG, означающее организацию.

Кратность ассоциаций показана рядом с линиями. Как показано на рисунке G.17, класс Organization, обозначенный зеленым прямоугольником, контролирует нуль и более пациентов (идентифицируемых экземплярами класса Patient) (т. е. организация назначает роль пациента нулю или более лиц, идентифицируемых экземплярами класса Person). В то же время каждый пациент контролируется одной (и только одной) организацией. Вместе с тем роль пациента исполняется только одним лицом, идентифицируемым классом Person, а каждое лицо может выступать в роли нуля или более пациентов (например, оно может получать медицинскую помощь в нескольких организациях).

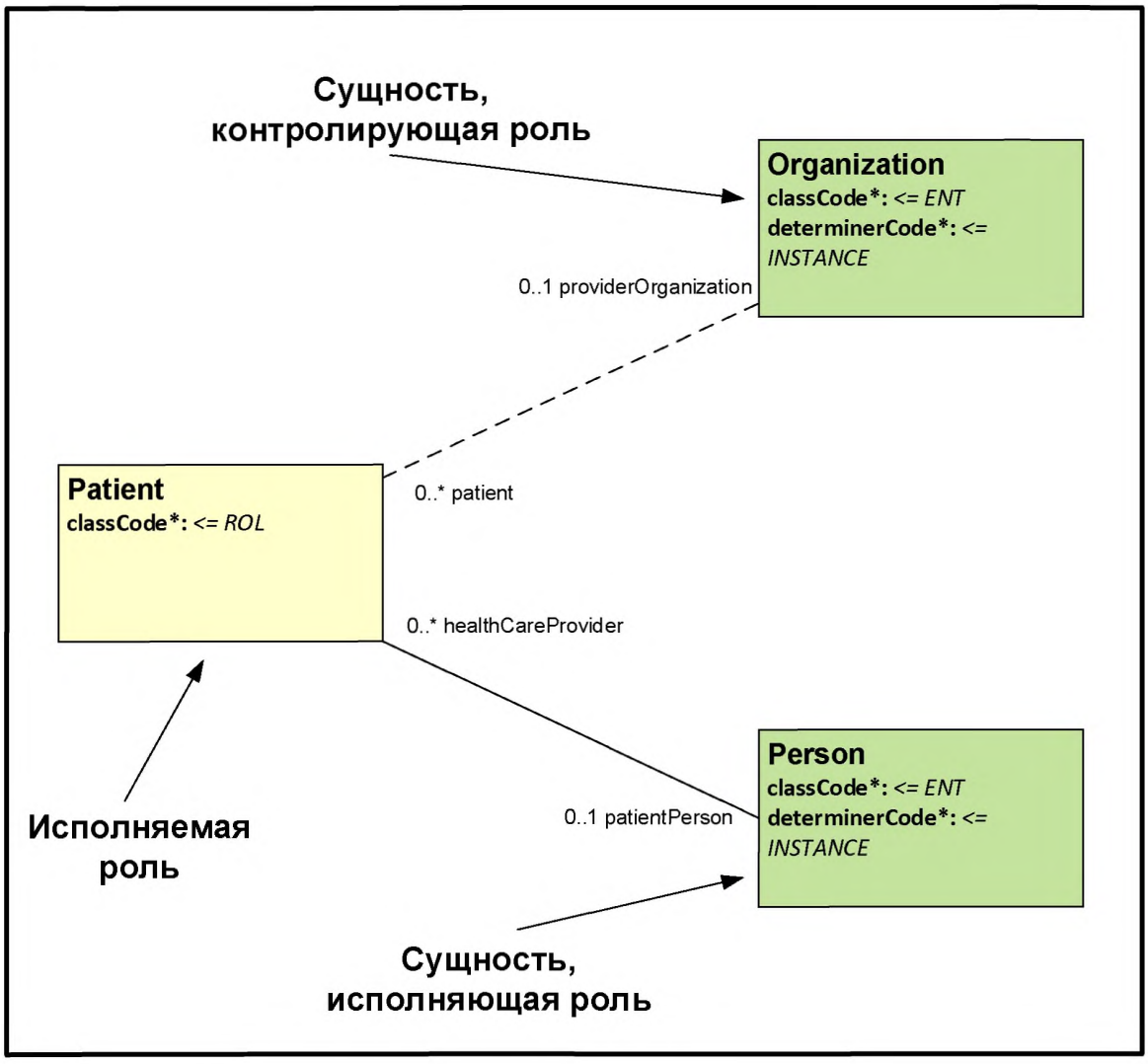


Рисунок G.17 — Контролирующие сущности и исполняемые роли

G.3.5.5 Классы отношений: стреловидные блоки

В диаграмме модели D-MIM могут встречаться три типа стреловидных блоков. Розовые блоки обозначают клоны класса Act_relationship (отношение действия), синие стреловидные блоки обозначают клоны класса Participation (участие), а желтые стреловидные блоки обозначают клоны класса Role_link (связь ролей). Хвостовая часть стреловидного блока связана с исходным классом, в то время как головная часть указывает на результирующий клон класса. Ниже описано каждое из направленных отношений:

- клоны класса Act_relationship — класс Act_relationship соединяет два экземпляра класса Act (действие) и тем самым определяет отношение между ними;
- клоны класса Participation — классы участия (participation) соединяют классы Act (действие) и Role (роль) и тем самым определяют отношения между ними. Значение атрибута type_cd класса Participation определяет, каким образом сущность (Entity) исполняет роль (Role) в конкретном экземпляре класса Act;
- клоны класса Role_link — эти клоны выполняют то, что указано в названии класса, а именно, связывают два экземпляра класса Role, определяя отношение между ними.

На рисунке приведены условные обозначения клонов классов Relationship, Participation и Role_link:

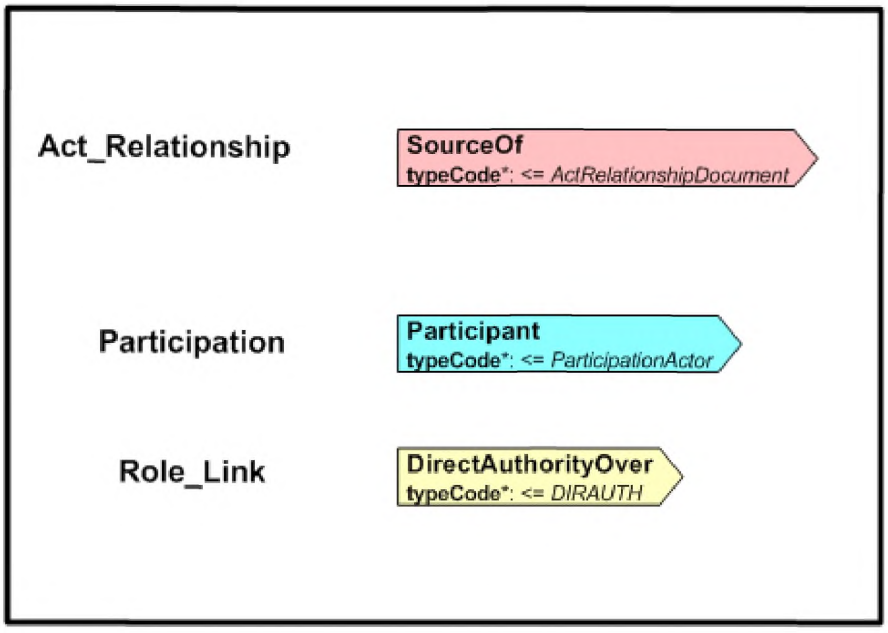


Рисунок G.18 — Условные обозначения клонов классов Relationship, Participation и Role_link

Дополнительные ассоциации классов (Non-core Class Associations) — обозначаются, как и классы, из которых они исходят, темно-синими стрелками. Рядом со стрелками расположены имена ассоциаций и их кратности.

На рисунке G.19 приведены условные обозначения дополнительных ассоциаций классов.

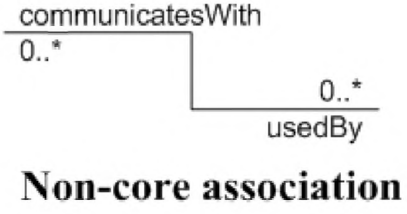


Рисунок G.19 — Условные обозначения дополнительных ассоциаций классов

G.3.5.6 Рекурсивные отношения

Отношения действий (Act_relationship) и связи ролей (Role_link) могут быть рекурсивными. На диаграммах моделей D-MIM эти рекурсивные отношения обозначаются цветными блоками, стреловидный конец которых заменен пилообразным вырезом. Граница блока представлена стреловидной линией, указывающей на пилообразный вырез блока. На рисунке G.19 показаны условные обозначения классов рекурсивных отношений.

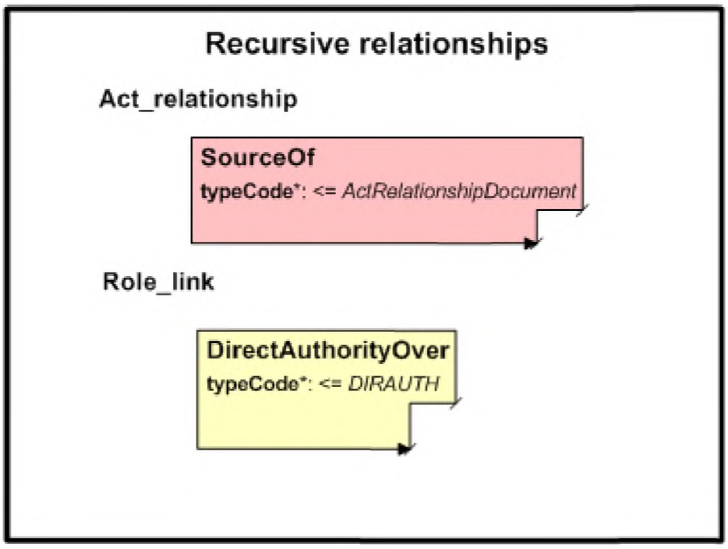


Рисунок G.19 — Условные обозначения рекурсивных отношений

G.3.5.7 Наследование

Недавнее дополнение к стандарту Версии 3 позволяет динамически управлять стандартными правилами наследования, которые обычно используются в языке UML и в объектном моделировании.

Атрибут ControlCode контекста

Этот атрибут может быть при необходимости добавлен к любому «участию» (participation) для управления тем, заменяет ли данная ассоциация ранее описанное участие, или добавляется к нему (т. е. распространяет предшествующий контекст). Получатель должен будет реализовать механизмы, которые могли бы активно учитывать значение данного атрибута в любом экземпляре сообщения, чтобы корректно работать с этой информацией об участии.

G.3.5.8 Типы CMET

Типы CMET (Common Message Element Type — тип общего элемента сообщения) представляют собой предопределенные компоненты, используемые в нескольких моделях R-MIM или в сообщениях. Подробному описанию типов CMET посвящен отдельный раздел настоящего руководства. На диаграммах моделей D-MIM они обозначаются блоками с пунктирными границами. Цвет блока определяется в соответствии с цветом базового класса: розовый для класса Act (действие), желтый для класса Role (роль) и зеленый для класса Entity (сущность). Блок CMET содержит имя, тип базового класса, идентификатор артефакта и уровень атрибутивности. Для каждого типа CMET существует отдельная диаграмма соответствующей ему модели R-MIM. Каждая из них описывается отдельно и может быть найдена в подразделе CMET. На рисунке G.20 показаны условные обозначения CMET на диаграмме модели D-MIM.

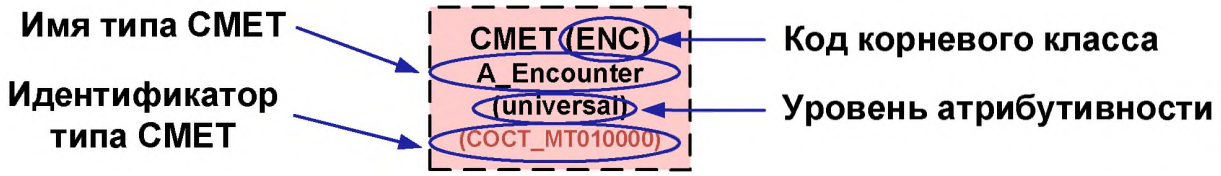


Рисунок G.20 — Представление CMET на диаграмме модели D-MIM

G.3.5.9 Альтернативы

Диаграммы моделей D-MIM могут содержать блоки «альтернатив» (Choice). Эти блоки имеют границы в виде пунктирных линий и включают в себе два или более класса, входящих в иерархию наследования (например, два или более клона класса Role, два или более клона класса Entity и т. д.). Важно отметить, что любые аннотации или ассоциации, связанные с блоком альтернатив, применяются ко всем классам блока. Ассоциации, связанные с отдельным классом блока альтернатив, применяются только к этому классу. На рисунке G.21 показано, что отношение Act_relationship AR_for_choice применяется ко всем классам блока альтернатив, а отношение Act_relationship AR_for_class применяется только к классу ActA.

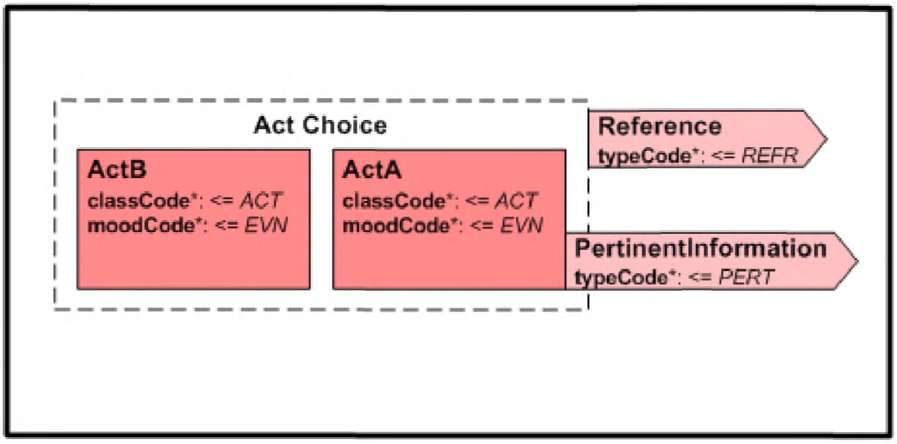


Рисунок G.21 — Ассоциации с блоком альтернатив

G.3.5.10 Соглашения по атрибутам

На диаграммах моделей D-MIM может использоваться ряд соглашений по атрибутам, передающих информацию, которая иным способом не была бы достаточно очевидной. Обычно эти соглашения используются в тех случаях, когда не определены значения по умолчанию, хотя они могут использоваться и в том случае, когда значения по умолчанию заданы. Соглашения, применяемые к конкретному атрибуту, указываются под ним с небольшим отступом. Могут использоваться следующие соглашения:

- обязательное включение (Mandatory inclusion) — выделение имени атрибута полужирным шрифтом означает, что атрибут является обязательным;
- домен (Domain) — словарный домен, связанный с конкретным экземпляром атрибута, обозначается символами «<=» или «=». Спецификацией домена является или имя домена, определенное в контролируемом словаре, или конкретный код из соответствующего домена;
- тип данных (Data type) — тип данных атрибута может быть указан после имени атрибута, отделенный двоеточием («:»);
- квалификатор расширения (Vocabulary strength) — обозначается с помощью кода CWE (coded with extension — кодированный с расширением), указывающего, что можно передать не кодированное значение, если соответствующего ему кода нет; или с помощью кода CNE (coded, no extension — кодированный без расширения), указывающего, что все значения должны быть кодированными;
- кратность (Cardinality) — кратность атрибутов указывается в квадратных скобках [ ]. Она задается как N...N, где N может быть числом или * для неограниченных повторений. Например, [0...1];
- описание (Description) — может быть добавлено краткое описание атрибута. Описание заключаются в круглые скобки;
- альтернативное имя (Alias) — исходное имя атрибута. Используется в тех случаях, когда имя атрибута, показанного на диаграмме модели D-MIM, отличается от имени атрибута по умолчанию, указанному в модели RIM. При этом имя атрибута по умолчанию, указанное в модели RIM, изображается в фигурных скобах, и ему предшествует «AKA» Например, {AKA repeat_nbr} обозначает атрибут, имеющий в модели RIM имя repeat_nbr.

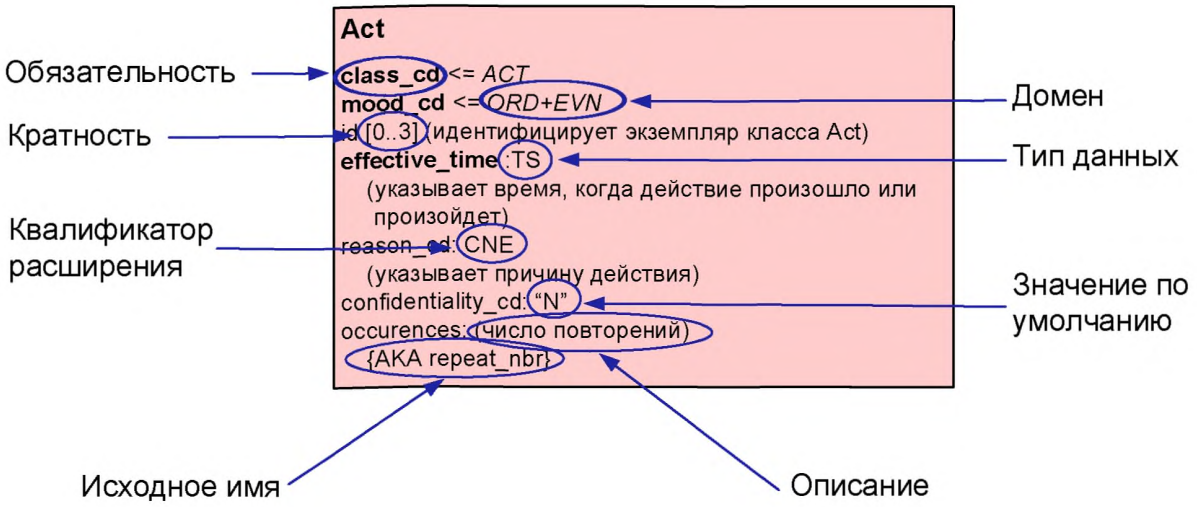


Рисунок G.22 — Обозначения соглашений по атрибутам в модели D-MIM

G.3.5.11 Примечания

На диаграмме модели D-MIM могут присутствовать примечания, содержащие дополнительную информацию о конкретном классе. Примечания изображаются в виде бесцветных продолговатых блоков с черной границей, у которых правый верхний угол подвернут. Из блока примечания исходит стрелка, указывающая на описываемый класс. Пример обозначения примечания показан на рисунке G.23.

Примечание используется для указания комментария, относящегося ко всему классу. Оно соединяется стрелкой с этим классом.

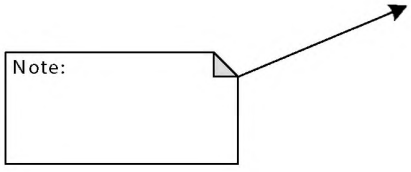


Рисунок G.23 — Обозначение примечания на диаграмме модели D-MIM

G.3.5.12 Ограничения

Ограничения могут быть наложены на класс или атрибут. Примером ограничения, наложенного на класс, может служить «должен быть задан либо атрибут nm, либо атрибут id». Ограничения обозначаются светло-синим затененным блоком, указывающим на ограничиваемый класс или атрибут. Если ограничение распространяется только на отдельный атрибут, в левом верхнем углу блока содержится имя атрибута. Если же это поле пусто, то ограничение распространяется на весь класс, на который указывает блок. Кроме того, блок ограничения содержит формальное описание ограничения, заключенное в фигурные скобки, а следом за ним — текстовое описание ограничения. Пример обозначения ограничения показан на рисунке G.24.

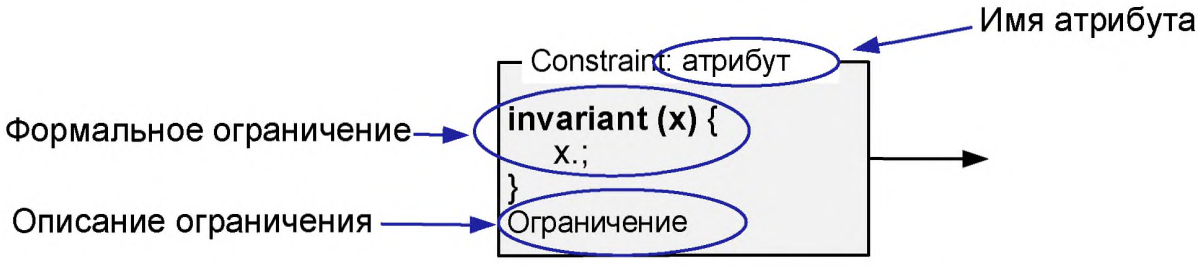


Рисунок G.24 — Обозначение ограничения на диаграмме модели D-MIM

G.3.5.13 Аннотации

В самих стандартах диаграммы моделей D-MIM сопровождаются аннотациями (annotations). Выделяют аннотации двух типов: комментарии к разработке или разбор модели. Комментарии к разработке поясняют, почему объекты смоделированы именно таким образом. Второй вид комментариев предназначен для того, чтобы помочь читателю разобраться в особенностях модели D-MIM.

G.3.6 Уточненные информационные модели сообщений

Уточненные информационные модели сообщений (Refined Message Information Models — R-MIM) используются для представления содержания информации, предназначенной для создания одного или нескольких иерархических описаний сообщений HMD, производимых от корневого класса, идентифицированного точкой входа (entry point) в модели R-MIM. Каждая модель R-MIM является подмножеством модели D-MIM и содержит только те классы, атрибуты и ассоциации, которые требуются для составления множества сообщений, производимых из иерархических описаний сообщений HMD, которые в свою очередь произведены от корневого класса модели R-MIM. Классы, атрибуты и ассоциации, которые не требуются для этих иерархических описаний сообщений HMD, исключаются, а иерархии генерализации сокращаются.

С каждым типом CMET связана определенная модель R-MIM, которые можно найти в главе, посвященной типам CMET и описывающей все используемые в настоящее время типы CMET.

В диаграммах моделей R-MIM используются те же соглашения и обозначения, что и в диаграммах моделей D-MIM. Для получения дополнительной информации о том, как правильно читать эти диаграммы, обратитесь к соответствующему подразделу. Однако для модели R-MIM предусмотрен дополнительный уровень документирования. Теперь модель R-MIM содержит гиперссылки на каждый входящий в нее класс, по которым можно перейти к табличному представлению. Это обеспечивает удобный способ ссылок на любую дополнительную документацию, представленную в табличной форме.

G.3.7 Иерархические описания сообщений и типы сообщений

Настоящий подраздел посвящен иерархическим описаниям сообщений (Hierarchical Message Descriptions — HMD) и получаемых из них типам сообщений, определяющих полезную нагрузку сообщения.

Часто задаются такие вопросы, как «Зачем создавать иерархическое описание сообщения HMD? Если известны отправитель, получатель, инициирующее событие и классы, почему бы просто не отправить данные?» Ниже приведено несколько важных ответов на этот вопрос.

На рисунке G.25 приведен фрагмент подраздела стандарта Версии 3, посвященного иерархическому описанию сообщения Person (PRPA_HD101001) (физическое лицо), определяющего содержание сообщений, уведомляющих об изменении состояния записи о лице в регистре физических лиц.

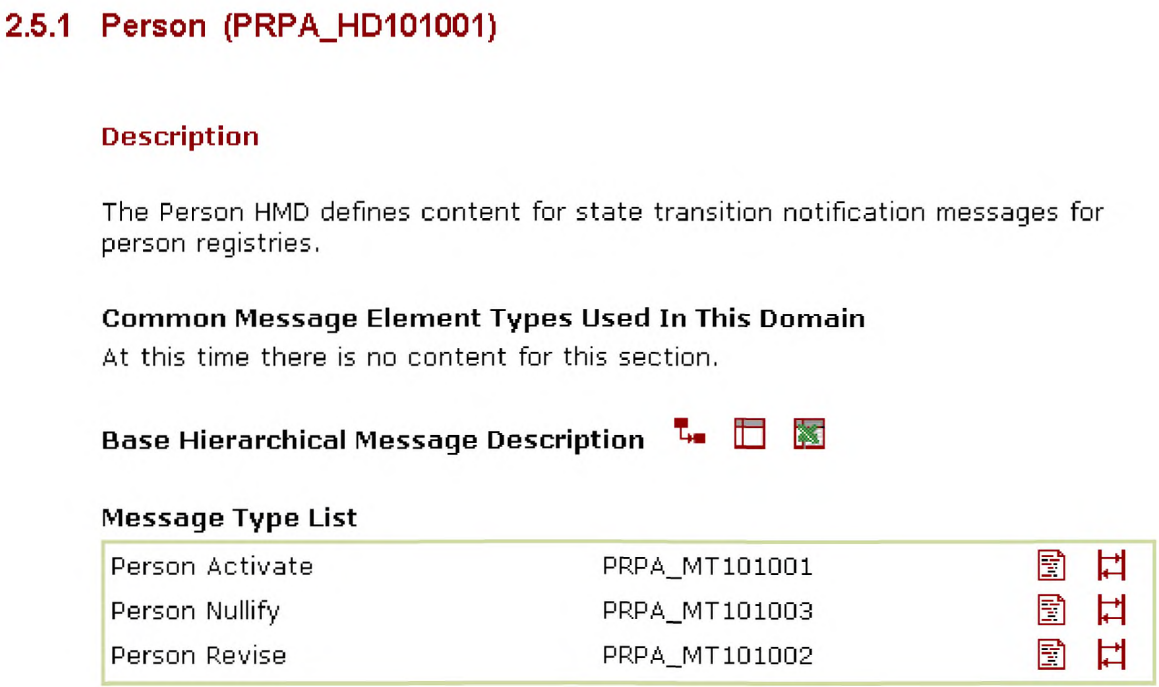


Рисунок G.25 — Фрагмент иерархического описания сообщения Person

Информационная модель содержит несколько классов, которые часто соединены более чем одним способом. Например, могут быть ассоциации, непосредственно соединяющие класс Patient (пациент) с классом Person (физическое лицо), и опосредованно соединяющие эти классы (при указании лица, являющегося близким родственником или лечащим врачом пациента).

Взаимодействующие системы должны иметь возможность определить, какой из объектов, произведенных от этих классов, содержит данные, подлежащие отправке. Кроме того, они должны иметь возможность переходить к связанным объектам по ассоциациям, описанных для этих классов.

Не все атрибуты могут требоваться для разных клонов одного и того же класса. Хотя как пациент, так и врач, указанные в направлении на исследование, являются людьми, маловероятно, что в каждом направлении будут переданы вероисповедание, дата рождения или пол врача.

Взаимодействующие системы должны заботиться о том, какой именно объект будет отправлен.

Наконец, для отправки информации по каналу передачи данных компьютер должен организовать ее последовательно. Существует множество различных способов последовательной организации данных, содержащихся в группе объектов, связанных ассоциациями. Если отправитель и получатель не согласуют точную последовательность передаваемых данных, то взаимодействие будет тщетным. Если отправитель передает информацию о лечащем враче стационара раньше информации об враче общей практики, а получатель ожидает данные в обратном порядке, то правильность обработки этой информации будет проблематичной.

Взаимодействующие системы должны знать точную последовательность передаваемой информации.

Эти варианты должны быть описаны в иерархическом описании сообщения HMD и в соответствующих ему типах сообщений. Проще говоря, иерархическое описание сообщения HMD является табличным представлением последовательности элементов, представленных в модели R-MIM (классов, атрибутов и ассоциаций), позволяющим описать сообщение без использования особенностей технологии реализации. В иерархическом описании сообщения HMD описана единственная базовая структура сообщения — «общий» тип сообщения. Эта базовая структура сообщения никогда не отправляется и потому не с ним не связано никакое инициирующее событие. Она выступает в качестве шаблона для определения других специфичных и соответствующих типов сообщений. По этой причине инструментальное средство RoseTree требует, чтобы было идентифицировано конкретное иерархическое описание сообщения HMD, по которому оно может выгрузить подходящее конкретное иерархическое описание сообщения HMD и создать XML-схему. Иерархическое описание сообщения HMD и произведенные из него типы сообщений могут выгружаться в виде листа Excel. Кроме того, доступен пример соответствующего XML-документа.

Тип сообщения представляет уникальный набор ограничений, применяемых к общему сообщению. В последующих выпусках стандарта могут быть представлены дополнительные типы сообщений, произведенные от иерархических описаний сообщений HMD, описанных в предыдущих выпусках. При этом типы сообщений будут дифференцированы в терминах инициирующих событий, для которых они предназначены.

G.3.7.1 Использование признаков обязательности и соответствия

В таблице G.13 описаны типы ограничений обязательности и соответствия, а в таблице G.14 показано, как признаки обязательности и соответствия используются при уточнении иерархического описания сообщения HMD.

Таблица G.13 — Типы ограничений обязательности и соответствия

Ограничение	Описание
Кратность (cardinality)	Задает минимальное и максимальное число экземпляров (повторений) поля/ассоциации. Например, 1..* означает, что минимальное количество экземпляров (повторений) равно 1, а максимальное число не ограничено
Обязательность (mandatory)	Возможные значения: M (mandatory — обязательный) или пусто (не обязательный). Значение M указывает, что поле или ассоциация должны присутствовать в сообщении, в противном случае сообщение считает-ся некорректным и не соответствующим. Для обязательных полей стандарт HL7 определяет минимальное значение кратности, равное 1 (значение поля должно присутствовать). Пустое значение обязательности означает, что поле или ассоциация могут не присутствовать в сообщении

Окончание таблицы G.13

Ограничение	Описание
Соответствие (conformance)	Возможные значения: R (required — требуется), NP (not permitted — запрещено), пусто (не требуется). Значение R (требуется) означает, что приложение-отправитель должно поддерживать это поле или ассоциа-цию. Если данные доступны, поле или ассоциация включаются в сооб-щение. Если минимальная кратность равна 0 и данные не доступны, то поле или ассоциация могут не включаться в сообщение и сообщение все еще считается соответствующим стандарту. Если минимальная кратность равна 1 и данные не доступны, то требуется передать значе-ние описанного ниже признака пустоты NullFlavor. Значение NP (запрещено) означает, что поле или ассоциация не добавляются в схему сообщения и никогда не включается в экземпляр сообщения. Пустое значение (не требуется) означает что поле или ассоциация могут отправляться, а могут и не отправляться и поддержка этого поля или ассоциации от приложения-отправителя не требуется. В этом случае соответствие элемента является предметом местных соглашений
Причина пустоты (NullFlavor)	Для требуемых полей или ассоциаций с минимальной кратностью 1, причина пустоты NullFlavor должна быть указана для всех полей или ассоциаций, для которых приложение-отправитель не имеет данных. Примером причины пустоты NullFlavor являются «нет информации», «неизвестный», «скрытый» (для приложений, обрабатывающих конфиденциальные данные), «не запрошены», «запрошены, но не известны» (пациенту задали вопрос, но он не знает правильный ответ)

Таблица G.14 — Использование признаков обязательности и соответствия

Кратность	Обязательность	Соответствие	Требуется NullFlavor?	Объяснения правил
0.., 1..			Нет	Поле или ассоциация могут присутствовать, а могут и не присутствовать в сообщении. Соответствие этого элемента является предметом местных соглашений
0.., 1..		NP	Не разрешен	Поле или ассоциация не отсутствуют в схеме сообщения и не должны включаться в экземпляр сообщения
0..		R	Нет	Приложение-отправитель должно поддерживать это поле или ассоциацию. Поле или ассоциация могут присутствовать, а могу и не присутствовать в сообщении
1..		R	Да	Приложение-отправитель должно поддерживать это поле или ассоциацию. Поле или ассоциация должны присутствовать в сообщении, но при этом могут не иметь значения. В этом случае должна быть указана причина пустоты NullFlavor
1..	M	R	Не разрешен	Приложение-отправитель должно поддерживать это поле или ассоциацию. Поле или ассоциация должны присутствовать в сообщении и иметь непустое значение

Тип общего сообщения и типы специфичных сообщений представляются в табличном виде, в виде листа Excel, а также в виде XML-схемы.

G.3.7.2 Описания столбцов типа сообщения в листах Excel

Пиктограмма обозначает гиперссылку на описание столбцов типа сообщения в листе Excel. Фрагмент представления типа сообщения в листе Excel показан на рисунке G.26. Описание столбцов этого представления приведено в таблице G.15.

No	Element Name (Link to tabular view)	Card	Mand	Conf	Rim Source	of Message Element Type	Src	Domain	CS
	Person Event Activate	**Common message for PRPA_HD101001UV01**							
1	**IdentifiedPerson**	0..1			RoleHeir	IdentifiedPerson	N		
2	classCode	1..1	M	R	Role	CS	D	IDENT	CNE
3	id	1..*	M	R	Role	SET<II>	D		
4	addr	0..*			Role	BAG<AD>	D		
5	telecom	0..*			Role	BAG<TEL>	D		
6	statusCode	1..1	M	R	Role	CS	D	RoleStatus	CNE
7	effectiveTime	0..1			Role	IVL<TS>	D		

Рисунок G.26 — Фрагмент описания типа сообщения

Таблица G.15 — Описание столбцов типа сообщения

Столбец	Описание
No (номер)	Номер строки. Строки нумеруются последовательно и определяют порядок, в котором данные будут сериализованы из модели R-MIM
Element Name (имя элемента)	Имя элемента в модели R-MIM. Это имя не обязательно соответствует значению в свойстве Property. Оно будет отличаться, если класс, ассоциация или роль клонированы и переименованы в процессе создания модели R-MIM
(row type) (тип строки)	Каждая строка представляет класс, ассоциацию или атрибут из модели R-MIM. Имена строк, соответствующих классам, выделены полужирным шрифтом; имена строк, соответствующих ассоциациям — курсивом; а имена строк, соответствующих атрибутам — обычным шрифтом
Card (кратность)	Кратность (Cardinality). Определяет минимальное и максимальное число экземпляров элемента сообщения
Mand (обязательность)	Обязательность. Допустимые значения: M (mandatory — обязательный) или пусто (не обязательный). Значение M в этом поле указывает, что передаваемый элемент должен содержать некоторые данные. Если данные не известны, то должно передаваться значение «неизвестное», «не полученное» и т. д. Значение M в данном столбце отличается от значения 1 в столбце кратности, поскольку указывает, что сообщение не может быть правильно разобрано, если не указано либо значение данного поля, либо его значение по умолчанию. Если значение по умолчанию не предусмотрено, то надо либо не передавать сообщение, либо задать значение этого поля. Значение M в этом столбце отличается и от значения R в столбце соответствия (Conformance), см. ниже
Conf (соответствие)	Соответствие. Допустимые значения: R (required — требуемое), NP (not permitted — запрещено) и пустое значение (не обязательное). Значение R (требуемое) указывает, что элементы сообщения должны присутствовать всякий раз, когда сообщение используется при взаимодействии. Если данные доступны, то элемент должен их содержать. Если же данные не доступны, может быть передано пустое значение. Значение NP (запрещено) означает, что этот элемент сообщения никогда не передается в сообщении данного типа. Пустое значение указывает, что соответствие этого элемента является предметом местных соглашений
RIM Source (источник в модели RIM)	Указывает класс, из которого получены атрибут или ассоциация
Of Message Element Type (тип элемента сообщения)	Определяет тип данных атрибутов или имя класса ассоциаций
SRC (источник)	Источник типа элемента сообщения. Допустимые значения: D (data type — тип данных), N (new — новый, элемент впервые определяется, начиная с текущей строки), U (use — использование, означает, что элемент, а не его значение, повторно используется из предыдущей строки иерархического описания сообщения HMD), C (CMET), I (Instance — экземпляр, приводит к повторному использованию элемента и его значения, определенных ранее в иерархическом описании сообщения HMD), и R (recursive — рекурсия на себя)
Domain (домен)	Спецификация словарного домена (Vocabulary Domain Specification). Представляет собой гиперссылку, по которой можно перейти на спецификацию словарного домена этого элемента
CS (квалификатор расширяемости)	Квалификатор расширяемости (Coding Strength). Допустимые значения: CWE (coded with extensions — кодированный с расширениями, то есть домен значений может быть расширен в соответствии с местными нуждами реализации) и CNE (coded no extensions — кодированный без расширений, то есть домен значений не может быть расширен)

Окончание таблицы G.15

Столбец	Описание
Abstract (абстрактный)	Булевское значение, присваиваемое классам или ассоциациям, входящими в иерархию генерализации-специализации, из которых может быть сделан выбор в иерархическом описании сообщения HMD. Если оно равно true (истинно), то данный тип не может использоваться в экземплярах сообщений
Nt (примечание)	Примечание (Note). Если указано, представляет собой просто текстовый комментарий, сделанный комитетом

G.3.7.3 Типы сообщений. Табличное представление

Пиктограмма обозначает гиперссылку на табличное представление типа сообщения. Фрагмент табличного представления типа сообщения New Person (новое лицо) показано на рисунке G.27.

PRPA_HD101001 **New Person**	This HMD defines the message used to report that a new record was added to a person registry. This R_MIM defines the message used to report that a new record was added to a person registry. (Link to Excel View) Derived from RMIM: PRPA_RM101001
IdentifiedPerson	The primary record for the focal person in a person registry
classCode [1..1] (M) Role (CS) {CNE:**IDENT**, fixed value= "IDENT"}	Structural attribute; this is an "identified" entity
id [1..*] (M) Role (SET<II>)	One or more identifiers designated for the focal person by this person registry. Note, this is a mandatory attribute; it makes no sense to have a record in an identified person registry that does not have an identifier.
addr [0..*] Role (BAG<AD>)	Address(es) for corresponding with the focal person in the context of this person registry
telecom [0..*] Role (BAG<TEL>)	Telecommunication address(es) for communicating with the focal person in the context of this person registry

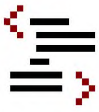
Рисунок G.27 — Фрагмент табличного описания сообщения New Person

Табличное представление общего сообщения и отдельных типов сообщений является сжатым представлением определения типа сообщения, помещающимся на одной печатной странице.

G.3.7.4 Представление в виде XML-схемы

Пиктограмма обозначает гиперссылку на XML-схему типа сообщения, используемую для проверки правильности всех XML-сообщений, соответствующих данному типу сообщения. Предусмотрена для отдельных типов сообщений.

G.3.7.5 Пример



Пиктограмма обозначает гиперссылку на пример сообщения. Примеры предусмотрены только для базовых иерархических описаний сообщений HMD. Они представляют собой XML-сообщения, произведенные из схемы и определения типа сообщения.

G.3.8 Взаимодействия

Взаимодействия (Interactions) лежат в основе передачи сообщений. Приведем формальное определение взаимодействия:

Уникальная ассоциация между специфичным типом сообщения (передачей информации), инициирующим событием, запускающим или «инициирующим» передачу, и ролями приложений, отправляющих и получающих сообщения этого типа. Представляет собой уникальную одностороннюю передачу информации.

Описание отдельного взаимодействия дает точный ответ на следующие вопросы:

1 какой тип компонента системы отправляет конкретный тип сообщения;

2 какому типу компонента системы-получателя передается сообщение этого типа;

3 как система определяет момент отправки конкретный тип сообщения;

4 что представляет собой этот тип сообщения.

Взаимодействия обсуждаются также в разделе, посвященном сценариям. Описание сценариев обычно сопровождается диаграммами взаимодействия. Как упоминалось в разделе «Сценарии», блоки в верхней части диаграммы взаимодействия и нисходящие от них вертикальные линии представляют роли приложений, то есть типы системных компонентов, отправляющих и принимающих конкретное сообщение. Горизонтальные линии на диаграмме взаимодействия являются собственно взаимодействиями и тем самым отвечают на два первых вопроса из числа приведенных выше.

Как указано в приведенном выше списке, каждое взаимодействие связано с ролью отправителя, ролью получателя (обе они идентифицированы на диаграмме взаимодействия), инициирующим событием и типом сообщения. В спецификациях Версии 3 для каждого взаимодействия указаны имя, идентификатор аретфакта, а также таблица, в которой перечислены роль отправителя, роль получателя, инициирующее событие, тип сообщения, тип события и тип обертки.

Тип события представляет собой тип инициирующего события. Допускаются следующие его значения: основан на взаимодействии, основан на переходе состояний, основан на действиях пользователя или не специфицирован. Для получения более подробной информации о типах инициирующего события обратитесь к подразделу «Инициирующие события» настоящего руководства по версии 3.

Будучи указан, тип обертки относится к обертке управляющего события. Дополнительными необязательными обертками являются обертки управляющего действия, содержащие служебную информацию, специфичную для предметной области и связанную с взаимодействием, инициировавшим данное сообщение (а именно, сообщение, к которому добавляется специфицированная обертка управляющего действия). Этот тип обертки не встречается в сообщениях, определенных в стандарте HL7, у которых нет дополнительного контекста, который требуется изменять динамически, или в сообщениях, содержащих команды, координирующие операции служб обработки сообщений. Текущее определение обертки управляющего действия затрагивает следующие две области:

1 запрос (query);

2 регистр (registry), например, изменение главного регистра пациентов.

На рисунке G.28 показано описание взаимодействие в том виде, как оно встречается в спецификации стандарта версии 3.

9.80 Combined Pharmacy Order Abort, Fulfillment Request - Community Tightly-coupled (PORX_IN002823)

Description

Structured Name: Combined Pharmacy Order Abort, Fulfillment Request - Community Tightly-coupled

An application asks another application to act on the discontinuation of a service request for a administration and dispensing of a drug or similar therapy. The applications are operating in an environment that is community-based (rather than institutional) and does NOT assume a common or shared repository of patient, provider and similar information.

Sending role	Combined Pharmacy Order Lifetime Placer Community Tightly-coupled	PORX_AR002133
Receiving role	Combined Pharmacy Order Lifetime Fulfiller Community Tightly-coupled	PORX_AR002143
Trigger role	Combined Pharmacy Order Abort, Fulfillment Request	PORX_TE002820
Message type		
Interaction Type	Request For Action	
Wrapper Type		

Receiver Responsibilities

Reason	Interaction	Trigger Event
Indicates that the request has been accepted and provides details of the plan to fulfill it.	PORX_TE003830	PORX_IN003833
Indicates that the request has been refused.	PORX_TE002840	PORX_IN002843

Рисунок G.28 — Пример описания взаимодействия

G.3.9 Обертки сообщения версии 3 и инфраструктура

Стандарт HL7, версия 3 обеспечивает существенный уровень функциональности, предусматривая использование «конвертов» (envelope), в которые вкладываются сообщения при передаче от отправителя к получателю. В стандарте HL7 они называются обертками (wrappers). Внутрь обертки помещается содержание, взятое из предметной области. Все обертки описываются таким же способом, как и содержание сообщения, а именно, с помощью определения классов объектов и их отношений. В дальнейшем эти спецификации могут использоваться для создания XML-схем или синтаксиса, определенного в другой технологии реализации ITS и описывающего формат передачи данных. Рабочая группа HL7 разрабатывает инструментальные средства, позволяющие автоматически выбирать нужный тип обертки в процесс генерации. Пакет документов по стандартизации инфраструктуры, представленный на голосование, состоит из следующих четырех отдельных предметных областей:

1 Инфраструктура передачи сообщений.

2 Управляющее действие сообщения.

3 Инфраструктура запроса.

4 Инфраструктура нормативно-справочных файлов.

Это позволяет применить общих подход к разработке этих стандартов, используемых при определении содержания сообщений в различных прикладных областях. Применение обычной структуры описания сообщения особенно полезно тем, что оно позволяет определять роли (приложений) и взаимодействия, что особенно применимо к функциональности более сложных оберток. Рабочая группа HL7 предпочитает определять такой богатый набор функций в рамках нескольких стандартов HL7, поскольку де-факто не существует способа предложить эти функции в единственном утверждаемом стандарте. В настоящее время Рабочая группа HL7 готовит спецификации транспорта, описывающие способы передачи сообщений стандартов HL7 с помощью таких базовых технологий транспорта, как eBXML или веб-службы.

G.3.10 Обертка передачи сообщений версии 3 (голосование: предметная область инфраструктуры)

Обертка передачи сообщений требуется для всех сообщений, определенных в версии 3. Она содержит большое число необязательных полей, поэтому накладные расходы на ее использование существенно зависят от контекста передачи. Техническим комитетам или сферам действия, выполняющим уточнение обертки передачи, необходима гарантия, что найдутся такие инструментальные средства, которые смогут использовать не только стандартную обертку, но и модифицированные ими обертки. Например, при использовании технологии реализации XML ITS на выходе должна получаться XML-схема, описывающая содержание сообщения, обертку передачи и, возможно, обертку управляющего действия.

С оберткой связано несколько служб, используемых при необходимости. Для обеспечения их функций определены три сообщения, с каждым из которых связана одна модель R-MIM и одно иерархическое описание сообщения HMD. Два из них представляют собой сообщения подтверждения с весьма разными ролями. Первое сообще-

ние используется для обеспечения надежной доставки другого сообщения по сети. Однако все чаще для передачи сообщений, определенных в стандарте HL7, используются внешние службы транспорта, не требующие использования подтверждений приема, предусмотренных в стандарте HL7. То же относится и к необязательному протоколу последовательной нумерации сообщений. Он является важным компонентом, предназначенным для обеспечения надежной доставки сообщений на сетевом уровне. Однако он опять-таки не требуется при использовании такого надежного внешнего транспорта, как те три транспортных профиля стандартов HL7, которые в настоящее время являются тестируемыми проектами стандартов (Draft Standard for Trial Use — DSTU). С подтверждением прикладной обработки ситуация совершенно иная. Оно инициируется только в том случае, когда оно включено во взаимодействие для определенного инициирующего события. Оно может нести содержание предметной области и поэтому сочетает общую функциональность подтверждения прикладной обработки со специфичным ответом, описанным для предмета данного взаимодействия.

Все сообщения в предметной области передачи сообщений имеют обязательные атрибуты, описывающие системы-отправители и получатели, участвующие в передаче (DeviceID). Дополнительная информация об отправителе и получателе содержится в других классах, но использование обязательных атрибутов гарантирует, что транспортный протокол, например, описанный в документе ebXML DSTU, может определить процедуру копирования атрибута, позволяющую значениям атрибута DeviceID храниться также в обертках протокола ebXML. В последнем абзаце настоящего подраздела приведены сходства и различия трех спецификаций транспортных протоколов DSTU.

Обертка передачи предоставляет дополнительный набор функций для организации периодического опроса. Эти функции предназначены для совместимости с более ранними выпусками стандарта версии 2. Вероятно, малая заинтересованность разработчиков к применению опроса не связан с конкретными унаследованными проблемами, а именно:

1 Отправка ответа на опрос с сообщением.

2 Отправка подтверждения приема/опрос следующего сообщения.

G.3.11 Обертка управляющего действия (голосование: предметная область инфраструктуры)

Дополнительные возможности функциональности обеспечиваются оберткой управляющего действия, используемой в тех случаях, когда информация об взаимодействии, инициировавшем передачу сообщения, должна быть передана получателю вместе с сообщением. В некоторых случаях эти взаимодействия будут полностью описаны моделью взаимодействия и потому не будет необходимости включать такую информацию в сообщение. Двумя конкретными областями применения обертки управляющего действия служат сеансы запросов и взаимодействия регистров (осуществляемые, например, при ведении главного регистра пациентов). Технические комитеты принимают индивидуальные решения о том, надо ли включать обертку управляющего действия в конкретное сообщение. Определение общей обертки включено в голосуемый документ инфраструктуры. Технические комитеты могут самостоятельно выполнять уточнение обертки управляющего действия, необходимое для их нужд. Во вновь разрабатываемых инструментальных средствах может понадобиться возможность включения в сообщения оберток, специфичных для предметной области. Существующие инструментальные средства не обладают подобной возможностью. Эти обертки описаны в пакете документов, представляемых на голосование Рабочей группе HL7. Для них предусмотрены три основных группировки, описанные как отдельные предметные области:

1 Инфраструктура управляющих действий сообщений (MCAI — Message Control Act Infrastructure) — описывает само управляющее действие (Control Act).

2 Инфраструктура запросов (QUQI — Query Infrastructure) — описывает методы использования оберток для поддержки запросов.

3 Инфраструктура нормативно-справочных файлов (MFMI — Master File Infrastructure).

G.3.11.1 Инфраструктура управляющих действий сообщений

Инфраструктура управляющих действий сообщений MCAI описывает детали управляющих действий. В основном она используется для приложений, работающих с запросами и нормативно-справочными файлами. Более подробно этом описано в двух следующих подразделах.

G.3.12.2 Инфраструктура запросов

Цель разработки этой инфраструктуры состоит в определении общей структуры запросов и ответов на запросы. Для этого наряду с соответствующими ролями и взаимодействиями определены три сообщения:

1 Общее активирование запроса и продолжения запроса (Query General Activate Query Continue).

2 Общий ответ на запрос (Query General Response).

3 Спецификация запроса с параметрами (Query by Parameter specification).

G.3.11.3 Инфраструктура нормативно-справочных файлов

Типичным приложением этой инфраструктуры служит главный регистр пациентов (Master Patient Index — MPI). Опять-таки, задача состоит в описании общего набора сообщений, позволяющих управлять базой данных любого типа пациента или медицинских работников, либо аналогичной базой данных. В этих функциональных областях нет специфичных сообщений, обеспечиваемых требуемую функциональность. Управляющее действие используется для описания необходимых действий, набора стандартных инициирующих событий, а также соответствующих диаграмм перехода состояний.

G.4 Ссылки

Метод разработки сообщений (Message Development Framework)

HL7 Version 3 Standard: Message Development Framework, HL7 Version 3 Taskforce

Стандарт HL7 версии 3: административное управление (HL7 Version 3 Standard: Administrative Management)

HL7 Version 3 Standard: Administrative Management, HL7 Version 3 Taskforce

Стандарт HL7 версии 3: управление инфраструктурой (HL7 Version 3 Standard: Infrastructure Management)

HL7 Version 3 Standard: Infrastructure Management, HL7 Version 3 Taskforce

Стандарт HL7 версии 3: управление здоровьем и медицинской помощью (HL7 Version 3 Standard: Health and Clinical Management)

HL7 Version 3 Standard: Health and Clinical Management, HL7 Version 3 Taskforce

Стандарт HL7 версии 3: словарь (HL7 Version 3 Standard: Vocabulary)

HL7 Version 3 Standard: Vocabulary, HL7 Version 3 Taskforce

Принципы разработки стандарта HL7 версии 3 (HL7 Version 3 Statement of Principles)

HL7 Version 3 Statement of Principles, HL7 Version 3 Taskforce

Метод разработки сообщений в сфере здравоохранения (Method for the Development of Healthcare Messages)

Method for the Development of Healthcare Messages, CEN TC 251

Объектно-ориентированная разработка программного обеспечения (Object Oriented Software Engineering)

Object Oriented Software Engineering, Ivar Jacobson, et al.

Объектно-ориентированный анализ (Object Oriented Analysis)

Object Oriented Analysis, Peter Coad & Ed Yourdon

Эталонная модель открытой передачи данных (Open EDI Reference Model)

Open EDI Reference Model, ИСО/IEC CD 14662

Унифицированный метод, версия 0.8 (Unified Method, Version 0.8)

Unified Method, Version 0.8, Grady Booch & James Rumbaugh

Краткий обзор языка UML (UML Distilled)

UML Distilled, Martin Fowler with Kendall Scott

Приложение ДА
(справочное)

Сведения о соответствии ссылочных международных стандартов национальным стандартам Российской Федерации

Таблица ДА

Обозначение ссылочного международного стандарта	Степень соответствия	Обозначение и наименование соответствующего национального стандарта
ИСО/HL7 21731:2006	—	*
HL7 2008	—	*
* Соответствующий национальный стандарт отсутствует. До его утверждения рекомендуется использовать перевод на русский язык данного международного стандарта.		

УДК 004:61 ОКС 35.240.80 П85 ОКСТУ 4002

Ключевые слова: здравоохранение, информатизация здоровья, электронная передача документов

Редактор *А.Ф. Колчин*
Компьютерная верстка *Е.А. Кондрашовой*

Сдано в набор 03.06.2016. Подписано в печать 23.06.2016. Формат 60×84⅛. Гарнитура Ариал.
Усл. печ. л. 67,43.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

Издано во ФГУП «СТАНДАРТИНФОРМ», 123995 Москва, Гранатный пер., 4.
www.gostinfo.ru info@gostinfo.ru