ФЕДЕРАЛЬНОЕ АГЕНТСТВО
ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р 56272— 2014 /ISO/TS 15926-8:2011

Системы промышленной автоматизации и интеграция

ИНТЕГРАЦИЯ ДАННЫХ ЖИЗНЕННОГО ЦИКЛА ПЕРЕРАБАТЫВАЮЩИХ ПРЕДПРИЯТИЙ, ВКЛЮЧАЯ НЕФТЯНЫЕ И ГАЗОВЫЕ ПРОИЗВОДСТВЕННЫЕ ПРЕДПРИЯТИЯ

Часть 8
Практические методы интеграции распределенных систем: практическая реализация сетевого языка онтологий (OWL)

ISO/TS 15926-8:2011
Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities
Part 8:
Implementation methods for the integration of distributed systems: Web Ontology Language (OWL) implementation
(IDT)

Издание официальное

Москва
Стандартинформ
2015

gost.e-ecolog.ru

Предисловие

1 ПОДГОТОВЛЕН ООО «НИИ экономики связи и информатики «Интерэкомс» (ООО «НИИ «Интерэкомс») на основе собственного аутентичного перевода на русский язык международного документа, указанного в пункте 4

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 100 «Стратегический и инновационный менеджмент»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 26 ноября 2014 г. № 1858- ст

4 Настоящий стандарт идентичен международному документу ISO/TS 15926–8:2011 «Системы промышленной автоматизации и интеграция. Интеграция данных жизненного цикла перерабатывающих предприятий, включая нефтяные и газовые производственные предприятия. Часть 8. Практические методы интеграции распределенных систем: практическая реализация сетевого языка онтологий (OWL)» (ISO/TS 15926-8:2011 «Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities — Part 8: Implementation methods for the integration of distributed systems: Web Ontology Language (OWL) implementation»).

5 ВВЕДЕН ВПЕРВЫЕ

Правила применения настоящего стандарта установлены в ГОСТ Р 1.0—2012 (раздел 8). Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок – в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске информационного указателя «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 (gost.ru)

© Стандартинформ, 2015

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 Термины, определения и сокращения ... 1
3 Фундаментальные понятия и допущения ... 5
4 Модель данных ИСО 15926-2 ... 10
5 Справочные данные ИСО/ТС 15926-4 ... 10
6 Спецификация шаблона языка OWL ... 10
7 Шаблоны ... 10
8 Информационные модели объектов и специализированные шаблоны ... 13
Приложение А (обязательное) Регистрация информационного объекта ... 16
Приложение В (обязательное) Шаблоны как N-арные отношения в среде RDF/OWL ... 17
Приложение С (обязательное) Правила использования языка OWL ... 20
Приложение D (обязательное) Компьютерно-представимый листинг ... 24
Приложение Е (справочное) Паттерны (образцы) онтологии справочных данных ... 25
Приложение F (обязательное) Метаданные ... 26
Приложение G (справочное) Утверждения и терминологические компоненты ... 28
Приложение Н (справочное) Примеры ... 29
Библиография ... 38

Введение

Комплекс международных стандартов ИСО 15926 предназначен для представления информации о жизненном цикле перерабатывающих предприятий, включая нефтяные и газовые производственные предприятия. Это представление определяется общей концептуальной моделью данных, которая является основой для совместного использования баз данных и хранилищ информации. Такая модель предназначена для применения совместно со справочными данными, например стандартными экземплярами, которые представляют информацию, общую для ряда пользователей, технологических установок или того и другого вместе. Поддержка деятельности на протяжении жизненного цикла зависит от использования необходимых справочных данных совместно с моделью данных.

Комплекс международных стандартов ИСО 15926 состоит из нескольких частей. Настоящий стандарт содержит описание практической реализации сетевого языка онтологий (OWL) с помощью семантических сетевых технологий консорциума всемирной сети (W3C).

Настоящий стандарт использует трансляцию классов ИСО/ТС 15926-4, справочных данных и методологии использования шаблонов (в соответствии с ИСО/ТС 15926-7) для практической реализации среды описания ресурса (RDF) и сетевого языка онтологий (OWL), необходимых для моделирования данных, обеспечения интеграции и взаимодействия. Настоящий стандарт служит базисом для рассматриваемой инфраструктуры интеграции, взаимодействия и методов испытаний.

Под шаблоном в настоящем стандарте понимается специальная структура данных, установленная в комплексе международных стандартов ИСО 15926. В настоящем стандарте шаблоны рассматриваются с нескольких точек зрения.

Во-первых, шаблон — это предикат, утверждение которого формулируется в качестве аксиомы. Шаблон формулирует в себе некое утверждение о модели данных и каждый его экземпляр — истинное утверждение в рамках используемой модели данных.

Во-вторых, шаблон — это способ передачи данных. Когда создается шаблон, четко оговаривается семантика каждой его свободной переменной, а требование к структуре модели данных, в рамках которой применяется шаблон, оформляется в виде формулы логики первого порядка. Предполагается, что справочные данные при обмене не передаются — к ним имеют доступ и получатель и отправитель, то есть они пользуются общим «словарем» (хотя понятие «справочные данные» намного шире, чем понятие «словарь»). В основном передаются экземпляры шаблонов (template instance), поскольку их семантика задана наперед, получатель и отправитель всегда знают, какой смысл вложен в переданный набор данных. Кроме того, в случае создания хранилища данных об индивидах в виде триплетов (реализация фасадов (от английского слова façade) как способа представления информации об индивидах на основе справочных данных) к нему возможен доступ при помощи SPARQL запросов, тогда вопрос получения необходимых данных сводится к построению соответствующего запроса. Таким образом, если шаблон соответствует моделям данных передающей и принимающей сторон, то достаточно передать лишь блок данных, идентифицирующий шаблон и содержащий в себе заполненные свободные переменные. Тогда принимающая сторона, зная его семантику, соотнесет переданные значения переменных со своей моделью данных необходимым образом.

В-третьих, шаблоны являются справочными данными, их спецификация хранится в библиотеке справочных данных RDL при помощи специальных структур данных.

Поскольку в соответствии с ИСО 15926 шаблон является предикатом логики первого порядка, удовлетворяющим аксиоматике модели данных настоящего стандарта, то, заполняя соответствующими значениями переменные такого предиката, мы получаем утверждения об объектах с фиксированной семантикой — экземплярах шаблона. Шаблоны позволяют как генерировать справочные данные (для создания однородной группы утверждений об элементах библиотеки справочных данных RDL достаточно применить шаблон необходимое количество раз с соответствующими значениями переменных), так и создавать связи между библиотеками справочных данных RDL и моделью данных приложения (например, PLMприложения) — так называемый маппинг (mapping).

Настоящий стандарт распространяется на:

— метод трансляции (отображения) классов ИСО/ТС 15926-4 в среду описания ресурсов и сетевого языка RDF/OWL;

— метод трансляции (отображения) шаблонов ИСО/ТС 15926-7 в среду описания ресурсов и сетевого языка RDF/OWL;

— структурные компоненты специализированных шаблонов;

— порядок использования моделей информационных объектов;

— структурные компоненты метаданных.

От пользователей настоящего стандарта требуется понимание концептуальных моделей данных и ИСО/ТС 15926-7.

Целевая аудитория настоящего стандарта:

— технические директора и руководители проектов, определяющие, подходит ли ИСО 15926 для конкретных нужд ведения бизнеса;

— разработчики программного обеспечения интерфейса связи между старыми системами и системами, соответствующими ИСО 15926;

— разработчики программного обеспечения, соответствующего ИСО 15926 и используемого для интеграции данных.

В настоящем стандарте одни и те же объекты (в оригинале именуемые английским термином «thing») могут относиться к:

1) сущностям реального мира,

2) представлению сущностей реального мира на языке EXPRESS,

3) представлению сущности в среде описания ресурса или на расширенном языке разметки (RDF/XML). Смысл слова (фразы) устанавливается с помощью типографских обозначений:

— если слово или фраза приведены обычным шрифтом, то это просто сущность реального мира;

Пример 1 — охлаждающий водяной насос.

— если слово или фраза приведены **жирным_шрифтом_с_нижним_подчеркиванием**, то сущность является представлением в среде EXPRESS для модели данных в соответствии с ИСО 15926-2;

Пример 2 — **class_of_inanimate_physical_object** ***(класс неодушевленных физических объектов).***

— если слово или фраза приведены **жирнымШрифтомБезПробелов**, то это аксиома подтипа в соответствии с ИСО/ТС 15926-7;

Пример 3 — **ClassOfInanimatePhysicalObject.**

— если слово приведено *курсивом,* то это «родной» тип рассматриваемой сущности RDF/RDFS/OWL.

Пример 4 — *rdfs:subClassOf.*

Используемые идентификаторы в данных примерах являются вымышленными.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Системы промышленной автоматизации и интеграция
ИНТЕГРАЦИЯ ДАННЫХ ЖИЗНЕННОГО ЦИКЛА ПЕРЕРАБАТЫВАЮЩИХ ПРЕДПРИЯТИЙ, ВКЛЮЧАЯ НЕФТЯНЫЕ И ГАЗОВЫЕ ПРОИЗВОДСТВЕННЫЕ ПРЕДПРИЯТИЯ
Часть 8
Практические методы интеграции распределенных систем: практическая реализация сетевого языка онтологий (OWL)

Industrial automation systems and integration. Integration of life-cycle data for process plants including oil and gas production facilities. Part 8.
Implementation methods for the integration of distributed systems.
Web Ontology Language (OWL) implementation

Дата введения — 2016—01—01

1 Область применения

В настоящем стандарте определены вопросы практической реализации методов интеграции, совместного использования, обмена и передачи информации о жизненном цикле перерабатывающих предприятий, полученной путем использования модели данных ИСО 15926-2 и методологии использования шаблонов ИСО/ТС 15926-7.

Настоящий стандарт распространяется на:

— определение правил использования среды описания ресурса RDF и сетевого языка онтологий OWL в контексте настоящего стандарта;

— отображение модели данных ИСО 15926-2 из формата EXPRESS на сетевой язык OWL-2;

— определение методологии создания онтологий OWL для справочных данных в соответствии с ИСО/ТС 15926-4;

— определение онтологий OWL с использованием базовых шаблонов и исходного множества основных шаблонов;

— определение методологии создания онтологий OWL для «специализированного шаблона» (см. 2.1.32), устанавливающего типы информации для любого экземпляра (реализации) индивидуального объекта **possible_individual** в течение его жизненного цикла.

Настоящий стандарт не распространяется на:

— особый тип языка правил, используемый для применения логики первого порядка;

— процедуру принятия решения о том, что хранение (обмен) данных производится либо с помощью верхних (поднятых) данных (lifted data), либо только с помощью нижних (опущенных) экземпляров шаблонов (lowered template instances) и объектов.

Примечание — Находится к компетенции бизнеса.

2 Термины, определения и сокращения

2.1 Термины и определения

В настоящем документе используются следующие термины и определения.

2.1.1 **базовый шаблон** (base template): Шаблон, содержащий только типы сущности в расширении соответствующей аксиомы шаблона.

[ИСО/ТС 15926-7:2011, определение 2.1.1]

2.1.2 **класс** (class): Категория или классификация элементов, выделенных по одному или нескольким критериям для последующего включения или исключения.

Примечание 1 — Класс не обязательно должен состоять из известных членов класса (сущностей, которые удовлетворяют критериям вхождения в данный класс).

Примечание 2 — Из-за пространственно-временной парадигмы, используемой для определения индивидуальных объектов в настоящем стандарте, не все классы являются хорошо обособленными множествами. Пояснение приводится в ИСО 15926-2.

Примечание 3 — Адаптировано из ИСО 15926-1:2004, определение 3.1.1.

Издание официальное

2.1.3 **основной класс** (core class): Класс, отражающий разделения индивидов и отношений в соответствии с общеупотребительными терминами, применяемыми в обычном языке.

Примечание — Условия включения в класс часто не определяются формально: понимание класса может иллюстрироваться примером.

Пример — Труба, пол, насос, лампа накаливания – основные классы.

[ИСО 15926-1:2004, определение 3.1.4]

2.1.4 **основной шаблон** (core template): Шаблон библиотеки справочных данных RDL, для которого все элементы справочных данных в расширении аксиомы шаблона являются основными классами.

[ИСО 15926-7:2011, определение 2.1.6]

2.1.5 **банк данных** (data store): Компьютерная система, обеспечивающая хранение данных для обращения к ним в будущем.

[ИСО 15926-1:2004, определение 3.1.6]

2.1.6 **тип данных** (data type): Область значений.

[ИСО 10303-11:2004, определение 3.3.5]

2.1.7 **хранилище данных** (data warehouse): Банк данных, в котором смежные данные объединяются для обеспечения интегрированного множества данных без дублирования или избыточности с поддержкой множества различных прикладных вариантов.

[ИСО 15926-1:2004, определение 3.1.7]

2.1.8 **сущность** (entity): Класс информации, определенный общими свойствами.

[ИСО 10303-11:2004, определение 3.3.6]

2.1.9 **тип данных сущности** (entity data type): Представление сущности.

Примечание 1 — Тип данных сущности устанавливает область значений, определенную обычными атрибутами и ограничениями.

Примечание 2 — В настоящем стандарте типы данных сущности определяются в соответствии с требованиям модели данных ИСО 15926-2.

Примечание 3 — Адаптировано из ИСО 10303-11:2004, определение 3.3.7.

2.1.10 **экземпляр сущности** (entity instance): Именованный блок данных, представляющий собой блок информации внутри класса, определенного некоторой сущностью.

Примечание 1 — Принадлежит области, установленной типом данных сущности.

Примечание 2 — Адаптировано из ИСО 10303-11:2004, определение 3.3.8.

2.1.11 **логика первого порядка** (first-order logic): Формализованные суждения, в которых каждое предложение или высказывание (утверждение) подразделяются на субъект (подлежащее) и предикат (сказуемое).

Примечание 1 — Сказуемое модифицирует (определяет) свойства субъекта (подлежащего). В логике первого порядка предикат (сказуемое) может относиться только к одному субъекту (подлежащему).

Примечание 2 — Логику первого порядка также называют исчислением сказуемого первого порядка или функциональным исчислением первого порядка.

[ИСО 15926-7:2011, определение 2.1.13]

2.1.12 **индивид; индивидуальный объект; возможный индивид** (individual, possible individual): Объект реального мира, существующий в пространстве и времени.

Примечание 1 — В настоящем контексте, существование имеет место внутри нашего реального мира или внутри некоего воображаемого мира. Этот мир включает фактические, гипотетические, планируемые, ожидаемые или требуемые индивидуальные объекты.

Пример — Насос с серийным номером ABC 123, электростанция Battersea, сэр Joseph Whitworth и предприятие Starship Enterprise – это примеры индивидуальных объектов.

Примечание 2 — Адаптировано из ИСО 15926-2:2003, определение 3.1.6.

2.1.13 **шаблон индивидуального объекта** (individual template): Шаблон утверждений об индивидуальном объекте.

[ИСО/ТС 15926-7:2011, определение 2.1.14]

2.1.14 **экземпляр** (instance): Именованное значение.

[ИСО 10303-11:2004, определение 3.3.10]

2.1.15 **интероперабельность** (interoperability): Способность двух и более систем (компьютеров, устройств связи, сетей, программного обеспечения, операционных систем, приложений и других компонентов информационных технологий) взаимодействовать друг с другом и обмениваться информацией предписанным методом для получения предсказуемого результата.

2.1.16 **информация о жизненном цикле** (life-cycle information): Информация (сведения) об объекте **possible_individual** (2.1.12), собранная в некоторый момент времени в течение жизненного цикла конкретного индивидуального объекта (индивида).

Примечание — Адаптировано из ИСО/ТС 15926-7:2011, определение 2.1.17.

2.1.17 **Манчестер-синтаксис** (Manchester syntax): Удобный для пользователя компактный синтаксис онтологий OWL 2. Данный синтаксис является фреймовым в отличие от прочих аксиомных синтаксисов языка OWL 2.

Примечание — См. [21].

2.1.18 **метаданные** (metadata): Данные, которые определяют и описывают другие данные.

[ИСО/МЭК 11179-1:2004, определение 3.2.16]

2.1.19 **нотация N-triple** (N-triple): Линейный ровный текстовый формат кодирования графов RDF.

2.1.20 **информационная модель объекта** (object information model; OIM): Онтология классов и отношений, для характеризации которой выделен особый класс.

Примечание — В терминологии описательной логики — это сущность **TBox** (см. приложение G).

2.1.21 **онтология** (ontology): <наука о компьютерах и информации> — формальное представление множества понятий в рамках некоторой области, а также отношения между этими понятиями.

Примечание 1 — Онтологии обычно используются для обоснования свойств некоторой области, но могут использоваться также и для определения этой области.

Примечание 2 — Онтологии обычно выражаются на логическом языке. Но это не требование, а отражение необходимости обоснования функциональности. Здесь могут использоваться отношения, классы, свойства, экземпляры (реализации) и аксиомы.

Примечание 3 — К онтологии может относиться подробная согласованная спецификация концепций области применения, не зависимая от любого использования этих концепций.

2.1.22 **«родной» язык OWL** (OWL native): Стиль моделирования, в котором отношение выражается как сказуемое среды описания ресурса RDF.

2.1.23 **трамбование**, **паннинг** (punning): Объявление класса и индивидуального объекта с одинаковыми идентификаторами с целью их использования в различных конструкциях модели, которые было бы неправильно использовать в отношении одного объекта.

Примечание — Паннинг также можно использовать для двух объявлений свойств с различными типами.

2.1.24 **граф RDF** (RDF graph): Структура графов, образованная множеством троек (триплетов) среды описания ресурса RDF.

2.1.25 **схема RDF** (RDF schema): Язык для описания словарей RDF, являющихся семантическим расширением RDF, предоставляющими механизм описания групп связанных ресурсов и отношений между данными ресурсами.

2.1.26 **формат RDF/XML** (RDF/XML): Формат синтаксиса расширяемого языка разметки (XML) для среды описания ресурсов RDF в соответствии с рекомендациями W3C «Спецификация синтаксиса RDF/XML».

2.1.27 **справочные данные** (reference data): Данные жизненного цикла перерабатывающих предприятий, предоставляющие информацию о классе или об отдельных его элементах, которые являются типовыми для большей части оборудования или представляют интерес для многих пользователей.

[ИСО 15926-1:2004, определение 3.1.18]

2.1.28 **библиотека справочных данных; RDL** (reference data library; RDL): Управляемый набор справочных данных.

[ИСО 15926-1:2004, определение 3.1.19]

Примечание — В настоящем стандарте, понятия «RDL» и «онтология» взаимозаменяемы.

2.1.29 **воплощение** (reification): Стиль моделирования, в котором отношение выражается как класс субъектов.

Пример — Отношение Employed-by (принят на работу) воплощается объектом Employment (прием на работу), соединенным с объектами Employee (служащий) и Organization (организация). Смысл данного отношения (с определенным количеством элементов с обеих сторон) заключается в том, что "количество служащих организации должно быть больше или равно нулю". Воплощенный объект Employment может быть подлежащим в других отношениях, определяя их.

Примечание — Реляционные типы данных сущности ИСО 15926 — это все типы данных сущности, имеющей два атрибута, за исключением класса отношений **class_of_relationship**.

[ИСО/ТС 15926-7:2011, статья 2.1.21]

2.1.30 **сигнатура** (signature): Именованный, упорядоченный и напечатанный список ролей шаблона.

2.1.31 **точка доступа SPARQL** (SPARQL endpoint): Служба, удовлетворяющая требованиям протокола SPARQL и используемая (в соответствии с протоколом SPARQL W3C) для среды описания ресурса RDF (SPROT).

Примечание — В работе [23] «точка доступа» определена как «ассоциация между полностью определенной компоновкой интерфейса **InterfaceBinding** и сетевым адресом, определенным идентификатором URI [IETF RFC 2396], который можно использовать для связи с экземпляром сетевой услуги. Точка доступа указывает особое место обеспечения доступа к сетевой службе с помощью особого протокола и формата данных».

2.1.32 **специализированный шаблон** (specialized template): Множество утверждений об индивидуальных объектах (классах), являющееся подклассом основного шаблона или другого специализированного шаблона, и которое имеет одно или несколько ролевых ограничений.

2.1.33 **таксономия** (taxonomy): Набор управляемых словарных терминов, организованных в иерархическую структуру, где каждый термин участвует в одном или нескольких родительских/дочерних (широких/узких) отношениях с другими терминами данной таксономии.

Примечание — Таксономия это результат идентификации и структурирования знаний. Иерархическая классификация, которая помогает пользователям разобраться в процедуре группировки и категоризации формализованных знаний. Хорошая таксономия облегчает поиск и приобретение новых знаний

2.1.34 **шаблон** (template): Множество, включающее предикат логики первого порядка (для которого определение задается как аксиома), шаблонные подписи и расширение аксиомы шаблона.

[ИСО/ТС 15926-7:2011, определение 2.1.22]

2.1.35 **шаблон** (template): N-арное сказуемое, представленное в форме на языке OWL, в качестве класса с одним функциональным свойством *(ролью)* для каждой переменной.

2.1.36 **аксиома шаблона** (template axiom): Аксиома на языке шаблона, определяющая интерпретацию шаблонных высказываний (утверждений шаблона).

[ИСО/ТС 15926-7:2011, определение 2.1.25]

2.1.37 **экземпляр шаблона** (template instance): Упорядоченный список экземпляров сущности для которых шаблон является истинным.

Примечание 1 — В среде OWL экземпляр шаблона – это индивидуальный объект с ролевыми отношениями, инстанцирующими шаблон. Элементарное/фундаментальное утверждение позволяет заместить переменные шаблона индивидуальными объектами в среде OWL.

Примечание 2 — Адаптировано из ИСО/ТС 15926-7:2011, определение 2.1.26.

2.1.38 **язык шаблона** (template language): Аксиомы логики первого порядка, расширяющие модель данных ИСО 15926-2.

[ИСО/ТС 15926-7:2011, определение 2.1.27]

2.1.39 **роль шаблона** (template role): Поименованный и перенумерованный аргумент шаблона с требуемым типом, представленным как тип данных сущности, тип данных или класс справочных данных.

Примечание — Адаптировано из ИСО/ТС 15926-7:2011, определение 2.1.28.

2.1.40 **утверждение шаблона, шаблонное высказывание** (template statement): Утверждение (высказывание), сделанное путем задания значений (инстанцирования) ролей шаблона экземплярами сущностей.

[ИСО/ТС 15926-7:2011, определение 2.1.30]

2.1.41 **основная библиотека справочных данных RDL** (core RDL): Множество RDL, содержащее только основные классы и справочные индивидуальные объекты.

Примечание — Часть данного содержания нормализована.

2.1.42 **тройка (триплет); тройка среды описания ресурса RDF** (triple; RDF triple): Представление отношения между объектами или данными, которые оно соединяет.

Примечание — Тройка включает, по крайней мере:

— объект, называемый «субъект» (подлежащее);

— сказуемое (также называемое свойством), которое обозначает отношение между субъектом и объектом;

— объект или данные, называемые объектом.

2.1.43 **хранилище троек** (triple store): Хранилище данных, способное хранить **тройки** (2.1.42).

2.1.44 **значение** (value): Элемент (единица) данных.

[ИСО 10303-11:2004, определение 3.3.22]

2.2 Сокращения

DL — Описательная логика (description logic);

FOL — Логика первого порядка (first order logic);

OIM — Информационная модель объекта (object information model);

OWL — Сетевой язык онтологий (Web Ontology Language);

RDF — Среда описания ресурса (Resource Definition Framework);

RDFS — Схема RDF (RDF Schema);

RDL — Библиотека справочных данных (reference data library);

SPARQL — Протокол SPARQL и язык запросов среды описания ресурса RDF (SPARQL Protocol and RDF Query Language);

URI — Унифицированный идентификатор ресурса (Uniform Resource Identifier);

W3C — Консорциум всемирной сети (World Wide Web Consortium);

XML — Расширяемый язык разметки (eXtensible Markup Language).

3 Фундаментальные понятия и допущения

3.1 Общие положения

Модель данных ИСО 15926-2 является базовой и строго упорядоченной (нормализованной). Описание модели требует гибкости, оно является достаточно сложным. ИСО/ТС 15926-7 устанавливает шаблоны, являющиеся выражением предварительно определенных семантических блоков. Они обеспечивают оптимальный способ использование модели.

В настоящем стандарте указанные шаблоны преобразуются в конструктивы языка OWL, обеспечивающие хранение информации, интеграцию данных, обмен информации и способность к интероперабельности.

3.2 Шаблоны ИСО/ТС 15926-7 и правила логики первого порядка

Модель данных ИСО 15926-2 определяет фундаментальные понятия. Они представлены базовой, концептуальной моделью данных, являющейся основой для практической реализации коллективно используемых баз и хранилищ данных. Рассматриваемая модель данных используется вместе со справочными данными: либо это стандартные экземпляры, представляющие обычную информацию для нескольких пользователей, либо это технологические установки, либо то и другое. Поддержка указанной производственной деятельности с заданным жизненным циклом зависит от использования соответствующих справочных данных вместе с рассматриваемой моделью данных. В соответствии с ИСО/ТС 15926-7, они транслируются в логику первого порядка с типами данных сущности языка EXPRESS, представленными как унарные сказуемые, или с атрибутами языка EXPRESS, представленными как бинарные сказуемые.

Для настоящего стандарта, сигнатуры шаблонов являются базовыми структурирующими элементами. В настоящем стандарте, сигнатуры шаблонов транслируются на сетевой язык OWL со

ссылками на правила логики первого порядка.

3.3 Справочные данные ИСО/ТС 15926-4

ИСО/ТС 15926-4 устанавливает справочные данные, определяющие таксономию основных классов и используемые типы данных сущности ИСО 15926-2.

В настоящем стандарте справочные данные рассматриваются как постоянные термины используемых правил.

Примечание — В рамках настоящего подхода, связанного с использованием атрибутов, которые указывают на справочные данные, инстанцирующие типы данных сущности, рассматриваемые уравнения правил (выражения) упрощаются.

3.4 Таксономия, определенная пользователем

Пользователю часто нужно определять особые понятия в форме специализаций основных классов таксономии ИСО/ТС 15926-4.

Пример — «Насос модели 4НМ» № СР-834833 — это специализированный класс в каталоге поставщика корпорации XYZ Corporation. Тогда rdfs:subClassOf — это точка соответствия класса «Центробежные насосы» в основной библиотеке справочных данных RDL.

3.5 Шаблоны и специализированные шаблоны

Шаблоны (см. раздел 7) и специализированные шаблоны (см. раздел 8), определенные в настоящем стандарте и в специализациях пользователя, образуют «производные понятия» в соответствии с ИСО/ТС 15926-7.

3.6 Использование сетевого языка онтологий (OWL)

Настоящий стандарт основывается на использовании среды описания ресурса RDF, схемы RDF и сетевого языка онтологий OWL.

Приложение С устанавливает правила использования указанных языков в контексте настоящего стандарта.

Примечание — Среда описания ресурса RDF, схемы RDF и язык OWL позволяют представить одну и ту же семантику различными способами. Чтобы сделать коды более прозрачными и понятными, необходимы определенные правила. Это не препятствует подключению словарей RDF/OWL, использующих язык OWL, в различных контекстах.

3.7 Области имен

Система справочных данных ИСО 15926 с помощью языка OWL подразделяется на множество онтологий, соответствующих рассматриваемым областям имен.

Таблица 1 показывает префиксы областей имен с идентификаторами URI областей имен, используемыми в качестве примеров в настоящем стандарте. Данные идентификаторы установлены консорциумом всемирной сети W3C.

Таблица 1 — Унифицированные идентификаторы ресурса URI, установленные консорциумом всемирной сети

Префикс области имен	Идентификатор URI области имен	Описание
owl	http://gost.e-ecolog.ru	Язык OWL версия 2
rdfs	http://gost.e-ecolog.ru	Схема RDF
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#	Среда описания ресурса
xsd	http://gost.e-ecolog.ru	Схема XML

Таблица 2 показывает префиксы областей имен с идентификаторами URI областей имен, используемыми для примеров в настоящем стандарте.

Таблица 2 — Области имен, используемые в качестве примеров в настоящем стандарте

Префикс области имен	URI области имен	Описание
dm	gost.e-ecolog.ru gost.e-ecolog.ru a/data-model#	Нормативные классы языка OWL в соответствии с моделью данных ИСО 15926-2 и некоторыми расширениями
p7tm	gost.e-ecolog.ru l/tech/reference-data/p7tm#	Нормативные декларации языка OWL для шаблонов и ролей шаблонов (p7 обозначает ИСО/ТС 15926-7)
p7tpl	gost.e-ecolog.ru l/tech/reference-data/p7tpl#	Шаблоны, определенные ИСО/ТС 15926-7, протошаблоны и основные шаблоны "исходного множества". (p7 обозначает ИСО/ТС 15926-7).
meta	gost.e-ecolog.ru gost.e-ecolog.ru a/metadata#	Метаданные

Примечание 1 — Префиксы **dm, p7tm, p7tpl** и **meta** — это файлы, а не точки доступа SPARQL. Указанные декларации неизменны. Можно предположить, что указанные файлы могут быть локально кэшированы функционирующей системой. Указанные файлы имеются на компакт-диске CD-ROM для настоящего стандарта.

Примечание 2 — Следующая область имен (префиксы) с адресом **example.org** используется в качестве примера:

— **rdl -** gost.e-ecolog.ru — точка доступа с нормативными основными классами и справочными индивидуальными объектами, классами, претендующими на нормализацию, и специализированными шаблонами, известными как основная библиотека справочных данных **"Core RDL"**;

— **company1 -** gost.e-ecolog.ru — точка доступа с примерами экземпляров, специализированными шаблонами и классами (пример хранилища данных RDF компании).

Примечание 3 — Областям имен рассматриваемого примера присвоены адреса **example.org** и **example.com**, чтобы можно было ясно указать, что это только примеры, на которые нельзя ссылаться. Точно также, свойства *rdf:ID* и *rdf:about* — это тоже примеры, выраженные в удобочитаемом виде вместо буквенно-цифровых идентификаторов, которые обычно применяются на практике.

Пример 1 — В Манчестер-синтаксисе:

Prefix: xsd: <gost.e-ecolog.ru >
Prefix: owl: <gost.e-ecolog.ru >
Prefix: rdf: <gost.e-ecolog.ru >
Prefix: rdfs: gost.e-ecolog.ru

Prefix: dm: <gost.e-ecolog.ru >
Prefix: p7tm: <gost.e-ecolog.ru >
Prefix: p7tpl: <gost.e-ecolog.ru >
Prefix: meta: gost.e-ecolog.ru

Prefix: rdl: gost.e-ecolog.ru

Prefix: steplib: <gost.e-ecolog.ru >
Prefix: pca: gost.e-ecolog.ru

Prefix: company1: <gost.e-ecolog.ru >

Пример 2 — Если рассматриваются примеры кодов, то заголовок RDF опускается. Ниже представлен пример заголовка:

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
<!ENTITY xsd "gost.e-ecolog.ru                                    ">
<!ENTITY owl "gost.e-ecolog.ru                                    ">
<!ENTITY rdf "gost.e-ecolog.ru                                          ">
<!ENTITY rdfs "gost.e-ecolog.ru                                    ">
<!ENTITY dm "gost.e-ecolog.ru                                                                    ">
<!ENTITY p7tm "gost.e-ecolog.ru                                                                ">
<!ENTITY p7tpl "gost.e-ecolog.ru                                                               ">
<!ENTITY meta "gost.e-ecolog.ru                                                                  ">
<!ENTITY rdl "gost.e-ecolog.ru                  ">
<!ENTITY steplib "gost.e-ecolog.ru                        ">
<!ENTITY pca "gost.e-ecolog.ru                          ">
<!ENTITY company1 "gost.e-ecolog.ru                     ">
]>
<rdf:RDF xmlns="gost.e-ecolog.ru                          "
xml:base="gost.e-ecolog.ru                    m"
xmlns:owl="&owl;"
xmlns:xsd="&xsd;"
xmlns:rdf="&rdf;"
xmlns:rdfs="$rdfs;"
xmlns:dm="&dm;"
xmlns:p7tm="&p7tm;"
xmlns:p7tpl="&p7tpl;"
xmlns:meta="&meta;"
xmlns:rdl="$rdl;"
xmlns:steplib="&steplib;"
xmlns:pca="&pca;"
xmlns:company1="&company1;"
>
<owl:Ontology rdf:about="">
<rdfs:comment>This ontology provides instances for
various examples of specialized templates.
</rdfs:comment>
<owl:imports rdf:resource="gost.e-ecolog.ru                    >
        <owl:imports rdf:resource="gost.e-ecolog.ru                    org/iso/ts/15926/-8/ed-1/tech/reference-
    data/p7tpl.owl"/>
        <owl:imports rdf:resource="gost.e-ecolog.ru                    org/iso/ts/15926/-8/ed-1/tech/reference-
    data/metadata.owl"/>
</owl:Ontology>

<!-- RDF XML here -->

</rdf:RDF>
```

Указанные онтологии являются взаимозависимыми в том смысле, что одна онтология использует классы, определенные другой онтологией. Рисунок 1 показывает эти зависимости (стрелка между префиксами обозначает отношение импортирования **owl: imports**).

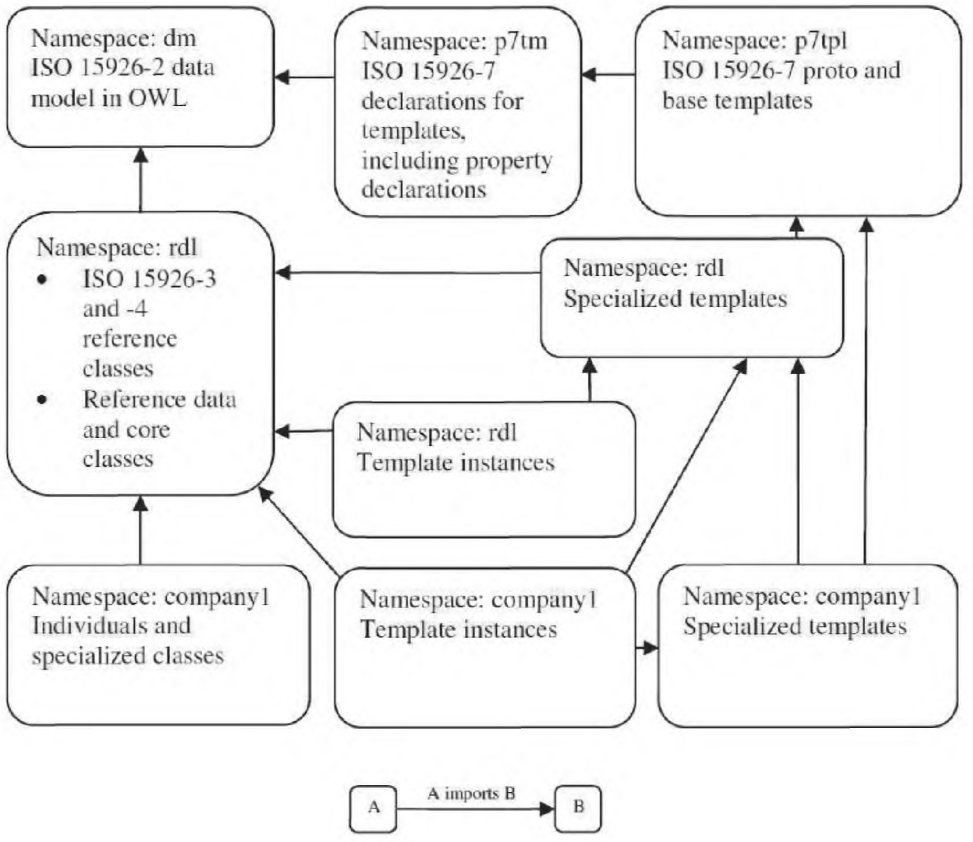

Рисунок 1 — Зависимости между онтологиями

Namespace: dm ISO 15926-2 data model in OWL	Область имен: **dm** Модель данных ИСО 15926-2 в среде OWL
Namespace: p7tm ISO 15926-7 declarations for templates, including property declarations	Область имен: **p7tm** Декларации ИСО 15926-7 для шаблонов, включающих декларации свойств
Namespace: p7tpl ISO 15926-7 proto and base templates	Область имен: **p7tpl** Протошаблоны и базовые шаблоны ИСО 15926-7
Namespace: rdl - ISO 15926-3 and 4 reference classes - Reference data and core classes	Область имен: **rdl** – Справочные классы ИСО 15926-3 и ИСО 15926-4); – Справочные данные и основные классы
Namespace: rdl Specialized templates	Область имен: **rdl** Специализированные шаблоны
Namespace: company1 Individuals and specialized classes	Область имен: **company1** Индивидуальные объекты и специализированные классы
Namespace: company1 Template instances	Область имен: **company1** Экземпляры шаблонов
Namespace: company1 Specialized templates	Область имен: **company1** Специализированные шаблоны
A imports B	A импортирует B

Примечание 4 — В настоящем стандарте декларации области имен модели OIM не рассматриваются. Модели OIM скорее следует рассматривать как неформальные справочники, чем онтологии в среде OWL. Это не означает, что модели OIM являются модулями справочных данных, на которые можно ссылаться в точке доступа SPARQL. Для пояснения представлений шаблонов OWL/RDF, нормативные представления моделей OIM не требуются.

Примечание 5 — Каждая область имен на рисунке 1 может импортировать мета-области имен, так как любой объект может обладать метаданными.

4 Модель данных ИСО 15926-2

На основе правил, установленных в приложении C, модель данных ИСО 15926-2 может быть отображена на классы языка OWL. Это определяется онтологией OWL (для) модели данных.

В приложении D устанавливается связь с нормативным листингом RDF/XML (для) модели данных.

5 Справочные данные ИСО/ТС 15926-4

Справочные данные представляются в соответствии с приложением E.

Примечание — ИСО/ТС 15926-4 содержит справочные данные, относящиеся к настоящему стандарту.

6 Спецификация шаблона языка OWL

6.1 Общие положения

Спецификация шаблона (см. раздел 7) содержит всю информацию, необходимую:

— человеку для понимания содержания экземпляров шаблонов;

— для создания языка OWL и листинга правил для шаблонов, в качестве дополнения к онтологиям шаблонов (см. ИСО/ТС 15926-7).

Если шаблон определен, то формат спецификации шаблона должен соответствовать 6.2.

Описание связей исходного множества спецификаций шаблона приведено в приложении F.

6.2 Содержание

Спецификация шаблона должна содержать:

— имя шаблона: уникальный идентификатор шаблона типа **English CamelCase** (горбатый шрифт на английском языке), содержащий большие и маленькие английские буквы без пробелов (см. раздел 7.1, верхний и нижний шаблоны);

— имя: наименование шаблона;

— назначение: краткое описание случаев применения шаблона;

— описание: полное описание семантики шаблона;

— верхний граф (lifted graph) и нижний граф (lowered graph): граф, показывающий, как верхний шаблон собирает узлы предметного графа (scope graph), а нижний шаблон ссылается на различные узлы (типы узлов) верхнего шаблона;

— верхний элемент шаблона: упорядоченный в алфавитном порядке список идентификаторов URI всех узлов, собранных верхним шаблоном, с соответствующими номерами «элементов»;

— код OWL для нижнего шаблона: связь с онтологией OWL шаблона;

— спецификацию логики первого порядка, используемую для формализации верхнего и нижнего шаблонов;

— экземпляр – образец нижнего шаблона (sample lowered template instance) в формате RDF/XML;

— определение свойств нижнего шаблона, содержащего перечень свойств нижнего шаблона с описанием объектов *rdf:objects.*

Примечание — Для верхнего и нижнего графов может быть использовано представление, являющееся традиционным в области RDF [24] или в Манчестер-синтаксисе [21].

7 Шаблоны

7.1 Общие положения

Для целей настоящего стандарта, информация представляется как комплексное отношение между независимыми объектами. Это комплексное отношение может быть бинарным или N-арным.

Бинарное отношение образуется на предикатах тройки RDF.

Пример — Отношение «специализация» (specialization) образуется с помощью rdfs:subClassOf. Отношение «классификация» (classification) образуется с помощью rdf:type.

N-арное отношение образуется с помощью шаблона, а ссылки на независимые объекты рассматриваются как внешние ссылки.

Шаблоны имеют вид пары:

— верхний, поднятый шаблон (lifted template) (см. 7.4);
— нижний, опущенный шаблон (lowered template) (см. 7.5).

Здесь нижний шаблон формирует структуру данных, которыми обмениваются с помощью данного шаблона, а верхний шаблон задает дефинициональное отображение (definitional mapping) нижнего шаблона на модель данных и справочные данные ИСО 15926.

Примечание — Если шаблон состоит только из слов, то это всегда нижний шаблон.

7.2 Модель шаблона

Экземпляр шаблона является элементом специализированного шаблона (отношение *rdf:type)*.

Специализированный шаблон — это подкласс специализированного или основного шаблона (отношение *rdfs:subClassOf*).

Основной шаблон — это подкласс класса шаблонов (отношение *rdfs:subClassOf*).

Класс шаблонов — это подкласс класса «Многомерные объекты» (отношение *rdfs:subClassOf*).

Класс шаблонов — это подкласс класса «Представления информации» (отношение *rdfs:subClassOf*).

Класс «Многомерные объекты» имеет тип данных сущности **class_of_multidimensional_objects** с нижеследующими атрибутами:

— количество элементов **cardinalities:** OPTIONAL LIST [1:?] OF **cardinality;**
— элемент по выбору **optional_element**: LIST [1:?] OF BOOLEAN (список булевских величин);
— параметры **parameters**: OPTIONAL LIST [1:?] OF **things** (список объектов по выбору)**;**
— позиция параметра **parameter_position**: OPTIONAL LIST [1:?] OF INTEGER (список целых);
— роли **roles**: LIST [1:?] OF **roles_and_domains** (список ролей и областей (доменов)).

Примечание — Класс «Шаблоны» более ограничен, чем класс «Многомерный объект», так как ИСО 15926-2 определяет списки параметров по выбору и позиции параметров **parameter_position**, не используемые для шаблонов. Для класса «Представления информации», не существует аргументов по выбору в шаблоне, а списки параметров и позиций параметров **parameter_position** не используются.

Примеры того, как все это вместе работает приведены в приложении Н.

7.3 Метаданные

Классы, индивидуальные объекты и шаблоны могут иметь метаданные.

Примечание — Некоторые элементы метаданных нормализованы в ИСО/ТС 15926-6.

В настоящем стандарте добавлены нижеследующие элементы метаданных экземпляров шаблонов:

— **annRule** (правило);
— **annAccessCode** (код доступа).

См. таблицу F.2 определений метаданных. Полное множество элементов метаданных и их трансляция на сетевой язык OWL приведены в приложении F.

7.4 Верхние (поднятые) шаблоны

Верхние (поднятые) шаблоны (lifted templates) служат для указания точной семантики нижних шаблонов.

Инстанцированные нижние шаблоны поднимаются наверх с помощью аксиом логики первого порядка, перечисленных в спецификации шаблона.

Примечание 1 — Верхние шаблоны могут быть инстанцированы по следующим причинам (и не только):

— если формат обмена выполнен полностью в верхнем формате;
— для валидации типов данных сущности данных, связанных с шаблоном;
— если обеспечиваемая полная семантика необходима так же, как необходим субъект (подлежащее) для формирования семантической сети;
— если требуется отображение в формат, определенный ИСО 10303-21 (так называемые файлы части 21);
— если данные отображаются с одного шаблона на другой, причем при этом отсутствует возможность обработки нижних шаблонов.

Верхние данные могут существовать без использования информации и метаданных верхнего шаблона. Понижение таких данных потребует сопоставления паттернов (образцов; от английского слова pattern). Вместе с тем, если поднимается нижний шаблон, то информация и метаданные также поднимаются. При этом понижение рассматриваемого верхнего шаблон дает тот же нижний шаблон без потери данных.

Примечание 2 — Верхние данные нужны, чтобы сделать компьютерную информацию интерпретируемой. В соответствии с ИСО 15926 данные хранятся в явном виде. Если эти данные поднимаются, компьютер может устанавливать отношения и делать заключения о данных.

Примечание 3 — Если принято решение, что данные хранятся и обмениваются только с помощью «верхних данных» или только с помощью нижних экземпляров шаблонов (объектов), то это бизнес-решения, и в настоящем стандарте они не рассматриваются. В принципе поддерживаются обе методологии.

Примечание 4 — Конкретизация верхних шаблонов дает крайне многословные представления. Но так как поднимать данные достаточно просто (с помощью экземпляров нижних шаблонов), то не следует постоянно пользоваться только верхним представлением данных, если это не обусловлено реальной ситуацией.

Верхний шаблон использует множество классов, определенных в онтологии OWL для моделей данных или подклассов конкретных шаблонов соответственно. Указанные подклассы (для данных шаблонов) имеют смысл только в контексте рассматриваемого шаблона.

Примечание 5 — Примеры верхних шаблонов см. в ИСО/ТС 15926-7.

7.5 Нижние (опущенные) шаблоны

Нижние, опущенные шаблоны (lowered templates) инстанцируются и используются для представления информации (в среде OWL) об индивидуальных объектах (экземплярах классов языка OWL).

Примечание — Исходное множество основных шаблонов приведено в ИСО 15927-7.

7.5.1 Инстанцирование нижних шаблонов

Определение классов шаблонов формируется с помощью структурных компонентов языка OWL. Свойства определяются в зависимости от обстоятельств наследования и имеющихся ограничений. Каждый экземпляр шаблона должен удовлетворять указанным ограничениям.

Примечание — Язык OWL предоставляет необходимые средства для определения схемы получения экземпляра и, таким образом, обеспечивает тщательность выполнения работы.

Любой экземпляр нижнего шаблона связан с соответствующим классом шаблонов с помощью свойства *rdf:type*.

В приложении H приведены примеры нижних шаблонов, а также показаны отношения между классами *owl:Classes* и их экземплярами.

Атрибут *rdf:resource* свойства экземпляра шаблона должен быть элементом соответствующего класса *owl:Class* в определении класса (подкласса) шаблона.

Пример — MPO-4983302 — это элемент класса owl:Class MaterializedPhysicalObject (Материализованный физический объект), а также класса owl:Class PhysicalObject (Физический объект) в соответствии с моделью данных ИСО 15926-2.

Информация об экземплярах может быть размещена на различных сайтах в Интернете, но она не должна дублироваться.

7.6 Шаблоны среды описания ресурса RDF

Методология представления шаблонов (их сигнатур) и экземпляров шаблонов (их сигнатур) в среде RDF/OWL рассмотрена в приложении B.

7.7 Пример: протошаблон

Пример протошаблона приведен в разделе H.2.

7.8 Шаблоны как экземпляры шаблонов

7.8.1 Классы p7tm для меташаблонов

Для представления шаблонов (сигнатур шаблонов в качестве данных) используются меташаблоны TemplateDescription (описание шаблона) и TemplateRoleDescription (описание роли шаблона). Ниже см. определения класса, соответствующие таблице сигнатур раздела B.2:

Class: p7tm:TemplateDescription
SubClassOf:
p7tm:MetaTemplateStatement,
p7tm:hasTemplate some p7tm:Template,
p7tm:valNumberOfRoles some integer

Class: p7tm:TemplateRoleDescription
SubClassOf:
p7tm:MetaTemplateStatement,
p7tm:hasRole some p7tm:TemplateRole,

p7tm:hasTemplate some p7tm:Template,
p7tm:hasRoleFillerType some Class,
p7tm:valRoleIndex some integer

Пример — Пример утверждения, содержащего шаблон, см. в Н.2.2.

8 Информационные модели объектов и специализированные шаблоны

8.1 Общие положения

Информационная модель объекта, рассмотренная в настоящем стандарте, является справочной. Для этих моделей не требуется однозначного соответствия требованиям ИСО/ТС 15926-8. Модели OIM могут служить руководством при специализированном моделировании шаблонов.

В контексте настоящего стандарта, методология разработки OIM рассматривается как часть ИСО 15926. Фактическое определение модели OIM индивидуального объекта, добавляющей онтологию к таксономии (при условии централизации и унификации) является частью основной библиотеки справочных данных RDL (Core RDL). Если централизации нет, то онтологии, определенные пользователем, могут существовать везде, они могут стать элементами централизации.

Модель OIM является онтологией классов и отношений (в терминологии логики описания, см. объекты **TBox** в приложении G), для характеризации которой выделен особый класс. Модель OIM устанавливает какая информация должна и может быть использована для описания членов указанного класса. Обычно такая модель строит фрагмент более широкой онтологии (библиотеку справочных данных) и добавляет ограничения для поддержки особых приложений.

Специализированные шаблоны являются множеством утверждений языка OWL об индивидуальных объектах или классах (см. объект **Abox** в приложении G), построенных в соответствии с моделью OIM, выступающей в роли руководства. Специализированные шаблоны нужны для инстанцирования моделей OIM, но сами они являются схемами языка OWL.

Техническое сопряжение специализированных шаблонов и моделей OIM отсутствует.

Специализированный шаблон характеризует индивидуальный объект обозначенного класса моделей OIM, используя только словарь классов и отношений, использованный в модели OIM. Специализированные шаблоны используются моделями OIM, так как они задают явные экземпляры для каждого существующего квалификатора и не нарушают какие-либо универсальные ограничения.

Примечание 1 — Специализированные шаблоны могут доставлять информацию, которая не является обязательной для модели OIM. Если кардинальное число модели OIM равно 0..*, то это как раз рассматриваемый случай.

Возможность коммерческого использования моделей OIM и специализированных шаблонов поясняется в примере ниже:

Пример — Пример отказа от использования моделей OIM: компания А обменивается данными с компанией В. Система технических баз данных содержит экземпляр «центробежный насос» и свойство «диаметр рабочего колеса». Интерфейс ИСО 15926 отображает эти данные как нижеследующие утверждения, представляющие собой экземпляр шаблона:

Объект P12345 является центробежным насосом

Объект P12345 имеет рабочее колесо диаметром 51 мм

Компания В также планирует импортировать эти данные в свою систему баз данных. Такое отображение предполагает наличие нижеследующих экземпляров:

Object P12345 is a Centrifugal Pump

Object P12345 has a assembly P56789

Object P56789 is a Impeller

Object P56789 has a Impeller Diameter of 51 mm

Обе указанные модели удовлетворяют требованиям ИСО 15926. Вместе с тем, связать их непросто. Чтобы импортировать данные, компания В должна установить правило отображения. Может оказаться, что правило отображения устанавливается, когда компания А уже отправила свои данные, и пришло сообщение об ошибке. Тогда нужно будет постараться заставить интерфейс работать. Вероятность такого риска высока.

Данный пример показывает, что если модели OIM существуют, то компании могут использовать эти концептуальные модели для определения архитектуры отображения интерфейса по ИСО 15926. Системы будут взаимодействовать сами без привлечения человека для установления правил

отображения.

Во-вторых, компаниям придется потрудиться, чтобы разработать отображение интерфейса. Использование информационных моделей объекта даст экономию затрат: вместо изобретения новых моделей, можно воспользоваться опытом создания уже существующих.

Примечание 2 — Если специализированные шаблоны лежат в центральном хранилище (например, основной библиотеке справочных данных RDL или в центральном хранилище проекта), то использование моделей OIM также может оказаться полезным. Использование моделей OIM не обязательно, так как риски нивелируются. Все отображения используют специализированные структуры шаблонов.

Примечание 3 — Каждая модель OIM является результатом консультаций и консенсуса между экспертами в данной области. Так как данные консультации идут постоянно онтологии языка OWL для информационных моделей объекта со временем расширяются.

Примечание 4 — Отображение данных в любой пользовательской системе на формат, определенный в настоящем стандарте, может иметь детерминистический характер, если данные уже определены в рассматриваемой модели OIM.

Примечание 5 — В принципе, в модель OIM уже все включено. Но в реальной жизни ее содержание может изменяться.

В приложении H приведен пример листинга RDF/OWL для онтологии OIM.

8.2 Информационные модели объектов и специализированные структурные компоненты шаблонов

Модели OIM не ограничиваются требованиями особого языка, тогда как специализированный шаблон должен быть выражен в среде RDF/OWL с помощью основных шаблонов в соответствии с настоящим стандартом.

Модели OIM служат примером того, как надо отображать и моделировать данные специализированных шаблонов.

Примечание — Модели OIM можно представить как на универсальном языке моделирования UML, так и на оригинальном языке OWL. В приложении H приведен пример представления на оригинальном языке OWL.

8.3 Специализация

Любой класс *owl:Class* наследует все свойства объектов *owl:ObjectProperties* от своего суперкласса. В контексте модели OIM это означает, что наследуется модель OIM суперкласса. На унаследованные отношения, при необходимости, могут быть наложены дополнительные ограничения.

Отношения, применимые только к подклассам, добавляются к отношениям, унаследованным от суперклассов.

Пример — Класс owl:Class RDS327239 «Насос» участвует в отношении, по которому у данного насоса должно быть от 2 до 6 подшипников. Подкласс RDS416834 «Центробежный насос» наследует данное отношение, но оно ограничено числом элементов (кардинальностью) от 2 до 3 подшипников.

8.4 Количество элементов множества

Имеется два вида кардинальности (мощности множества):

— количество допустимых экземпляров шаблона, связанных с экземплярами самого класса *owl:Class* или его потомков (дочерних элементов);

— количество объектов, инстанцированных в качестве целевых (для использования в той или иной роли) в инстанцированном шаблоне.

Оба указанных типа кардинальности могут быть определены утверждениями кардинальности модели OIM с помощью соответствующего языка моделирования.

Утверждения кардинальности используются, только если количество элементов не равно:

— 0.. 1 для допустимых экземпляров шаблона;

— 1.. 1 для допустимых экземпляров ролей (целевых).

8.5 Информационные модели OIM для сборок

Информация об объекте, являющемся частью другого объекта (сборочный объект), не принадлежит модели OIM, частью которой он является.

Примечание — Объект в целом не участвует прямо в отношениях, представляющих информацию о его части.

Пример — Диаметр рабочего колеса центробежного насоса не принадлежит модели OIM данного насоса, он относится к рабочему колесу. Шаблон с информацией, что рабочее колесо является (может быть) частью центробежного насоса, может включаться как в модель OIM рабочего колеса, так и в модель OIM насоса.

8.6 Соглашение о присвоении имен ролей шаблона

Для специализированных шаблонов названия ролей должны соответствовать соглашению о присвоении имен (конвенции именования). Имя роли имеет приставку:

— "**has**" для свойств объекта;
— "**val**" для свойств данных;
— "**ann**" для свойств аннотаций.

Пример — В соответствии с ИСО15926-7 роль «Значения свойств» декларируется как:

```
<owl:DatatypeProperty rdf:about="#valPropertyValue">
<rdfs:label>Property value</rdfs:label>
<rdfs:comment></rdfs:comment>
<rdfs:subPropertyOf rdf:resource="#valInitialSetTemplateDataRoleFiller"/>
</owl:DatatypeProperty>
```

В некоторых случаях одно и то же имя роли (транслированное с языка EXPRESS в соответствии с ИСО 15926-2 на язык OWL) может использоваться в смысле свойства объекта и в смысле свойства данных. Это относится к протошаблонам, где имена ролей должны соответствовать ИСО 15926-2. Чтобы имена классов *owl:Class* и имена свойств отличались, имена *свойств* имеют приставки "**has**", "**val**" и "**ann**". Это обеспечивает разделение в среде OWL между свойствами объекта и свойствами данных.

8.7 Контекст уникальности ролей шаблонов

В настоящем стандарте, как в тексте, так и в присоединенных файлах, декларации ролей являются уникальными в контексте области имен. Диапазон ролей шаблона определяется декларацией шаблона.

Вместе с тем, правильным является наличие субсвойств у ролей, имеющих одинаковую метку *rdfs:label*. Тогда они являются уникальными в контексте декларации шаблона.

Пример — Роль «Значения свойств» декларируется как:

```
<owl:DatatypeProperty rdf:about="#role_925636494">
<rdfs:label>Property value</rdfs:label>
<rdfs:comment></rdfs:comment>
<rdfs:subPropertyOf rdf:resource="#valPropertyValue"/>
</owl:DatatypeProperty>
```

Приложение А
(обязательное)

Регистрация информационного объекта

А.1 Идентификатор документа

Для однозначной идентификации информационного объекта в открытой системе настоящему стандарту присвоен следующий идентификатор:

{iso standard 15926 part(8) version(l)}

Смысл данного идентификатора определен в ИСО/МЭК 8824-1, описание приведено в ИСО 10303-1.

А.2 Идентификация схемы

А.2.1 Онтология языка OWL для модели данных

Для однозначной идентификации онтологии OWL для модели данных в открытой информационной системе, онтологии OWL для модели данных (см. раздел 4) назначается следующий идентификатор объекта:

{iso standart 15926 part(8) version(l) object(2) owl-ontology-for-data-model(1)}

Смысл идентификатора определен в ИСО/МЭК 8824-1, описание приведено в ИСО 10303-1.

А.2.2 Спецификации шаблонов

Для однозначной идентификации спецификации шаблона в открытой информационной системе, спецификации шаблона (см. раздел 6) назначается следующий идентификатор объекта:

{iso standart 15926 part(8) version(l) object(2) template-specifications(2)}

Смысл данного идентификатора определен в ИСО/МЭК 8824-1, описание приведено в ИСО 10303-1.

А.2.3 Онтология OWL для шаблонов

Для однозначной идентификации онтологии OWL для шаблонов в открытой информационной системе, онтологии OWL для шаблона (см. раздел 7) назначается следующий идентификатор объекта:

{iso standart 15926 part(8) version(l) object(2) owl-ontology-for-templates(3)}

Смысл данного идентификатора определен в ИСО/МЭК 8824-1, описание приведено в ИСО 10303-1.

А.2.4 Онтология OWL для справочных данных

Для однозначной идентификации онтологии OWL для информационной модели объекта в открытой информационной системе, онтологии OWL для справочных данных (см. раздел 5) назначается следующий идентификатор объекта:

{iso standart 15926 part(8) version(l) object(2) owl-ontology-for-reference-data(4)}

Смысл данного идентификатора определен в ИСО/МЭК 8824-1, описание приведено в ИСО 10303-1.

А.2.5 Онтология OWL для метаданных

Для однозначной идентификации онтологии OWL для метаданных в открытой информационной системе, онтологии OWL для метаданных (см. раздел 7.3) назначается следующий идентификатор объекта:

{iso standart 15926 part(8) version(l) object(2) owl-ontology-for-metadata(5)}

Смысл данного идентификатора определен в ИСО/МЭК 8824-1, описание приведено в ИСО 10303-1.

Приложение В
(обязательное)

Шаблоны как N-арные отношения в среде RDF/OWL

В.1 Общие положения

Для представления шаблонов (сигнатур) и экземпляров шаблонов (сигнатур) в среде RDF/OWL, принимается тип представления, описанный в *Паттерне 1, Пример использования 3 (Pattern 1, Use Case 3)* см. [22].

В данном примере рассматривается процесс покупки *Purchase,* в которой участвуют понятия: покупатель, продавец, объект, цель и количество. Все происходит в соответствии с иллюстрацией[1)] ниже:

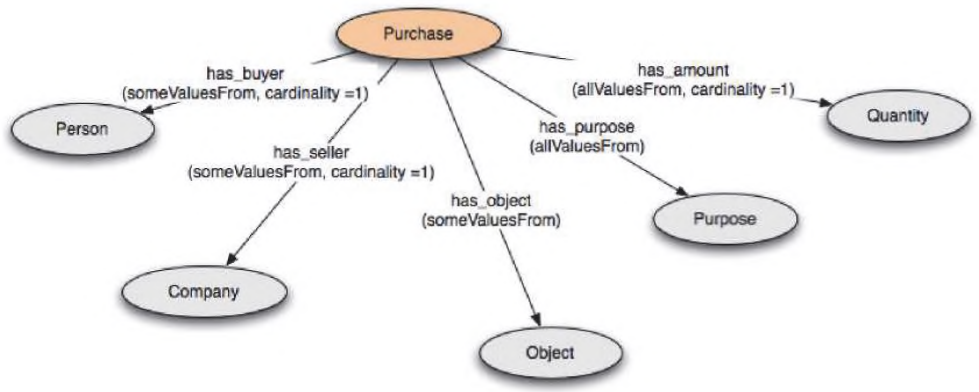

Рисунок В.I — Примечание рабочей группы W3C: определение N-арных отношений

Purchase	Покупка
has_buyer (someValuesFrom, cardinality =1)	Покупатель (приставка **has_** показывает, что рассматриваются свойства объекта) (некоторые значения, кардинальность = 1)
has_amount (allValuesFrom, cardinality=1)	Количество (все значения, кардинальность = 1)
person	Индивидуум
has_seller (someValuesFrom, cardinality=1)	Продавец (некоторые значения, кардинальность = 1)
has_purpose (allValuesFrom)	Цель (все значения)
Quantity	Количество
has_object (someValuesFrom)	Объект (некоторые значения)
purpose	Цель
company	Компания
object	Объект

Имеются отличия настоящего представления отношения *Purchase* от шаблонов ИСО 15926. Например:

а) вся конструкция системы *Purchase* аналогична сигнатуре шаблона в соответствии с ИСО 15926.

Примечание 1 — В примечании рабочей группы W3C (также как и для сигнатуры по ИСО 15926) не уделяется внимания моделированию отношения к участникам рассматриваемого материализованного отношения. По ИСО/ТС 15926-7, аксиомы явного представления *Purchase* отсутствуют.

[1)] Оригинальная информация содержится в [22].

b) некоторые роли совсем не обязательны. Это недопустимо для шаблонов ИСО 15926.

Примечание 2 — Основная причина недопустимости необязательных ролей: резкое ужесточение требований к переработке данных;

c) не существует выражения, точно устанавливающего роли, используемые в N-арном отношении *Purchase*;

d) класс *Purchase* (как любой экземпляр шаблона ИСО 15926) формирует утверждение;

Примечание 3 — См. примечание *Примера использования* 3: «Мы создаем индивидуальный объект, чтобы представить экземпляр отношения, имеющий связи со всеми участниками». Это не покупка, а отношение, имеющее представление.

Пункт c) связан с процедурой *нумерации* ролей шаблонов ИСО 15926. В данном примечании не делается попытки определить явно и точно какие «атрибуты» используются в определении *Purchase.* Таким образом, определение является открытым. В отличие от шаблонов ИСО 15926, список ролей задается как перенумерованный и конечный перечень.

Пункт d) является принципиальным для структуры N-арных отношений. В них элементы класса *Purchase* (а также экземпляры шаблонов ИСО 15926) не предназначены для представления покупок. Это только утверждения, которые описывают данные покупки. Например, *было бы неправильно использовать идентификатор для экземпляра* Purchase, *чтобы ссылаться на некоторую покупку.* Если идентификатор покупки все-таки нужен, то он должен быть добавлен к N-арному отношению в качестве роли. Данное указание содержится в примечании W3C [22]. На него можно сделать дополнительный акцент.

В большинстве случаев, указанные индивидуальные объекты не используются самостоятельно. Это только вспомогательные функции, нужные для группировки других объектов. Поэтому давать им особое имя нет смысла.

Их место становится понятным, если учесть факт, что в отношении конкретной покупке может быть сформулировано много различных утверждений (экземпляров шаблона). Класс *Purchase,* описанный в примечании W3C, не очень для этого подходит, так как он не учитывает роль идентификатора покупки. Это хороший пример моделирования шаблонов: все, что можно указать в утверждении шаблона, требует назначения роли (и только в исключительных случаях появляется необходимость рассматривать утверждения сами по себе.)

Используя настоящее представление, будьте внимательны: язык OWL не имеет достаточных ресурсов для определения идентичности экземпляров. Ясно, что в реляционных базах данных не бывает двух отдельных строк с одинаковыми элементами. Об этом, в частности, в примечании ничего не говорится. Та же ситуация складывается с представлением в среде OWL по ИСО 15926 (также можно получить аппроксимацию с помощью нумерующего представления, см. ниже).

Для полного представления критерия идентичности необходимо использовать логику первого порядка.

В контексте ИСО 15926 принято обоснованное допущение, что идентичность скорее следует использовать на уровне реализации/приложения, нежели вместе со структурными компонентами при семантическом моделировании. Это согласуется с наилучшей практикой [использования] семантической сети.

B.2 Структурные компоненты настоящего стандарта для сигнатур шаблонов

В разделе определены малые онтологии для представления шаблонов. Для файлов - примеров настоящего документа, онтология модели шаблона ИСО/ТС 15926-7 имеет область имен:

gost.e-ecolog.ru

с сокращенным префиксом **p7tm**.

Классы и отношения **p7tpl**, включающие в себя представления сигнатур шаблонов, имеют вид:

p7tm:TemplateStatement класс *owl:Class* — это результирующий суперкласс всех шаблонов. Суффикс "**Statement**" (утверждение) очень важен. Он указывает, что экземпляры шаблонов представляют утверждения. Правильный **dm**-суперкласс класса **p7tm:TemplateStatement** — это **dm:MultidimensionalObject** (многомерный объект).

Подклассы **p7tm:TemplateStatement**:

p7tm:BaseTemplateStatement — используется как суперкласс всех базовых шаблонов, которые по очереди используются как суперклассы основных шаблонов;

p7tm:MetaTemplateStatement — структура модели для всех шаблонов с нижеследующими подклассами:

p7tm:TemplateDescription (описание шаблона);

p7tm:TemplateRoleDescription (описание роли шаблона);

p7tm:TemplateSpecialization (специализация шаблона);

p7tm:has0bjectRoleFiller — данное свойство объекта *owl:objectProperty* — результирующее суперотношение всех ролей, которое использует индивидуальные объекты как наполнители ролей;

Примечание 1 — Не существует естественного **dm**-суперотношения для указанного отношения, так как в ИСО 15926-2 (см. **dm:MultidimensionalObject**) соответствующие сущности представлены как списки. Рассматриваемое отношение — это «техническое» устройство, поддерживающее представление N-арных сказуемых. Данное отношение не имеет естественной интерпретации в терминах ИСО 15926-2.

p7tm:hasDataRoleFiller Настоящее свойство типа данных *owl:DatatypeProperty* – это результирующее суперотношение для всех ролей, принимающих значения типа данных.

Примечание 2 — Также как и для **p7tm:hasObjectRoleFiller**, для указанного отношения не существует естественного **dm**-суперотношения.

Сигнатура шаблонов является подклассом утверждения **p7tm: TemplateStatement**.

Представление информации о шаблонах очень важно. В частности, важно указать количество ролей шаблона и позицию каждой роли. Для этого шаблон представляется как индивидуальный объект с использованием механизма *трамбования* в среде OWL 2. Сущность **p7tm** определяет два шаблона для описания самих шаблонов и их ролей;

p7tm:TemplateDescription С ролями *template* и *numberOfRoles* указанный шаблон можно использовать для описания других шаблонов;

Примечание 3 — Имя, определение и прочие характеристики шаблонов можно найти где-либо, например, с помощью обобщенных (типовых) идентификационных шаблонов.

№	Название роли	Тип роли
1	**p7tm:hasTemplate**	**p7tm:Template**
2	**p7tm:valNumberOfRole**	**xsd:integer**

p7tm:TemplateRoleDescription С ролями *template, positionInTemplate, role* и *type,* настоящий шаблон можно использовать для описания ролей других шаблонов.

№	Название роли	Тип роли
1	**p7tm:hasRole**	**p7tm:TemplateRole**
2	**p7tm:hasTemplate**	**p7tm:Template**
3	**p7tm:valvalRoleIndex**	**Xsd:Integer**
4	**p7tm:hasRoleFillerType**	**dm:Class**

Указанные «металшаблоны» обеспечивают представление самих шаблонов как данных.

Примечание 4 — Важно, что они представляют потенциальную проблему актуализации: если определение класса шаблонов изменяется, то все экземпляры металшаблона, описывающие рассматриваемый шаблон, нужно соответственно обновлять.

Представление *специализации шаблона* **p7tm:TemplateSpecialization** используется для указания на то, что один шаблон является специализацией другого.

Примечание 5 — Это подразумевает разделение языка OWL на подклассы. Но это не означает, что специализацию шаблонов следует понимать в соответствии с ИСО/ТС 15926-8.

№	Название роли	Тип роли
1	**p7tm:hasSupertemplate**	**p7tm:Template**
2	**p7tm:hasSubtemplate**	**p7tm:Template**

Приложение С
(обязательное)

Правила использования языка OWL

С.1 «Исходное» представление языка OWL

Предполагается, что представление контента ИСО 15926 должно без ограничений соответствовать требованиям среды OWL. Типы данных сущности представляются как классы и свойства языка OWL.

Примечание 1 — Вышесказанное находится в контрасте с (*воплощенным)* форматом ИСО 15926-2, в котором классы и свойства определены в качестве индивидуальных объектов. Данный (воплощенный) формат представляет модель данных ИСО 15926-2 с учетом требований языка EXPRESS. Однако это не подходит для случаев использования форматов шаблонов, установленных в настоящем стандарте.

Пример трансляции модели данных (верхней онтологии) ИСО 15926-2 на язык OWL приведен в приложении D.

Следующие рисунки поясняют некоторые аспекты представления ИСО 15926 на «исходном» языке OWL. При этом отношения не воплощаются, а заменяются простыми бинарными отношениями (объект OWL и свойства данных) вместо структурного компонента, показанного на рисунке С.1,

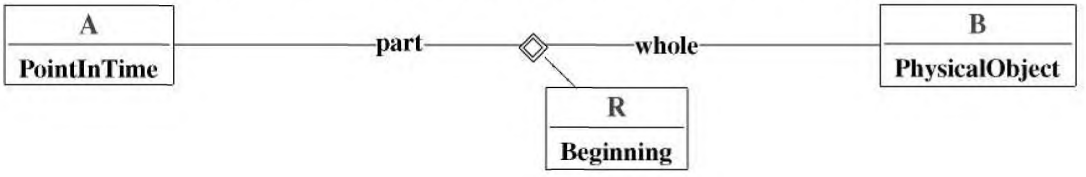

Рисунок С.1 — Воплощенное отношение

PointInTime	Момент времени
-part-	Часть
-whole-	Целое
beginning	Начало
PhysicalObject	Физический объект

где каждое отношение **Beginning** представлено индивидуальным объектом как элементом **Beginning**, используется паттерн (образец), приведенный на рисунке С.2.



Рисунок С.2 — Отношение на «исходном» языке OWL

PointInTime	Момент времени
-hasBeginning-	Начало (с приставкой **has**)
PhysicalObject	Физический объект

Примечание 2 — Представление на «исходном» языке OWL предъявляет более высокие семантические требования к разработчику модели, чем подход с «воплощениями». Если используются воплощения, то сущность может быть представлена как в качестве ролей отношений, так и классов индивидуального объекта. В случае «исходного» представления выбирается только один из них. Некоторые конструктивы ИСО 15926-2 полагаются на гибкость воплощения, так как трансляция на «исходную» форму языка OWL всегда является неполной. Однако это не является большим недостатком, так как конструктивы «исходного» языка OWL имеют очень широкое распространение: можно рассчитывать на широкий диапазон инструментов, использующих данный язык как основу.

Главные особенности представлений на «исходном» языке OWL:

типы данных сущности как классы: типы данных сущности по ИСО 15926-2, естественно представленные унарными предикатами, заданы представлениям на «исходном» языке OWL как классы (*owl:Class*);

типы данных сущности как отношения: типы данных сущности **ClassOfClassOfRelationship, ClassOfRelationship,** и **Relationship** по ИСО 15926-2 естественно представляются бинарными предикатами (бинарными отношениями). Они представлены как отношения в среде OWL. Имеется два типа этих отношений:

индивидуальный объект — отношения индивидуальных объектов: типы данных сущности, относящиеся к парам индивидуальных объектов, обозначаются как *owl:ObjectProperty*;

индивидуальный объект — отношения значений данных: типы данных сущности (**ClassOfRepresentationsOfThings** и их подтипы), сопоставляющие индивидуальные объекты со значениями данных, обозначаются как *owl:DatatypeProperty*.

С.2 Трансляция шаблона модели экземпляра с «исходного» языка EXPRESS на «исходный» язык OWL

В случае практической реализации нижних шаблонов с помощью аксиом ИСО/ТС 15926-7, роли шаблонов используются как свойства предикатов. Внешние данные, такие как диапазон значений указанного объекта, свойства данных, а также результат их обработки представляются на «исходном» языке OWL.

Для верхних шаблонов, их расширение, систематизированное по ИСО/ТС 15926-7, позволяет получить представление модели экземпляра (instance model) на «исходном» языке OWL.

Примечание — Оба метода могут не удовлетворять требованиям настоящего стандарта в части трансляции модели экземпляра EXPRESS в другую модель, представленную на «исходном» языке OWL.

С.3 Трансляция объекта библиотеки с «исходного» языка EXPRESS в модель экземпляра на «исходном» языке OWL

Для объектов библиотеки (классов и справочных данных), размещаемых в основной библиотеке RDL и в других хранилищах данных, установлены следующие правила трансляции.

Отображение модели данных ИСО 15926-2 на язык OWL в настоящем приложении не рассматривается: такая трансляция уже выполнена, и ее результаты доступны в качестве точки доступа (endpoint) (см. приложение D).

При отображении модели данных ИСО 15926-2 на онтологию OWL, могут иметь место трудности, обусловленные несовместимостью двух языков.

В настоящем приложении определяется отображение структурных компонентов языка EXPRESS по ИСО 15926-2 на компоненты языка OWL (включая некоторые доработки для преодоления несовместимостей).

Представление информации не ограничивается использованием среды описания ресурса RDF и сетевого языка онтологий OWL. В настоящем приложении альтернативные представления запрещены. Рассмотрены особые варианты представлений.

Примечание — Возможны связи словарей языка OWL с различными действующими представлениями.

С.4 Класс *owl:Class*

Для каждого типа данных сущности модели данных ИСО 15926-2 можно создать экземпляр класса *owl:Class*.

Создание подтипов на языке EXPRESS сопровождается отображением на компоненты *rdfs:subClassOf*.

Пример 1 — В схеме языка EXPRESS для модели данных ИСО 15926-2 атрибут point_in_time является подтипом события event. Данный атрибут представляется в среде OWL как:

```
<owl:Class rdf:ID="PointInTime">
<rdfs:subClassOf rdf:resource="#Event"/>
</owl:Class>
```

Свойства *owl:ObjectProperties* и *owl:DatatypeProperties* наследуются от суперкласса, описанного в разделе С.8. В среде OWL поддерживается множественное наследование.

Примечание 1 — Свойства *owl:AnnotationProperties* от суперкласса не наследуются.

Экземпляр класса *owl:Class* может быть частью файла RDF/XML. Это справедливо и в отношении любого элемента экземпляра класса *owl:Class.*

Примечание 2 — В среде OWL данный элемент называется «индивидуальным объектом» (individual) (отличается от объекта **possible_individual**).

С.5 Индивидуальный объект

Индивидуальный объект (individual) в среде OWL — это экземпляр класса *owl:Class.*

Примечание — Иногда трудно отличить экземпляр класса от элемента класса *owl:Class.* В настоящем стандарте понятие «экземпляр» (instance) класса указывает на применимый тип данных сущности ИСО 15926-2, а «элемент» (member) класса указывает, какому множеству объектов (с присущими характеристиками) принадлежит рассматриваемый объект в соответствии с критерием принадлежности к заданному множеству объектов.

С.6 Свойство

Свойство в среде RDF — это имя предиката (predicate) в *тройке* RDF. Оно дает информацию о *субъекте (subject)* указанной *тройки*, путем ссылки на *объект (object)* этой тройки.

Примечание — Понятие свойства **property** (типа данных сущности ИСО 15926-2) отличается от понятия свойства *property* (отношение в среде OWL).

В среде OWL определяются три вида *свойств*:

— *owl:ObjectProperty (свойство объекта);*

— *owl:DatatypeProperty (свойство типа данных);*

— *owl:AnnotationProperty (свойство аннотации).*

С.6.1 Свойство объекта

Свойство *owl:ObjectProperty* характеризует (индивидуальный) субъект *(subject individual)* путем ссылки на индивидуальный объект *object individual.*

Примечание — Важно, что только экземпляры классов языка OWL могут быть *субъектом (subject)* или *объектом (object)* свойства *owl:ObjectProperty.*

С.6.2 Свойство типа данных

Свойство типа данных *owl:DatatypeProperty* характеризует (индивидуальный) субъект путем ссылки на значение типа данных XML схемы объекта.

Для строк, номеров, сочетаний «день — время» и других типов данных следует использовать экземпляры типа данных применяемой XML схемы (см. [18]) по аналогии с библиотекой справочных данных ИСО/ТС 15926-4.

Свойство *owl:DatatypeProperty* используется:

— в верхних данных ИСО 15926-2 для сущностей, позволяющих хранить символьные константы, например *ClassOfInformationRepresentation* с атрибутом содержания «content»;

— для ролей нижних шаблонов ссылающихся на символьные константы (буквенные выражения и цифры).

С.6.3 Свойство аннотации

Свойство аннотации *owl:AnnotationProperty* представляет метаданные *класса субъекта (subject class)* или *индивидуального объекта (individual)* путем использования *ссылки на объект URI.* Пример использования приведен в приложении F.

С.7 Идентификаторы

С.7.1 Идентификатор *rdf:ID*

Каждый экземпляр типа данных сущности, определенный в модели данных ИСО 15926-2, имеет уникальный атрибут **id**, унаследованный от сущности **thing**.

В документе RDF/XML, требующем соответствия настоящего стандарта, представление атрибута **id** с помощью атрибута *rdf:ID*, должно быть выполнено в формате RDF ссылок URI в соответствии с [15].

В настоящем стандарте, атрибуты *rdf:ID* предпочтительно представляются в формате идентификаторов фрагментов **URI#fragnment-identifier**.

Примечание 1 — В настоящем стандарте не указано, что все идентификаторы ID должны иметь тип идентификатора-фрагмента **URI#fragment-identifier**. Идентификаторы ID, взятые из других частей и других источников могут также быть универсальными идентификаторами ресурса URI (не содержащими символа #). Вместе с тем, если можно сделать выбор, то предпочтение следует отдавать **URI#Fragment-identifier**. См. [19].

Примечание 2 — Ограничение, накладываемое путем использования *rdf:ID,* заключается в том, что значения всех атрибутов и элементов, имеющих тип данных ID, должны быть уникальными. Если используемый идентификатор не является уникальным (так как он является ссылкой на другой идентификатор *rdf:ID* как внутри, так и вне рассматриваемого хранилища данных), то следует использовать атрибут *rdf:about*. В работе [14] сказано: «Атрибут *rdf:ID* узлового элемента (это не элемент свойства с другим смыслом) может быть использован вместо атрибута *rdf:about.* Он дает относительную ссылку на идентификатор URI в среде RDF, эквивалентный идентификатору #-типа, конкатенированному со значением атрибута *rdf:ID*. Так, например, если *rdf:ID*="name" (имя), то это эквивалентно заданию *rdf:about*="#name". Атрибут *rdf:ID* обеспечивает дополнительную проверку, так как то же самое имя может появиться только один раз в области *xml:base* значений (или документа, если других данных нет). Он полезен при определении множества самостоятельных связанных терминов по отношению к той же ссылке на идентификатор URI в среде RDF».

Примечание 3 — Типом данных *rdf:ID* является **NCName**. Ниже приведена выдержка из [18] (раздел 2):

NCName ::= (Letter | '_') (NCNameChar)*

NCNameChar ::= Letter | Digit | '.' | '-' | '_' | CombiningChar | Extender

Идентификатор *rdf:ID* должен начинаться с буквы. Дальше могут идти буквы, цифры, точки, тире и нижнее подчеркивание. Использование двоеточий недопустимо. Полное определение см. в описании расширяемого языка разметки «Extensible Markup Language (XML) 1.0 (издание 5)».

C.7.2 Атрибут *rdf:about*

Используемый в среде RDF Production, синтаксический термин *rdf:about* используется при повторном появлении объекта в той позиции, где он не определен, или внутри множества данных, где данные объекта не являются самодостаточными.

Примечание — Если атрибут *rdf:ID* используется для объявления класса или индивидуального объекта, то атрибут *rdf:about* используется для добавления некоторой информации об указанном классе или индивидуальном объекте.

C.7.3 Ссылочный идентификатор ресурса URI для передаваемых объектов

В случае, когда объект передается в другое хранилище данных, ссылочный идентификатор ресурса URI меняться не должен. Он должен сохранять свое значение и быть уникальным в среде URI.

Примечания

1 Если данные создаются компаниями при разработке проектов с полным циклом производства и сбыта или несколькими подрядчиками и субподрядчиками проекта EPC, то идентификатор URI следует брать из последнего хранилища данных RDF после передачи данных «по факту». Это должен быть идентификатор URI оборудования (установки), поставленной клиенту.

2 Данные, созданные в ходе технологического процесса, предшествующего рассматриваемой передаче данных, располагаются в местных хранилищах RDF. Это создает проблему, потому что должна существовать возможность переименовать связанные идентификаторы данных URI (для обеспечения надежности функционирования URI). См. примеры использования W3C для решения указанной проблемы [19].

C.8 Наследование

Класс *owl:Class,* связанный с другим классом *owl:Class* с помощью свойства *rdfs:subClassOfProperty,* наследует определение путем наследования отношений для специализированных шаблонов.

Исключением понятия наследование являются все экземпляры аннотационных свойств *owl:AnnotationProperty* (см. приложение F, используемое для получения мета-информации о классе *owl:Class).*

Элемент класса *owl:Class,* связанный с данным классом *owl:Class* посредством свойства *rdf:type,* должен удовлетворять ограничениям, накладываемым определением указанного класса *owl:Class.*

C.8.1 Декларация

В некоторых случаях OWL ресурсы не декларируются (не объявляются), но в настоящем стандарте это является обязательным требованием.

Единственное исключение, это когда временная часть (являющаяся объектом *rdf:object* свойства шаблона) не должна декларироваться явно.

Примечание — Понятие «временная часть» (temporal part) определяется в модели данных ИСО 15926-2 в контексте отношения **temporal_whole_part.** Временная часть наследует существенную классификацию от своего временного целого. Явное декларирование может привести к избыточности и, возможно, к потере данных.

Приложение D
(обязательное)

Компьютерно-представимый листинг

D.1 Онтология модели данных

Рассматриваемый ниже листинг — это результат отображения модели данных, определенной в ИСО 15926-2, в среду OWL.

В том случае, когда правила отображения (в соответствии с настоящим стандартом) противоречат результату отображения (см. ниже файл OWL), приоритет имеет определение файла OWL.

Полная модель данных представлена на компакт-диске CD-ROM настоящего стандарта как файл RDF «**data-model.owl**».

D.2 Онтология модели шаблона

Рассматриваемый ниже листинг включает декларации, используемые структурными компонентами шаблонов.

В том случае, когда правила отображения (в соответствии с настоящим стандартом) противоречат результату отображения (см. ниже файл OWL), приоритет имеет файл OWL.

Полная модель шаблона представлена на компакт-диске CD-ROM настоящего стандарта, как файл RDF «**p7tm.owl**».

D.3 Онтология протошаблонов и исходного множества шаблонов

Рассматриваемый ниже листинг включает структурные компоненты протошаблонов и исходного множества шаблонов, определенные ИСО/ТС 15926-7.

В том случае, когда правила отображения (в соответствии с настоящим стандартом) противоречат результату отображения (см. ниже файл OWL), приоритет имеет файл OWL.

Листинг представлен на компакт-диске CD-ROM настоящего стандарта, как файл RDF «**p7tpl.owl**».

D.4 Онтология метаданных

Рассматриваемый ниже листинг содержит несколько примеров метаданных, а также метаданных, определенных в настоящем стандарте.

В случае, когда правила отображения (в соответствии с настоящим стандартом) противоречат результату отображения (см. ниже файл OWL), приоритет имеет файл OWL.

Листинг представлен на компакт-диске CD-ROM настоящего стандарта, как файл RDF «**metadata.owl**».

Приложение Е
(справочное)

Паттерны (образцы) онтологии справочных данных

Классы библиотеки RDL определяются своими метаданными (см. приложение F) и экземплярами шаблонов. В настоящем приложении рассмотрен пример основного класса RDL, содержащего метаданные.

Пример — Рассматривается экземпляр класса «centifugal pump» (центробежный насос):

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
<!ENTITY xsd "gost.e-ecolog.ru ">
<!ENTITY owl "gost.e-ecolog.ru ">
<!ENTITY rdf "gost.e-ecolog.ru ">
<!ENTITY rdfs "gost.e-ecolog.ru ">
<!ENTITY dm "gost.e-ecolog.ru ">
<!ENTITY p7tm "gost.e-ecolog.ru ">
<!ENTITY p7tpl "gost.e-ecolog.ru ">
<!ENTITY meta "gost.e-ecolog.ru ">
<!ENTITY rdl "gost.e-ecolog.ru ">
<!ENTITY steplib "gost.e-ecolog.ru ">
<!ENTITY pca "gost.e-ecolog.ru ">
<!ENTITY company1 "gost.e-ecolog.ru ">
]>
<rdf:RDF xmlns="gost.e-ecolog.ru "
xml:base="gost.e-ecolog.ru "
xmlns:owl="&owl;"
xmlns:xsd="&xsd;"
xmlns:rdf="&rdf;"
xmlns:rdfs="$rdfs;"
xmlns:dm="&dm;"
xmlns:p7tm="&p7tm;"
xmlns:p7tpl="&p7tpl;"
xmlns:meta="&meta;"
xmlns:rdl="&rdl;"
xmlns:steplib="&steplib;"
xmlns:pca="&pca;"
>

<owl:Class rdf:ID="#RDL7436"> <!-- from unique number -->
<rdfs:subClassOf rdf:resource="#RDL7611"/> <!-- dynamic pump -->
<rdf:type rdf:resource="&dm;ClassOfInanimatePhysicalObject"/>
<rdf:type rdf:resource="#RDL1957"/> <!-- rotating equipment class -->
<meta:annUniqueName>centrifugal pump</meta:annUniqueName>
<meta:annTextDefinition rdf:datatype="&xsd;string">
A dynamic pump that contains impellers provided with vanes to generate
centrifugal force to achieve the required pressure head.
</meta:annTextDefinition>
<steplib:identifier rdf:resource="&steplib;SL130058"/>
<pca:identifier rdf:resource="&pca;RDS416834"/>
<meta:annCreationDate rdf:datatype="&xsd;dateTime">
1999-07-01T21:32:52Z
</meta:annCreationDate>
<meta:annRegistrationStatus rdf:datatype="&xsd;string">
IS
</meta:annRegistrationStatus>
</owl:Class>

</rdf:RDF>
```

Приложение F
(обязательное)

Метаданные

F.1 Общие положения

В настоящем стандарте метаданные используются для аннотирования происхождения инстанцированных и постоянных (устойчивых) объектов, а также групп безопасности и правил. Если данные, связанные с объектом, не требуют указанного аннотирования, то их можно использовать при моделировании не как аннотацию, а с помощью шаблонов и других структурных компонентов.

Примечание 1 — Некоторые метаданные, требующие аннотирования происхождения, опубликованы в ИСО/ТС 15926-6, но к ним не следует относиться как к нормативным данным.

Примечание 2 — Метаданные не являются обязательными.

Примечание 3 — Метаданные могут быть свойствами:

— классов;
— экземпляров;
— шаблонов.

Примечание 4 — Так как метаданные являются свойством самого объекта (а не его дочерних элементов), то здесь уместно говорить об отсутствии наследования.

Примечание 5 — В соответствии с определениями среды OWL, объект свойства аннотации — это буквенные данные, ссылка на идентификатор URI или на индивидуальный объект.

Примечание 6 — Свойства аннотаций обычно игнорируются механизмами (блоками) рассуждений.

F.2 Декларации метаданных о происхождении

Если метаданные:

— удовлетворяют требованиям их использования для аннотирования происхождения;
— ссылаются на буквенные данные (строки или цифры);
— не используются в сущностях *rdfs:label, rdfs:seeAlso* и *rdfs:comment,*

то нужно использовать точные имена (например, идентификаторы ID в декларации аннотационных свойств). При этом пробелы и подчеркивания следует убрать, слова с приставкой «**ann**» записываются в «горбатом» формате (**CamelCase**). Исключением является свойство, используемое для определения уникального номера объекта и непосредственно отображаемое на атрибуты *rdf:ID* и *rdf:about.*

Пример 1 — Имя метаданных:
дата создания
Декларация аннотационного свойства:
<owl:AnnotationProperty rdf:ID="annCreationDate" />

Если заявленные метаданные:

— удовлетворяют необходимым требованиям для установления происхождения;
— могут использоваться в сущностях *rdfs:label, rdfs:seeAlso* и *rdfs:comment,*

то нужно использовать точные имена (например, как идентификаторы ID в декларации аннотационных свойств). Пробелы и подчеркивания следует убрать, а слова с приставкой «**ann**» (если они относятся к буквенной константе) и с приставкой «**has**» (если они относятся к объекту) указываются в «горбатом» формате (**CamelCase**). Они декларируются с учетом отношения *rdf:subPropertyOf.*

Пример 2 — Пример свойств метаданных, используемых для сущностей rdfs:label, rdfs:seeAlso и rdfs:commen:

Таблица F.1 — Примеры свойств сущности *rdf:subPropertyOf*, являющихся стандартными аннотационными свойствами в среде OWL

Имя	Свойство метаданных	rdf:subPropertyOf
Уникальное имя	*meta: annUniqueName*	*rdfs:label*
Источник	*meta:hasSource*	*rdfs:seeAlso*
Текстовое определение	*meta: annTextDefinition*	*rdfs:comment*
Примечание	*meta:annNotes*	*rdfs:comment*
Административное примечание	*meta:annAdministrativeNote*	*rdfs:comment*
Пояснительный комментарий	*meta:annExplanatoryComment*	*rdfs:comment*
Описание изменения	*meta:annChangeDescription*	*rdfs:comment*

F.3 Декларации метаданных о безопасности и правилах применения

В таблице F.2 приведены декларации свойств метаданных, определенных в разделе 7.3.

Таблица F.2 — Элементы метаданных ИСО/ТС 15926-8

Имя	Декларация свойства в среде OWL	Кардинальное число	Описание
Правило	*<owl:AnnotationProperty rdf:ID="#annRule"> <rdfs:domain rdf:resource="#TemplateStatement"/> </owl:AnnotationProperty>*	0 — ко многим	Строка, определяющая правила, методы или сценарии, применяемые к любому программному обеспечению.
Код доступа	*<owl:AnnotationProperty rdf:ID="#annAccessCode"> <rdfs:domain rdf:ресурс="#TemplateStatement"/> </owl:AnnotationProperty>*	0 — ко многим	Строка, содержащая код безопасности (группы ресурса) информации, представляемой объектом. Используется как параметр в списке ACL (список контроля доступа).

Примечание — Область указывается в декларации аннотационных свойств как утверждение шаблона **TemplateStatement** или подкласса соответственно. Это означает, что эти два указанных аннотационных свойства могут быть использованы только для шаблонов. Диапазон значений декларации аннотационных свойств не указан. Это означает, что указанное свойство может указывать на символьную константу, на ссылочный идентификатор URI или на индивидуальный объект.

Приложение G
(справочное)

Утверждения и терминологические компоненты

Сущность **ABox** — это компонент утверждения (высказывания). Это факт, ассоциированный с терминологическим словарем внутри базы знаний.

Сущность **TBox** — это также терминологический компонент. Это словарь, ассоциированный с множеством фактов **ABox.**

Термины **ABox** и **TBox** используются для описания двух различных типов утверждений онтологий. Утверждения **TBox** описывают систему в терминах управляемых словарей (например, множества классов и свойств).

Утверждения **ABox** в отношении словаря удовлетворяют требованиям **TBox**.

Утверждения **TBox** по своей природе более конкретны.

Пример — Словарь слов — это TBox.

Примечание — Обычно утверждения **ABox** имеют следующую форму:
«А — это экземпляр В»
или
«изделие Р-101 классифицируется как насос»,
или
«максимальное расчетное давление изделия Р-101 равно 33 Бар».
Утверждения **Abox** отличаются от утверждений **TBox** своей терминологией:
«все пожарные насосы — это насосы»
или
«существует два типа пожарных насосов: поршневые и центробежные»,
или
«входная температура насосов водяного охлаждения имеет диапазон от 0 до 18 градусов Цельсия».

Утверждения **TBox** более устойчивы в рамках используемой базы знаний. Они хранятся в модели данных или реестре метаданных. Утверждения **ABox** по своей природе более динамичны. Они хранятся в базах данных как данные экземпляров транзакционных систем.

Приложение Н
(справочное)

Примеры

Н.1 Общие положения

Настоящее приложение содержит примеры.

Приведенные данные являются вымышленными, они не существуют в реальном времени. Идентификаторы ID выбраны из соображений удобства чтения. Для этого среда описания ресурса RDF приведена в Манчестер-синтаксисе [21].

Н.2 Пример: протошаблон

Протошаблон **p7tpl:Beginning** используется как простейший пример представления шаблона в среде RDF.

Н.2.1 Шаблон, определенный как класс

Класс **p7tpl:Beginning** определяется как высказывание (утверждение) протошаблона **ProtoTemplateStatement**:

```
Class: Beginning
    SubClassOf:
        ProtoTemplateStatement,
        hasPart some dm:Event,
        hasWhole some dm:PossibleIndividual
```

Роли **p7tpl:hasPart** и **p7tpl:hasWhole** сами являются подтипами **p7tm:hasObjectRoleFiller**.

```
ObjectProperty: hasObjectRoleFiller
    Range:
        dm:Thing

ObjectProperty: hasPart
    Characteristics:
        Functional
    Range:
        dm:PossibleIndividual
    SubPropertyOf:
        hasObjectRoleFiller
```

Н.2.2 Высказывание, использующее шаблон

В настоящем примере приведен экземпляр шаблона.

Пусть **Alfred** — это индивидуальный объект, и пусть **_1970-01-01** — это событие **Event,** являющееся для объекта **Alfred** начальным событием.

Примечание — Начальное событие в дальнейшем представляется как **xsd:dateTime**.

Объект **Alfred** и начальное событие определяются как:

```
Individual: company1:Alfred
    Types:
        WholeLifeIndividual

Individual: company1:_1970-01-01
    Types:
        Event
    Facts:
        ClassOfRepresentationOfThing "1970-01-01T00:00:00"^^xsd:dateTime
```

Экземпляр шаблона **Beginning**, соответствующий высказыванию (утверждению):
«Начальным событием для объекта **Alfred** является событие **_1970-01-01**», которое

представляется в среде RDF:

```
Individual: company1:Beginning_of_Alfred_statement
    Types:
        p7tpl:Beginning
    Facts:
        p7tpl:hasWhole company1:Alfred,
        p7tpl:hasPart company1:_1970-01-01
```

Шаблон **Beginning**, описанный выше, представляется далее с помощью экземпляров меташаблона.

Во-первых, принято, что шаблон **Beginning** имеет два аргумента.

```
Individual: p7tpl:TemplateDescription_of_Beginning

    Types:
        p7tm:TemplateDescription
    Facts:
        p7tm:hasTemplate p7tpl:Beginning,
        p7tm:valNumberOfRoles 2
```

Во-вторых, два другие экземпляра шаблона определяют, какие роли используются и их индекс (позицию внутри списка ролей шаблона).

```
Individual: p7tpl:TemplateRoleDescription_of_Beginning_1
    Types:
        p7tm:TemplateRoleDescription
    Facts:
        p7tm:hasTemplate p7tpl:Beginning,
        p7tm:hasRole p7tpl:hasPart,
        p7tm:hasRoleFillerType Event,
        p7tm:valRoleIndex 1
Individual: p7tm:TemplateRoleDescription_of_Beginning_2
    Types:
        p7tm:TemplateRoleDescription
    Facts:
        p7tm:hasTemplate p7tpl:Beginning,
        p7tm:hasRole p7tpl:hasWhole,
        p7tm:hasRoleFillerType PossibleIndividual,
        p7tm:valRoleIndex 2
```

H.3 Пример: информационная модель OIM

H.3.1 Описание

Модель OIM в настоящем примере представлена на «исходном» языке OWL. Это означает некоторую неточность сравнения использования шаблонов. Например, модель максимального расчетного давления **OIM:MaximumDesignPressure** имеет только значения **xsd:float** и не имеет единицы измерительного сегмента.

Модель OIM содержит указания, как построить специализированную модель шаблона. Она показывает, что шаблон «**identification**» — это отношение класса **PossibleIndividual.** Наследование упрощает идентификацию всех подклассов **PossibleIndividual**. Специализированный шаблон «**identification**» классифицирует идентификацию, определяет ее тип (например, **Tag Name** (название тега), **Line Number** (номер строки), **Serial Number** (серийный номер), **Activity Code** (код операции) или идентификатор RFID).

Модель OIM также показывает, как построить комплексные и внутренние отношения между различными OIM.

Отметим, что сущность «всасывание насоса» **PumpSuction** включает «процесс течения жидкости» **ProcessItemFluidStream**. Сущность **ProcessItemFluidStream,** относящаяся к технологическим режимам **ProcessCases,** может быть интерпретирована как пусковой режим «**Startup Case**» и расчетный режим «**Design Case**». Технологические режимы **ProcessCase** протекают в определенных технологических условиях **ProcessConditions,** которые интерпретируются

как минимальный (**Minimum**), нормальный (**Normal)** и максимальный (**Maximum**). Указанные технологические условия **ProcessConditions** определяют параметры процесса: давление, температуру, вязкость и т.д.

H.3.2 Манчестер-синтаксис

```
Class: rdl:PossibleIndividual
    Types:
        rdl:IndividualClass
    SubClassOf:
        rdl:Thing
        oim:hasIdentifier xsd:string

Class: rdl:CentrifugalPump
    SubClassOf:
        rdl:DynamicPump
    oim:Assembly
        some rdl:PumpImpeller
        some rdl:PumpSuction
        some rdl:PumpDischarge
    oim:DirectConnection some rdl:PumpDriver
    oim:MaximumDesignPressure xsd:float

Class: rdl:PumpImpeller
    SubClassOf:
        rdl:Impeller

Class: rdl:Impeller
    SubClassOf:
        rdl:Artefact
        oim:ImpellerDiameter xsd:float

Class: rdl:PumpDischarge
    SubClassOf:
        rdl:Artefact
        oim:Contains some rdl:ProcessItemFluidStream

Class: rdl:ProcessItemFluidStream
    SubClassOf:
        rdl:FluidStream
        oim:InvolvedIn
        rdl:ProcessItemFluidStreamCase some rdl:ProcessCase

Class: rdl:ProcessCase
    SubClassOf:
        rdl:Processing
        oim:InvolvedIn
        rdl:ProcessItemFluidCondition some rdl:ProcessCondition

Class: rdl:ProcessCondition
    SubClassOf:
        rdl:Processing
        oim:Pressure xsd:float
        oim:MassFlow xsd:float
        oim:Temperature xsd:float
        oim:Viscosity xsd:float
```

H.4 Пример: временные части

H.4.1 Общие положения

Настоящий пример представления временных частей соответствует ИСО 15926-2. Сначала показаны паттерны моделирования рассматриваемой конструкции по ИСО/ТС 15926-8, затем

рассмотрено конкретное представление моделей и шаблонов в среде RDF/OWL.

H.4.2 Обобщенное представление

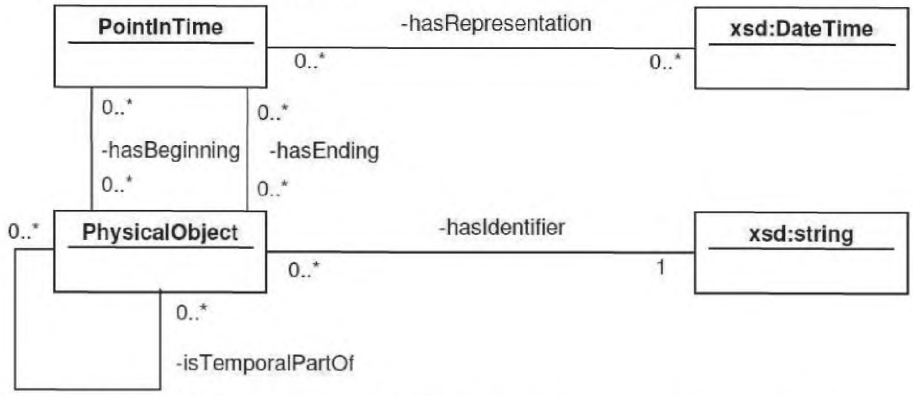

Рисунок H.1 — Обобщенные временные части, идентификация OIM

PointInTime	Момент времени
hasRepresentation	Представление
hasBeginning	Начальное событие
hasEnding	Конечное событие
PhysicalObject	Физический объект
hasIdentifier	Идентификатор
xsd:string	Строка
isTemporalPartOf	Временная часть

Диаграмма UML на рисунке H.I иллюстрирует информационную модель объекта для обобщенных аспектов временных частей, начального и конечного событий. Здесь также назначены имена:

— **PhysicalObject** может быть временной частью другого объекта;

— **PhysicalObject** может относиться и к начальному событию, и к конечному событию типа **PointInTime.** Указанные события могут быть представлены датами (типа *xsd:dateTime*);

— **PhysicalObject** может быть поименован строкой (типа *xsd:string*).

H.4.3 Специализированный шаблон IdentificationOfPhysicalObject (идентификация физического объекта)

Паттерны моделирования, приведенные на рисунке H.1 используются для описания индивидуальных объектов, выражающих информацию, которая используется при моделировании (позволяет инстанцировать модель). Шаблоны удобны для рассмотрения паттернов инстанцирования. В качестве примера рассмотрен специализированный шаблон **IdentificationOfPhysicalObject**.

Это пятиместный шаблон, утверждающий, что физический объект (x_1) имеет временную часть (x_2), появляющуюся в определенный момент времени (x_3), и имеющую определенный тип (x_4) имени (x_5).

Примечание 1 — Информационная модель OIM поясняет выбранный метод конструирования шаблона. Более широкая онтология (библиотека справочных данных RDL) служит базисом, из которого извлекается модель OIM, чтобы сфокусироваться на объектах особого вида. Затем выбирается шаблон для использования паттернов (образцов) при построении модели OIM.

Сигнатура шаблонов **rdl:IdentificationOfPhysicalObject** выглядит следующим образом:

№	Название роли	Тип роли
1	*p7tpl:hasWhole*	*dm:PhgysicalObject*
2	*p7tpl:hasPart*	*dm:PhysicalObject*
3	*p7tpl: valStartTime*	*xsd:dateTime*
4	*p7tpl:hasContext*	*dm:ClassOfClassOfIdentification*
5	*p7tpl: valIdentifier*	*xsd:string*

Примечание 2 — **IdentificationOfPhysicalObject** — это шаблон, аксиома которого содержит класс шаблонов **ClassifiedIdentification** и **BeginningOfIndividual**. См. ИСО/ТС 15926-7.

Примечание 3 — Роль **hasContext**, представляющая тип идентификации, не имеет элемента модели, представленного на рисунке H.I. Дело в том, что не существует символа языка UML, представляющего подтипы отношений. Это означает, что диаграмма не может отразить целевые аспекты, и представление диаграммы является неполным. Для получения полной интерпретации шаблона, следует использовать аксиому шаблона в соответствии с ИСО 19526-7. Выраженная в терминах воплощенного формата (с использованием всего множества конструктивов ИСО 15926-2), указанная аксиома дает полное явное представление.

H.4.4 Специализированный шаблон Identification Of Physical Object-process plant (идентификация физического объекта — технологическое оборудование

Ниже рассмотрен пример специализированной информационной модели OIM и соответствующий шаблон. На рисунке H.2 приведена модель, являющаяся ограниченной версией модели, показанной на рисунке H.1. Фокусом настоящей информационной модели OIM являются временные части технологического оборудования **ProcessPlant**. Здесь именование временных частей ограничено назначениями **ProcessPlantName** (в отличие от других имеющихся типов именования).

Примечание 1 — Объект настоящей информационной модели OIM представлен классом **ProcessPlant** в середине диаграммы. Здесь достаточно использовать традиционные средства, например, цветные ячейки классов, чтобы выделить то, что находится в фокусе.

Для ограниченных моделей OIM добавляются ограничения кардинальности, которых не было в более обобщенных моделях. В соответствии с рисунком H.1, временная часть рассматриваемого технологического оборудования **ProcessPlant** является, по крайней мере, частью какого-либо другого **ProcesPlant**. Более того, рассматриваемой временной части назначается строго один идентификатор **ProcessPlantName** (типа *xsd:string*) и строго один шаблон **Beginning**, для которого имеется только одна временная отметка (типа *xsd:dateTime*).

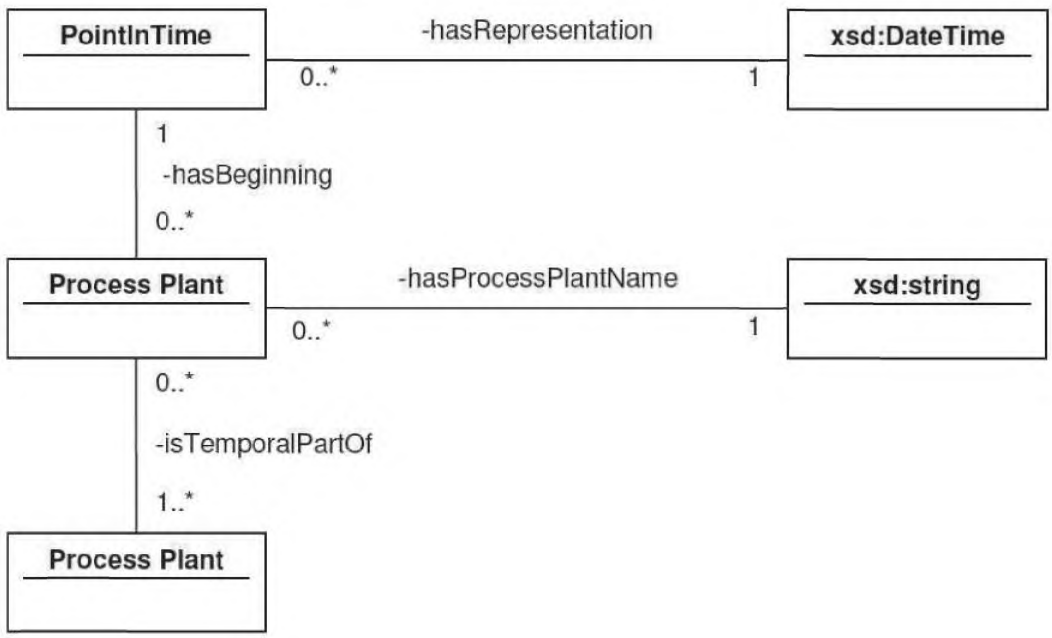

Рисунок H.2 — Временная часть ProcessPlant, наименование модели OIM

Также как одна информационная модель OIM является ограниченной версией другой модели, шаблон **IdentificationOfPhysicalObjectprocessplant** представлен как специализация шаблона **IdentificationOfPhysicalObject.** Для выполнения специализации, первые две роли шаблона **IdentificationOfPhysicalObject** ограничены классом **ProcessPlant,** а четвертая роль шаблона **IdentificationOfPhysicalObject** зафиксирована на значении **ProcessPlantName** (*rdl:R1234*).

Примечание 2 — Роль 4 зафиксирована на константу путем утверждения, что тип задается одноэлементным множеством **ProcessPlantName**, то есть множеством, содержащим только элемент **ProcessPlantName**. Требования для фиксированных значений остаются неизменными, так как тип роли имеет только один элемент, и любой правильный экземпляр шаблона будет обязан иметь **ProcessPlantName** в роли 4.

№	Название роли	Тип роли
1	*p7tpl:hasWhole*	*rdl:ProcessPlant*
2	*p7tpl:hasPart*	*rdl:ProcessPlant*
3	*p7tpl: valStartTime*	*xsd:dateTime*
4	*p7tpl:hasContext*	*rdl:ProcessPlantName*
5	*p7tpl:valIdentifier*	*xsd: string*

Примечание 3 — Ссылки на идентификаторы в настоящем примере являются вымышленными. Идентификаторы выбраны из соображений удобства прочтения.

H.5 Специализированный шаблон и экземпляр шаблона

Пусть **t100101 —** это экземпляр шаблона **IdentificationOfPhysicalObject-processplant**. Тогда:

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
    <!ENTITY xsd "gost.e-ecolog.ru                                    ">
    <!ENTITY owl "gost.e-ecolog.ru                                    ">
    <!ENTITY rdf "gost.e-ecolog.ru                                        ">
    <!ENTITY rdfs "gost.e-ecolog.ru                                     ">
    <!ENTITY dm "gost.e-ecolog.ru                                                                                    ">
    <!ENTITY p7tm "gost.e-ecolog.ru                                                                                ">
    <!ENTITY p7tpl "gost.e-ecolog.ru                                                                               ">
    <!ENTITY meta "gost.e-ecolog.ru                                                                                  ">
    <!ENTITY rdl "gost.e-ecolog.ru                  ">
    <!ENTITY steplib "gost.e-ecolog.ru                    ">
    <!ENTITY pca "gost.e-ecolog.ru                      ">
    <!ENTITY company1 "gost.e-ecolog.ru                 ">
]>
<rdf:RDF xmlns="gost.e-ecolog.ru                  "
    xml:base="gost.e-ecolog.ru                "
    xmlns:owl="&owl;"
    xmlns:xsd="&xsd;"
    xmlns:rdf="&rdf;"
    xmlns:rdfs="$rdfs;"
    xmlns:dm="&dm;"
    xmlns:p7tm="&p7tm;"
    xmlns:p7tpl="&p7tpl;"
    xmlns:meta="&meta;"
    xmlns:rdl="$rdl;"
    xmlns:steplib="&steplib;"
    xmlns:pca="&pca;"
    xmlns:company1="&company1;"
>

<rdl:IdentificationOfPhysicalObject-processplant rdf:ID="#t100101">
    <meta:annUniqueName rdf:datatype="&xsd;string">
        pcs125425 identified by process plant B14 since 2009-12-16T14:28:00Z
    </meta:annUniqueName>
    <meta:annRule rdf:datatype="&xsd;string">#com1_26539436548</meta:annRule>
    <meta:annAccessCode rdf:datatype="&xsd;string">#com1_836379</meta:annAccessCode>
    <p7tpl:hasWhole rdf:resource="#pcs125425"/>
    <p7tpl:hasPart rdf:resource="#pcs125789"/>
    <p7tpl:valStartTime rdf:datatype="&xsd;dateTime">
        2009-12-16T14:28:00Z
    </p7tpl:valStartTime>
```

```
    <p7tpl:hasContext rdf:resource="&rdl;ProcessPlantName"/>
    <p7tpl:valIdentifier rdf:datatype="&xsd;string">
        B14
    </p7tpl:valIdentifier>
</rdl:IdentificationOfPhysicalObject-processplant>

</rdf:RDF>
```

Настоящий пример показывает строку идентификации B14 особого технологического оборудования как экземпляра шаблона.

Примечание 1 — Символьные константы **valStartTime** и **valIdentifier** не могут моделироваться как типы языка EXPRESS. Они моделируются как символьные константы RDF. Как следует из опыта использования языка OWL, это не всегда соответствует требованиям ИСО 15926-2 к символьным константам языка EXPRESS.

Примечание 2 — Рассматриваемый шаблон содержит временные конструктивы целого (частей), моделирующие идентификацию как временную часть. Моделирование аспекта как временной части не является обязательным, но это наилучший способ создания модели, практически оправданный. Иногда полезно моделировать аспекты как временные части, иногда этого делать не следует. Иногда возможно использовать временные части при пересмотре аспекта.

Дополнительные примеры:

Базовая область имен «**company1**» (например, база данных проекта):

```
<rdl:ProcessPlant rdf:ID="#pcs125425">
    <meta:annUniqueName rdf:datatype="&xsd;string">
        process plant B14
    </meta:annUniqueName>
    <meta:annAccessCode rdf:datatype="&xsd;string">#com1_836379</meta:annAccessCode>
</rdl:ProcessPlant>

<rdl:ProcessPlant rdf:ID="#pcs125789">
    <meta:annUniqueName rdf:datatype="&xsd;string">
        temporal part - identified by process plant
    </meta:annUniqueName>
    <meta:annAccessCode rdf:datatype="&xsd;string">#com1_836379</meta:annAccessCode>
</rdl:ProcessPlant>
```

Базовая область имен "**rdl**" (основная библиотека RDL):

```
<dm:ClassOfInanimatePhysicalObject rdf:ID="#ProcessPlant">
    <rdfs:subClassOf rdf:resource="#PhysicalObject"/>
    <meta:annUniqueName rdf:datatype="&xsd;string" xml:lang="en">
        PROCESS PLANT
    </meta:annUniqueName>
    <meta:annUniqueName rdf:datatype="&xsd;string" xml:lang="nl">
        PROCESS FABRIEK
    </meta:annUniqueName>
    <meta:annTextDefinition rdf:datatype="&xsd;string" xml:lang="en">
        A plant employed in carrying out chemical processes,
        including the required supporting processes.
    </meta:annTextDefinition>
    <meta:annTextDefinition rdf:datatype="&xsd;string" xml:lang="nl">
        Een fabriek voor het verrichten van chemische processen,
        inclusief de vereiste ondersteunende processen.
    </meta:annTextDefinition>
</dm:ClassOfInanimatePhysicalObject>
```

Для данного экземпляра основной шаблон в среде OWL имеет вид:
Базовая область имен «**rdl**» (основная библиотека RDL):

```
<p7tpl:Template rdf:ID="#IdentificationOfPhysicalObject">
    <meta:annUniqueName xml:lang="en">Identification of Physical Object</meta:annUniqueName>
    <rdfs:subClassOf>
        <owl:Class>
            <owl:intersectionOf rdf:parseType="Collection">
                <owl:Restriction>
                    <owl:onProperty rdf:resource="&p7tpl;hasWhole"/>
                    <owl:allValuesFrom rdf:resource="&dm;PhysicalObject"/>
                </owl:Restriction>
                <owl:Restriction>
                    <owl:onProperty rdf:resource="&p7tpl;hasWhole"/>
                    <owl:onClass rdf:resource="&dm;PhysicalObject"/>
                    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">
                    1
                    </owl:qualifiedCardinality>
                </owl:Restriction>
            </owl:intersectionOf>
        </owl:Class>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Class>
            <owl:intersectionOf rdf:parseType="Collection">
                <owl:Restriction>
                    <owl:onProperty rdf:resource="&p7tpl;hasPart"/>
                    <owl:allValuesFrom rdf:resource="&dm;PhysicalObject"/>
                </owl:Restriction>
                <owl:Restriction>
                    <owl:onProperty rdf:resource="&p7tpl;hasPart"/>
                    <owl:onClass rdf:resource="&dm;PhysicalObject"/>
                    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">
                    1
                    </owl:qualifiedCardinality>
                </owl:Restriction>
            </owl:intersectionOf>
        </owl:Class>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="&p7tpl;valStartTime"/>
            <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
        </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
        <owl:Class>
            <owl:intersectionOf rdf:parseType="Collection">
                <owl:Restriction>
                    <owl:onProperty rdf:resource="&p7tpl;hasContext"/>
                    <owl:allValuesFrom rdf:resource="&dm;ClassOfClassOfIdentification"/>
                </owl:Restriction>
                <owl:Restriction>
                    <owl:onProperty rdf:resource="&p7tpl;hasContext"/>
                    <owl:onClass rdf:resource="&dm;ClassOfClassOfIdentification"/>
                    <owl:qualifiedCardinality rdf:datatype="&xsd;nonNegativeInteger">
                    1
                    </owl:qualifiedCardinality>
                </owl:Restriction>
            </owl:intersectionOf>
```

```
            </owl:Class>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Restriction>
                <owl:onProperty rdf:resource="&p7tpl;valIdentifier"/>
                <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
            </owl:Restriction>
        </rdfs:subClassOf>
    </p7tpl:Template>
```

Примечание 3 — Это специализированный шаблон, обозначенный как «**p7tpl:Template**» и относящийся к классу *owl:Class*.

Для данного экземпляра специализированный шаблон в среде OWL имеет вид:
Базовая область имен «**rdl**» (основная библиотека RDL):

```
    <p7tpl:Template rdf:ID="#IdentificationOfPhysicalObject-processplant">
        <meta:annUniqueName xml:lang="en">Identification of Physical
        Object</meta:annUniqueName>
        <rdfs:subClassOf rdf:resource="#IdentificationOfPhysicalObject"/>
        <rdfs:subClassOf>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="&p7tpl;hasWhole"/>
                        <owl:allValuesFrom rdf:resource="&rdl;ProcessPlant"/>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="&p7tpl;hasPart"/>
                        <owl:allValuesFrom rdf:resource="&rdl;ProcessPlant"/>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
            <owl:Class>
                <owl:intersectionOf rdf:parseType="Collection">
                    <owl:Restriction>
                        <owl:onProperty rdf:resource="&p7tpl;hasContext"/>
                        <owl:allValuesFrom rdf:resource="&rdl;ProcessPlantName"/>
                    </owl:Restriction>
                </owl:intersectionOf>
            </owl:Class>
        </rdfs:subClassOf>
    </p7tpl:Template>
```

Библиография

[1] ИСО/МЭК 8824-1:2008 (ISO/IEC 8824-1:2008)	Информационные технологии. Нотация абстрактного синтаксиса версии 1 (ASN.1). Часть 1. Спецификация базовой нотации (Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation)
[2] ИСО 10303-1:1994 (ISO 10303-1:1994)	Системы промышленной автоматизации и интеграция. Представление данных о продукции и обмен данными. Часть 1. Обзор и основные принципы (Industrial automation systems and integration — Product data representation and exchange — Part 1: Overview and fundamental principles)
[3] ИСО 10303-11:2004 (ISO 10303-11:2004)	Системы промышленной автоматизации и интеграция. Представление данных о продукции и обмен данными. Часть 11. Методы описания. Справочное руководство по языку EXPRESS (Industrial automation systems and integration — Product data representation and exchange — Part 11: Description methods: The EXPRESS language reference manual)
[4] ИСО 10303-21:2002 (ISO 10303-21:2002)	Системы промышленной автоматизации и интеграция. Представление данных о продукции и обмен данными. Часть 21. Методы реализации. Кодирование открытого текста структуры обмена (Industrial automation systems and integration — Product data representation and exchange — Part 21: Implementation methods: Clear text encoding of the exchange structure)
[5] ИСО 15926-1:2004 (ISO 15926-1:2004)	Системы промышленной автоматизации и интеграция. Интеграция данных жизненного цикла перерабатывающих предприятий, включая нефтяные и газовые производственные предприятия. Часть 1. Общее представление и основные принципы (Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities — Part 1: Overview and fundamental principles)
[6] ИСО 15926-2:2003 (ISO 15926-2:2003)	Системы промышленной автоматизации и интеграция. Интеграция данных жизненного цикла перерабатывающих предприятий, включая нефтяные и газовые производственные предприятия. Часть 2. Модель данных (Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities — Part 2: Data model)
[7] ИСО/ТС 15926-3:2009 (ISO/TS 15926-3:2009)	Системы промышленной автоматизации и интеграция. Интеграция данных жизненного цикла перерабатывающих предприятий, включая нефтяные и газовые производственные предприятия. Часть 3. Справочные данные, касающиеся геометрии и топологии (Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities — Part 3: Reference data for geometry and topology)
[8] ИСО/ТС 15926-4:2007 (ISO/TS 15926-4:2007)	Системы промышленной автоматизации и интеграция. Интеграция данных жизненного цикла перерабатывающих предприятий, включая нефтяные и газовые производственные предприятия. Часть 4. Исходные справочные данные (Industrial automation systems and integration — Integration of lifecycle data for process plants including oil and gas production facilities — Part 4: Initial reference data)
[9] ИСО/ТС 15926-7:2011	Системы промышленной автоматизации и интеграция. Интеграция данных жизненного цикла перерабатывающих

(ISO/TS 15926-7:2011)	предприятий, включая нефтяные и газовые производственные предприятия. Часть 7. Практические методы интеграции распределенных систем: методология шаблонов. Методология шаблонов (Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities — Part 7: Implementation methods for the integration of distributed systems: Template methodology)
[11] OWL 2 web ontology language — structural specification and functional-style syntax. [online]. W3C recommendation 2009-10-27. Available from World Wide Web: <gost.e-ecolog.ru >	
[12] OWL 2 web ontology language — quick reference guide. [online]. W3C recommendation 2009-10-27. Available from World Wide Web: <gost.e-ecolog.ru >.	
[13] RDF vocabulary description language 1.0: RDF schema. [online]. W3C recommendation 2004-02-10. Available from World Wide Web: <gost.e-ecolog.ru >.	
[14] RDF/XML syntax specification (revised). [online]. W3C recommendation 2004-02-10. Available fromWorld WideWeb: <gost.e-ecolog.ru >	
[15] Resource description framework (RDF): concepts and abstract syntax. [online]. W3C recommendation 2004-02-10. Available from World Wide Web: <gost.e-ecolog.ru gost.e-ecolog.ru >	
[16] RFC 2396 uniform resource identifiers (URI): generic syntax. [online]. IETF, 1998-08. Available from World Wide Web: <gost.e-ecolog.ru >	
[17] SPARQL protocol for RDF (SPROT). [online]. W3C recommendation 2008-01-15. Available from World Wide Web: <gost.e-ecolog.ru >	
[18] XML schema part 2: datatypes second edition. [online]. W3C recommendation 2004-10-28. Available from World Wide Web: <gost.e-ecolog.ru >	
[19] BERNERS-LEE, Tim. Linked Data. [online]. Status: personal view only. Editing status: imperfect but published. Available from World Wide Web: <gost.e-ecolog.ru >	
[20] BRAY, Tim, PAOLI Jean, SPERBERG-MCQUEEN, C.M., MALER, Eve, YERGEAU, Fran∟ ois, COWAN, John. Extensible markup language (XML) 1.1 (second edition). [online]. W3C recommendation 2006-08-16, edited in place 2006-09-26. Available from World Wide Web: <gost.e-ecolog.ru >	
[21] HORRIDGE, Matthew and PATER-SCHNEIDER, Peter F. OWL 2 web ontology language Manchester syntax. [online]. W3C working group note 2009-10-27. Available from World Wide Web: <gost.e-ecolog.ru >	
[22] NOY, Natasha, RECTOR, Alan, HAYES, Pat, WELTY, Chris. Defining n-ary relations on the semantic web (refer pattern 1). [online]. W3C working group note 2006-04-12. Available from World Wide Web: <gost.e-ecolog.ru >	
[23] SCHLIMMER, Jerey C. Web services description requirements. [online]. W3C working draft 2002-10-28. Available from World Wide Web: <gost.e-ecolog.ru >	
[24] Graph stylesheets (GSS) in IsaViz. [online]. Available from World Wide Web: <gost.e-ecolog.ru >	
[25] ИСО/МЭК 11179-1:2004 (ISO/IEC 11179-1:2004)	Информационные технологии. Реестры метаданных (MDR). Часть 1. Структура Information technology — Metadata registries (MDR) — Part 1: Framework

УДК 658.52.011.56 ОКС 25.040.40; 75.020

Ключевые слова: системы промышленной автоматизации, интеграция, жизненный цикл систем, управление производством, интеграция данных жизненного цикла перерабатывающих предприятий, шаблон индивидуального объекта, расширение шаблона, протошаблоны, сетевой язык онтологий (OWL), среда описания ресурса (RDF)

Подписано в печать 02.03.2015. Формат 60x84 1/8.
Усл. печ. л. 5.58. Тираж 31 экз. Зак. 1343.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

ФГУП «СТАНДАРТИНФОРМ»

123995 Москва, Гранатный пер., 4.

gost.e-ecolog.ru

gost.e-ecolog.ru