

**НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Система стандартов информационного моделирования зданий и сооружений**

**СТРОИТЕЛЬСТВО ЗДАНИЙ**

**Структура информации об объектах строительства**

**Часть 3**

**Основы обмена объектно-ориентированной информацией**

**System of standards on information modeling of buildings and structures. Building construction. Organization of information about construction works. Part 3. Framework for object-oriented information**

ОКС 91.010.01  
35.240.67  
35.240.01

Дата введения 2019-09-01

**Предисловие**

1 ПОДГОТОВЛЕН Ассоциацией организаций по развитию технологий информационного моделирования в строительстве и ЖКХ (БИМ-Ассоциация) на основе собственного перевода на русский язык англоязычной версии стандарта, указанного в пункте 4

2 ВНЕСЕН Проектным техническим комитетом по стандартизации ПТК 705 "Технологии информационного моделирования на всех этапах жизненного цикла объектов капитального строительства и недвижимости"

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 5 июня 2019 г. N 282-ст

4 Настоящий стандарт идентичен международному стандарту ИСО 12006-3:2007\* "Строительство зданий. Структура информации об объектах строительства. Часть 3. Основы обмена объектно-ориентированной информацией" (ISO 12006-3:2007 "Building construction - Organization of information about construction works - Part 3: Framework for object-oriented information", IDT).

Наименование настоящего стандарта изменено относительно наименования указанного международного стандарта для приведения в соответствие с [ГОСТ Р 1.5-2012](#) (пункт 3.5).

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

5 ВЗАМЕН [ГОСТ Р ИСО 12006-3-2017](#)

*Правила применения настоящего стандарта установлены в статье 26 Федерального закона от 29 июня 2015 г. N 162-ФЗ "О стандартизации в Российской Федерации". 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 ежемесячного 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 ([www.gost.ru](http://www.gost.ru))*

## **Введение**

Основную часть настоящего стандарта составляет описание спецификации таксономической модели, позволяющей определять понятия по их свойствам, группировать понятия и определять отношения между ними. Основными элементами этой модели являются объекты, коллекции объектов и их отношения (взаимосвязи). Набор свойств, присущих определенному объекту, обуславливает его формальное определение и типичное поведение. Свойства обладают значениями, которые могут выражаться количественно в единицах измерения.

Через модель определяется роль, которую должен играть объект, что дает возможность выявить контекст, в котором этот объект используется. Каждый объект может иметь несколько имен, что позволяет выражать его несколькими синонимами или на нескольких языках. Языковое имя каждого объекта всегда должно указываться на английском языке, служащим языком, используемым по умолчанию. Также объект можно назвать на языке той местности, на которой он определяется или используется. С помощью ссылок объекты могут быть связаны с формальными системами классификации.

В модели есть одна корневая сущность, из которой наследуются следующие три сущности, являющиеся ее подтипами: объекты, коллекции и отношения между ними. Корневая сущность позволяет назначать производным типам любой набор имен, меток, описаний и ссылок на любом языке, а также присваивать идентификаторы и даты.

Все объекты подразделяются на субъекты, действия, акторы, единицы, значения и меры с единицами и свойствами. Субъекты и действия - это описываемые вещи и процессы. Остальные объекты являются описательными сущностями, связанными с другими объектами через отношения.

Отношения обеспечивают механизм связи между объектами. Все отношения подразделяются на следующие основные типы: сопоставления, коллекции, специализации, композиции, участия (воздействия на), присваивания свойств, последовательности и присваивания меры (единиц измерения).

Под коллекциями подразумеваются все виды группирования объектов, включая вложенные коллекции, определенные посредством набора отношений в коллекции.

Свойства являются сущностями, создающими контекст данных, хранящихся в виде значений. Все свойства различаются по типу данных, которые они содержат, среди которых: перечисляемые значения, значения списка, значения ограниченного списка, ограниченные значения, одиночные значения и табличные значения.

Количественное значение того или иного свойства выражается соответствующей единицей измерения и хранится в зависимости от языка программирования в определенной части программы. Другой компонент определяет способ представления любого имени, описания, значения или ссылки, свой для каждого языка.

Модель, описанная в настоящем стандарте, предлагается как связующее звено между системами классификации, описанными в ISO 12006-2 [5] и моделирование продуктов, описанных в нескольких публикациях [2], [3], [6], [7].

## **1 Область применения**

В настоящем стандарте описывается не зависящая от языка информационная модель, применяемая при разработке словарей для хранения или предоставления информации о результатах строительных работ.

Она позволяет ссылаться на системы классификации, информационные, объектные и процессные модели в рамках общей структуры.

## **2 Нормативные ссылки**

В настоящем стандарте использованы нормативные ссылки на следующие стандарты:

ISO 10303-11, Industrial automation systems and integration - Product data representation and exchange - Part 11: Description methods: The EXPRESS language reference manual (Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 11. Методы описания. Справочное руководство по языку EXPRESS)

ISO/IEC 10646, Information technology - Universal Coded Character Set (UCS) (Информационные технологии. Универсальный многобайтовый кодированный набор символов (UCS))

## **3 Языковая кодировка**

Вся информация, указанная как тип String (строкового типа) или разрешающая тип String, должна быть выражена с помощью набора символов UNICODE [8] согласно ИСО/МЭК 10646, желательно с использованием кодировки UTF-8, схемы кодирования UTF-8 и 8-битного формата преобразования UCS Transformation Format 8 [4].

## **4 Спецификация**

### **4.1 Общие положения**

В настоящем стандарте модель описывается с помощью языка определения данных EXPRESS согласно ИСО 10303-11.

Неформальное описание модели представлено в 4.2 в нотации EXPRESS-G.

Формальное описание модели в нотации EXPRESS представлено в 4.3 (краткое) и 4.4 (подробное).

### **4.2 Спецификация EXPRESS-G**

Неформальная спецификация EXPRESS-G в нотации EXPRES-G приведена на шести диаграммах (рисунки 1-6), каждая из которых иллюстрирует одну из частей модели. Все сущности на этих диаграммах описаны формально в 4.3.

На рисунке 1 представлена диаграмма верхнего уровня с xtdRoot, его атрибуты и производные типы xtdObject, xtdRelationship и xtdCollection.

На рисунке 2 представлена диаграмма xtdLanguageRepresentation с его производными типами xtdName и xtdDescription, которые являются атрибутами xtdRoot и его подтипов.

На рисунке 3 представлены выведенные из xtdRelationship типы отношений, используемые для установления возможных отношений между xtdObjects, xtdCollections и xtdExternalDocuments или их подтипами.

На рисунке 4 представлено присвоение xtdProperties на xtdObjects с помощью xtdRelAssignsProperties и xtdMeasureWithUnit с назначением значений с помощью xtdRelAssignsMeasures.

На рисунке 5 представлен список базовых типов, используемых в модели, и соответствующие типы EXPRESS.

На рисунке 6 подробно представлены xtdValue и xtdExternalDocument.

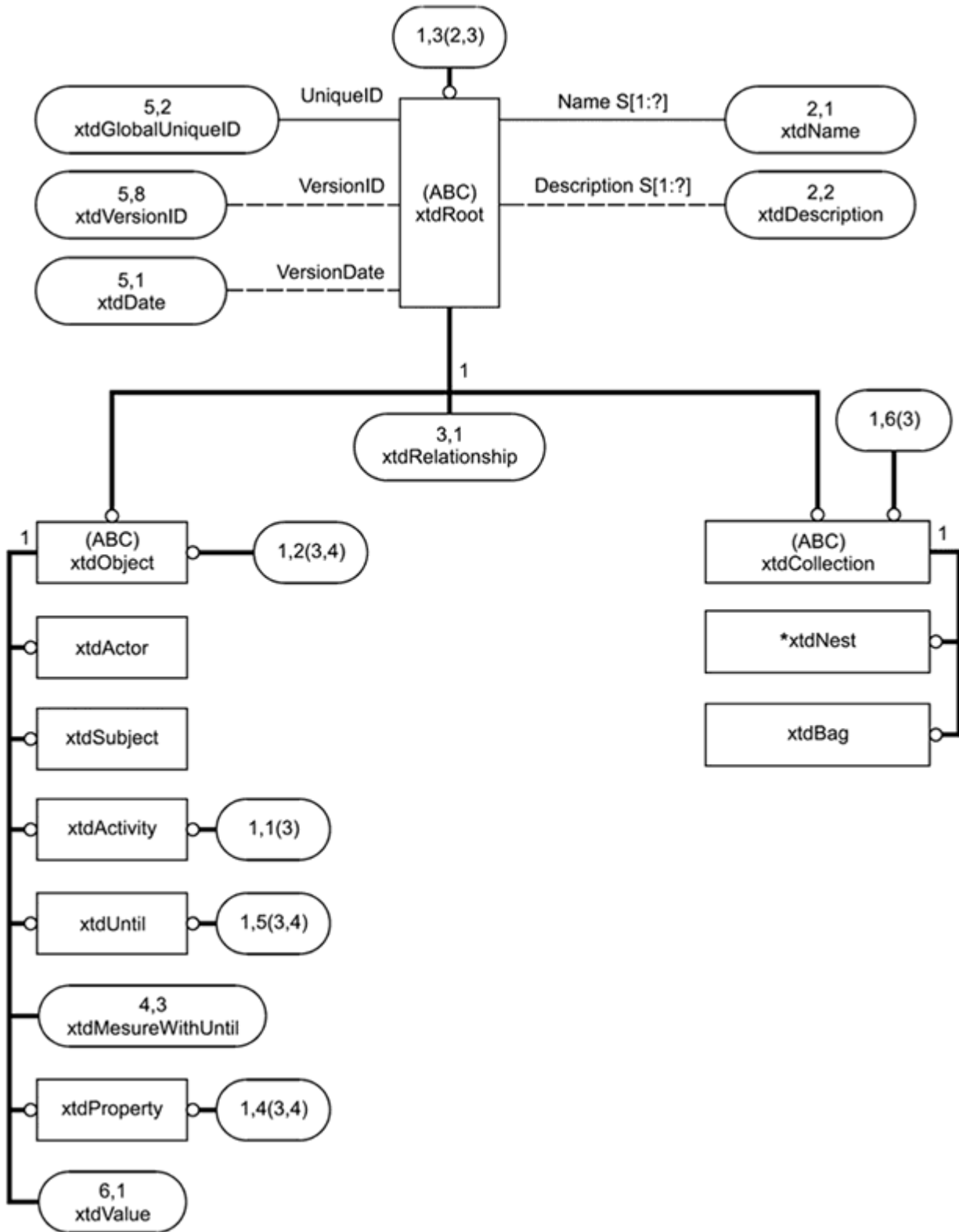


Рисунок 1 - Диаграмма 1 в нотации EXPRESS-G. Верхний уровень с корневой концепцией

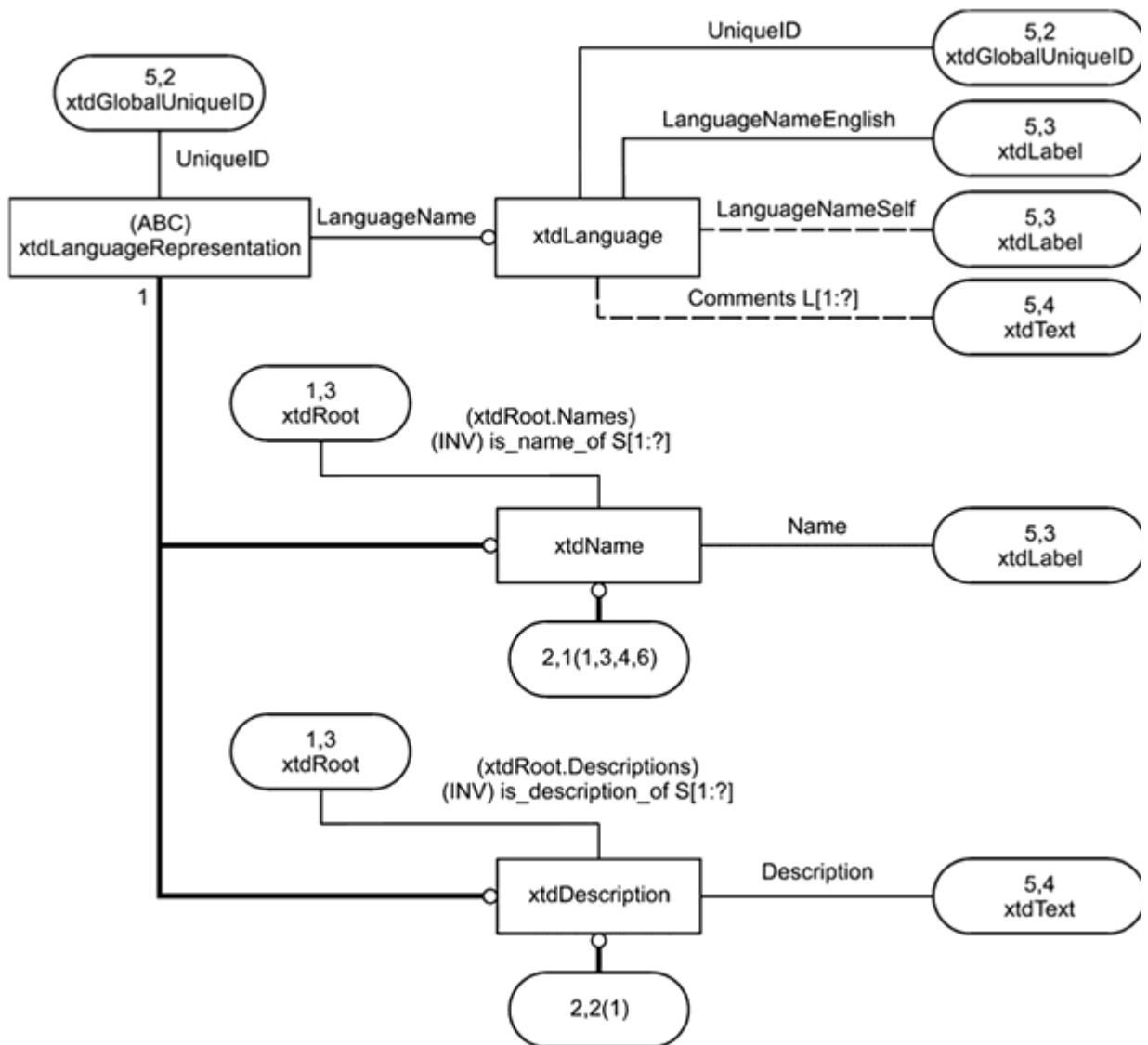


Рисунок 2 - Диаграмма 2 в нотации EXPRESS-G. Представление языка, наименования и описания

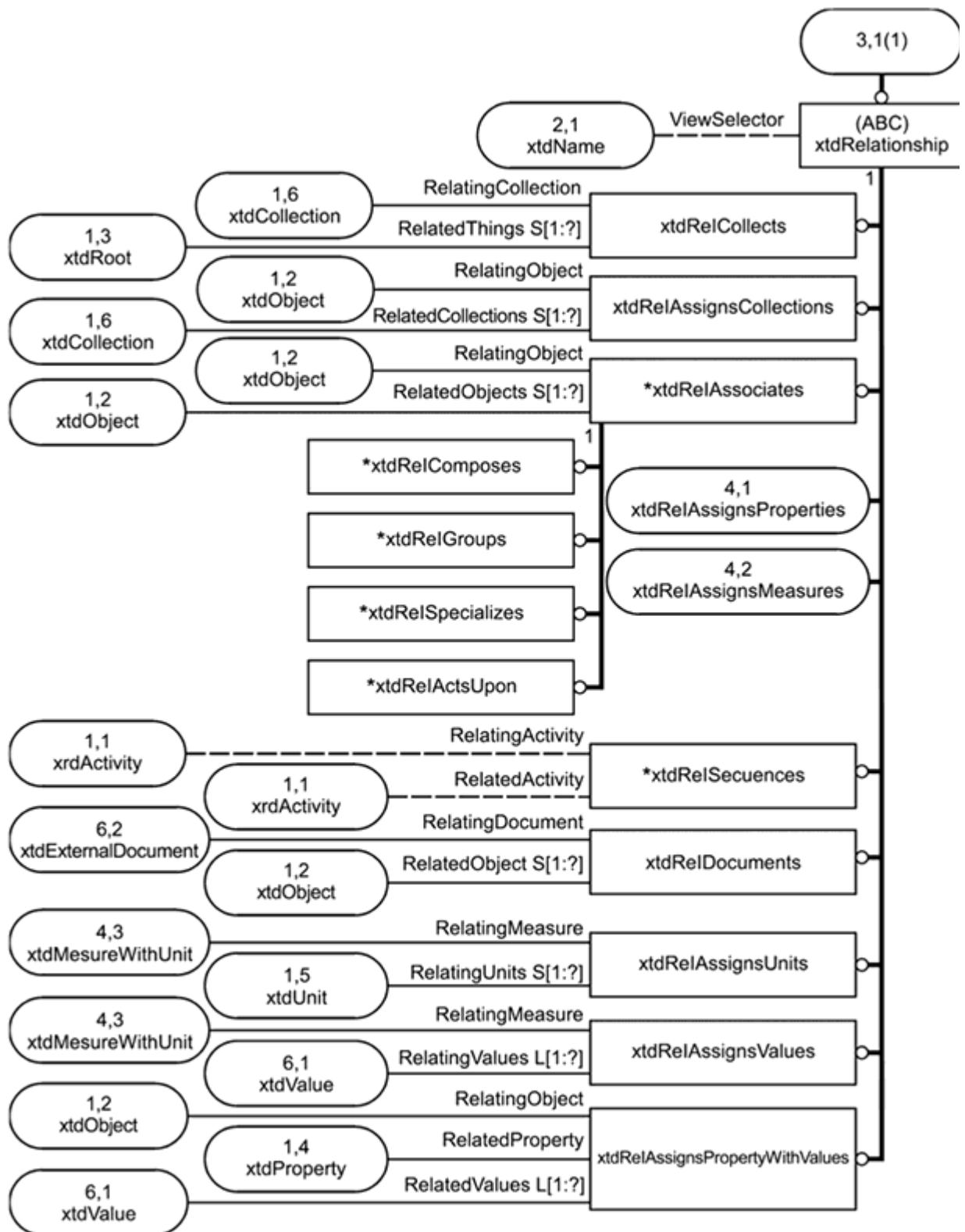


Рисунок 3 - Диаграмма 3 в нотации EXPRESS-G. Отношения

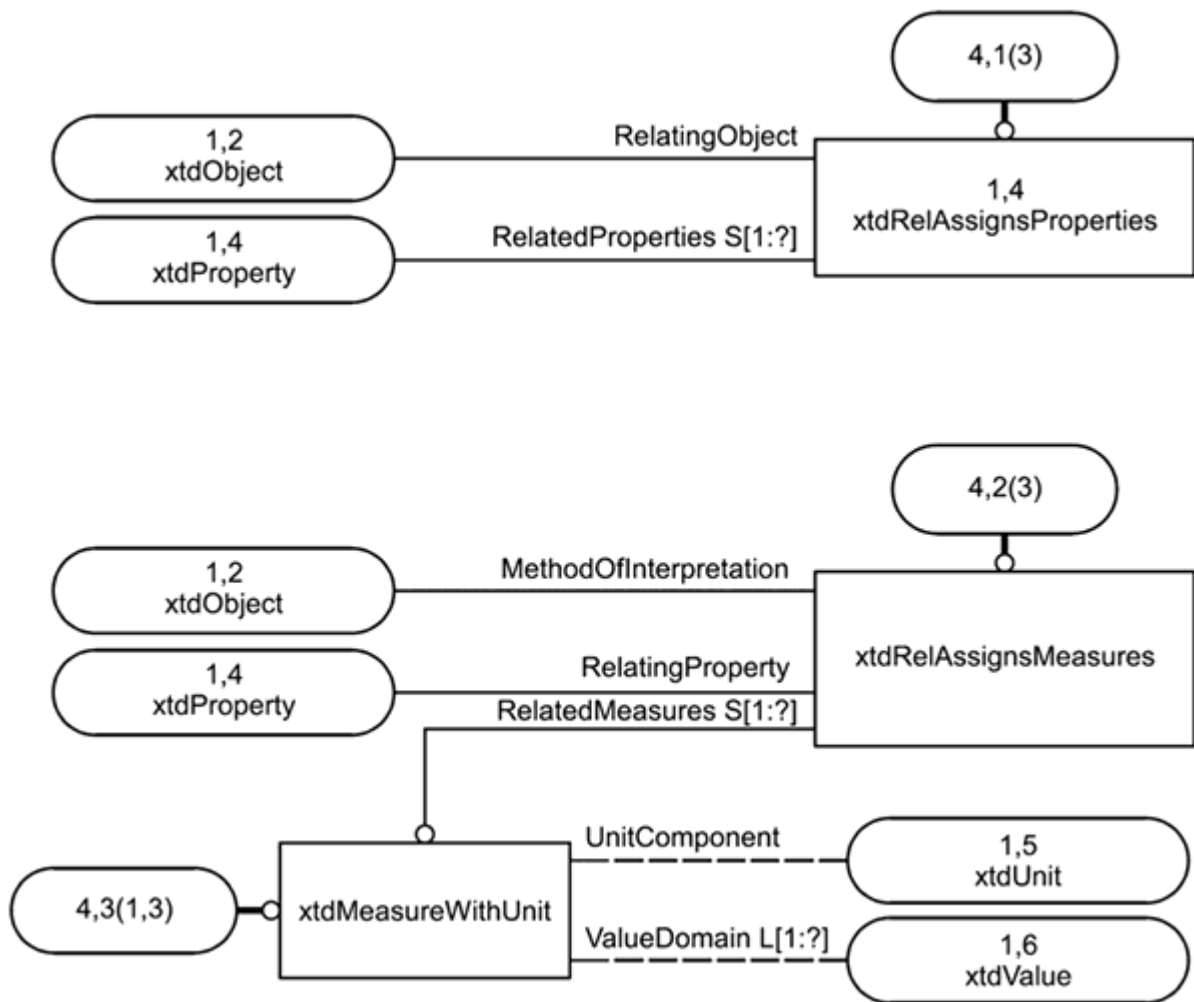


Рисунок 4 - Диаграмма 4 в нотации EXPRESS-G. Назначение свойств и единиц измерений

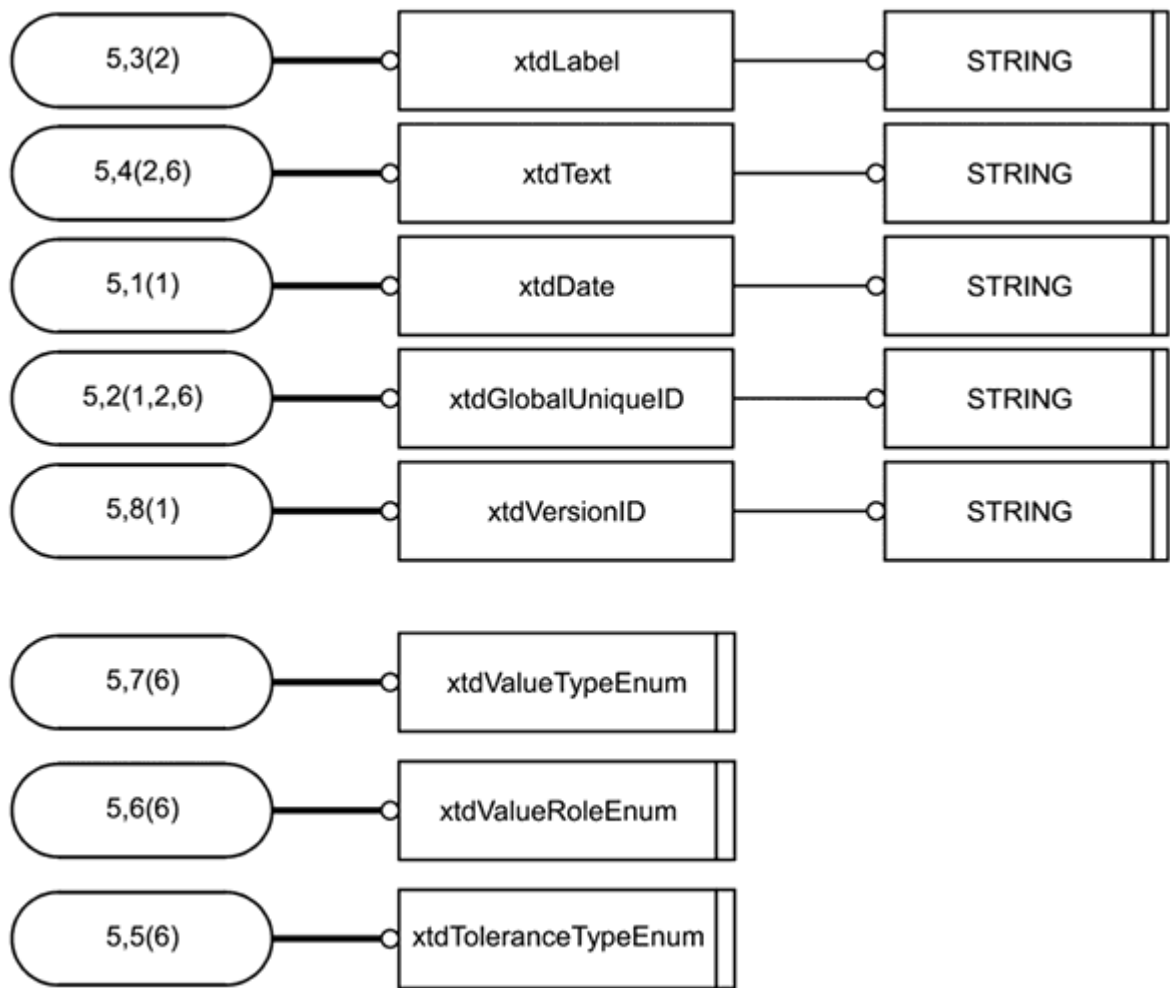


Рисунок 5 - Диаграмма 5 в нотации EXPRESS-G. Основные типы



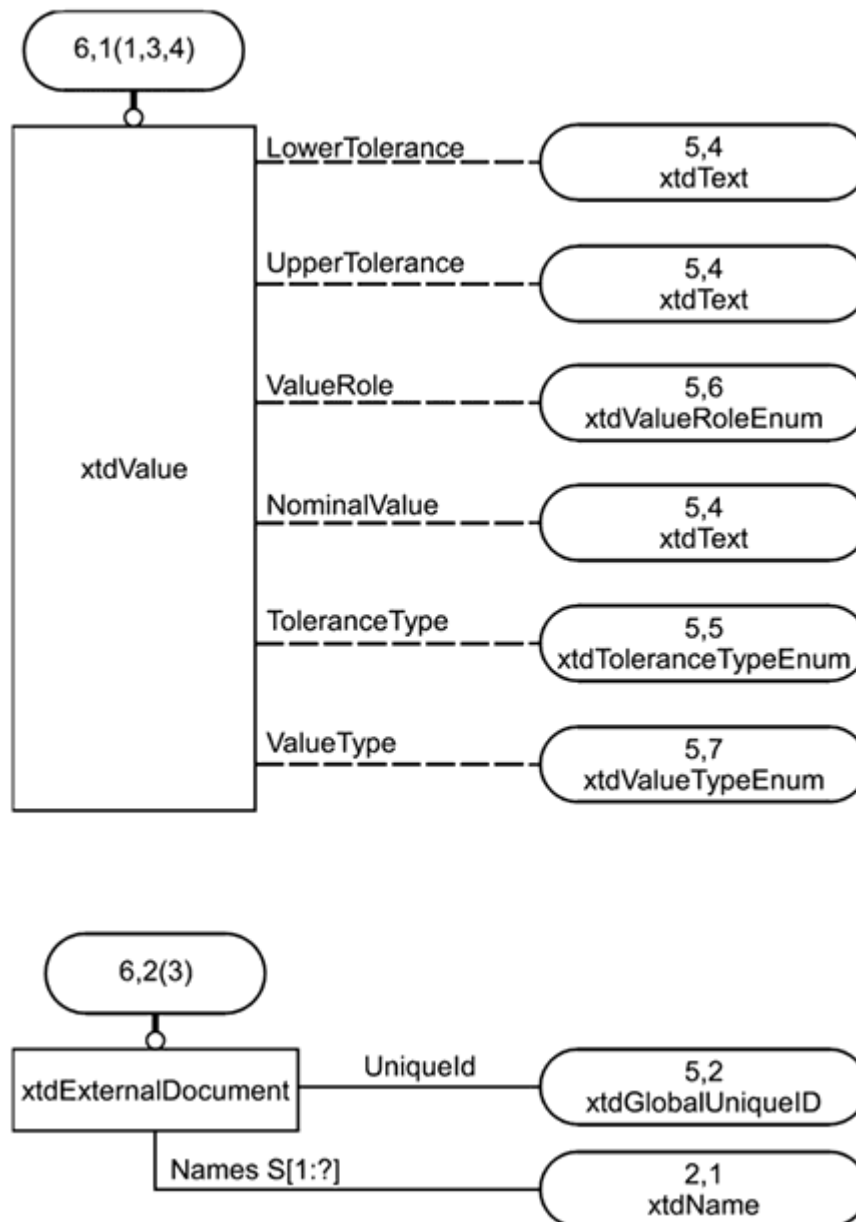


Рисунок 6 - Диаграмма 6 в нотации EXPRESS-G. Значения и внешние документы

#### 4.3 Спецификация EXPRESS

Данная формальная спецификация представлена на языке EXPRESS.

EXPRESS-спецификация:

(\*  
 ISO\_12006\_3\_VERSION\_3  
EXPRESS-спецификация:

\*)  
 SCHEMA ISO\_12006\_3\_VERSION\_3;

##### 4.3.1 xtdDate

xtdDate - определенный тип данных простого типа данных STRING, используемый для буквенно-

цифрового представления даты. Рекомендуемый формат представления даты - "YYYY.MM.DD"

**Пример - 31 мая 2000 года необходимо записывать как "2000.05.31"**

EXPRESS-спецификация:

```
*)  
TYPE xtdDate = STRING;  
END_TYPE;
```

```
(*  
4.3.2 xtdGlobalUniqueID
```

xtdGlobalUniqueID - определенный тип данных простого типа данных STRING, содержащий глобально уникальный идентификатор.

Обычно идентификатор создается с использованием алгоритма (например, опубликованного Object Management Group), основанного на IP-адресе компьютера, генерирующего этот идентификатор.

Существуют реализации указанного выше алгоритма, которые можно использовать для создания идентификатора.

Идентификатор представляет собой уникальное 128-битное число. Размер идентификатора рекомендуется сжимать для уменьшения объема служебных данных. Разработчикам следует сжимать идентификатор до строки из 22 символов с помощью алгоритма сжатия с 64 символами в качестве основного набора (кодировка BASE 64). Кодирование основного набора из 64 символов представлено ниже:

1	2	3	4	5	6
<u>0123456789012345678901234567890123456789012345678901234567890123</u>					
"0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz_\$";					

**Пример - 22-символьная строка "93f09e4A\_899402a9\$D013" является глобально уникальным идентификатором.**

Каждое понятие в библиотеке может иметь только один глобально уникальный идентификатор. Глобально уникальные идентификаторы похожих понятий в различных библиотеках, разрабатываемых независимо друг от друга, могут отличаться. В настоящем стандарте не представлен механизм, автоматически оценивающий похожие понятия с различными глобально уникальными идентификаторами из разных библиотек. Оценка понятий из разных библиотек в целях установления эквивалентности должна проводиться отдельно.

Примечание - Объяснение алгоритма представлено на <http://www.opengroup.org/dce/info/draft-leachuids-guids-01.txt>.

EXPRESS-спецификация:

```
*)  
TYPE xtdGlobalUniqueID = STRING;  
END_TYPE;
```

```
(*  
4.3.3 xtdLabel
```

xtdLabel - определенный тип данных простого типа данных STRING, представляющий собой набор буквенно-цифровых символов UNICODE для наименования объекта.

EXPRESS-спецификация:

```
*)  
TYPE xtdLabel = STRING;  
END_TYPE;
```

(  
**4.3.4 xtdText**

xtdText - определенный тип данных простого типа данных STRING, представляющий собой набор буквенно-цифровых символов UNICODE для аннотирования объекта.

EXPRESS-спецификация:

```
*)  
TYPE xtdText = STRING;  
END_TYPE;
```

(  
**4.3.5 xtdVersionID**

xtdVersionID - определенный тип данных простого типа данных STRING, представляющий собой набор буквенно-цифровых символов UNICODE для указания версии объекта.

*Пример - "1A", "12", "1.1" являются вариантами обозначения версий.*

EXPRESS-спецификация:

```
*)  
TYPE xtdVersionID = STRING;  
END_TYPE;
```

(  
**4.3.6 xtdToleranceTypeEnum**

xtdToleranceTypeEnum - тип перечислений, представляющий диапазон возможных типов допусков, которые можно применить к значению. Допустимые варианты:

Realvalue - определенный тип эквивалентного типа как значение, выражающее допуск как действительную величину;

Percentage - тип, определяющий процент, выражающий допуск в процентах от значения.

EXPRESS-спецификация:

```
*)  
TYPE xtdToleranceTypeEnum = ENUMERATION OF  
REALVALUE,  
PERCENTAGE);  
END_TYPE;
```

(  
**4.3.7 xtdValueRoleEnum**

xtdValueRoleEnum - тип перечислений, представляющий диапазон возможных типов интерпретации, применяемых к значению. Допустимые варианты:

Nominal - определенный тип, обозначающий значение номинального типа;

Maximum - определенный тип, обозначающий значение, как максимальное.

Minimum - определенный тип, обозначающий значение, как минимальное.

EXPRESS-спецификация:

```
*)  
TYPE xtdValueRoleEnum = ENUMERATION OF  
(NOMINAL,  
MAXIMUM,  
MINIMUM);  
END_TYPE;
```

#### **4.3.8 xtdValueTypeEnum**

xtdValueTypeEnum - перечисляемый тип, представляющий диапазон возможных типов простых значений, из которых можно выбирать. Допустимые варианты:

xtdString - определенный тип простого типа STRING для описания.

xtdNumber - определенный тип простого типа STRING.

xtdInteger - определенный тип простого типа INTEGER.

xtdReal - определенный тип простого типа REAL.

xtdBoolean - определенный тип простого типа BOOLEAN.

xtdLogical - определенный тип простого типа LOGICAL.

EXPRESS-спецификация:

```
*)  
TYPE xtdValueTypeEnum = ENUMERATION OF  
(XTDSTRING,  
XTDNUMBER,  
XTDINTEGER,  
XTDREAL,  
XTDBOOLEAN,  
XTDLOGICAL);  
END_TYPE;
```

#### **4.3.9 xtdActivity**

xtdActivity - специализация xtdObject, представляющая действие или процесс изменения xtdSubject.

***Пример - "Кладка кирпича" представляет собой действие или процесс кладки кирпичей и раствора для создания стены или других конструктивных элементов.***

EXPRESS-спецификация:

```
*)  
ENTITY xtdActivity  
SUBTYPE OF(xtdObject);  
END_ENTITY;
```

#### **4.3.10 xtdActor**

xtdActor - специализация xtdObject, представляющая актора, действующего на xtdSubject в пределах xtdActivity.

Актером может быть человек, должность, подразделение организации или оборудование.

**Пример 1 - "плотник" является xtdActor.**

**Пример 2 - "Международная организация по стандартизации" является xtdActor.**

**Пример 3 - "Кран грузоподъемный" может быть использован в качестве xtdActor.**

EXPRESS-спецификация:

```
*)  
ENTITY xtdActor  
SUBTYPE OF(xtdObject);  
END_ENTITY;
```

#### **4.3.11 xtdBag**

xtdBag - детализация xtdCollection, не имеющая ограничений на предметы, которые могут входить в эту коллекцию.

Коллекция акторов, действий и субъектов может быть собрана с помощью экземпляра xtdBag.

EXPRESS-спецификация:

```
*)  
ENTITY xtdBag  
SUBTYPE OF(xtdCollection);  
END_ENTITY;
```

#### **4.3.12 xtdCollection**

xtdCollection представляет группу или коллекцию предметов. xtdCollection может быть одной из xtdNest или xtdBag.

EXPRESS-спецификация:

```
*)  
ENTITY xtdCollection  
ABSTRACT SUPERTYPE OF (ONEOF(xtdNest, xtdBag))  
SUBTYPE OF(xtdRoot);  
END_ENTITY;
```

#### **4.3.13 xtdDescription**

xtdDescription содержит описание или определение предмета с помощью текста.

**Пример - "Створка двери - это открывающаяся часть двери, позволяющая людям проходить или перемещать предметы" - определительное описание объекта "створка двери" на русском языке.**

EXPRESS-спецификация:

```
*)  
ENTITY xtdDescription  
SUBTYPE OF(xtdLanguageRepresentation);  
Description :xtdText;  
INVERSE  
is_description_of :SET [1:?] OF xtdRoot FOR Descriptions;  
END_ENTITY;
```

Определение атрибута:

## **Description**

*Пример - xtdText, содержащий описание или определение.*

### **4.3.14 xtdExternalDocument**

xtdExternalDocument используется для представления внешних документов, книг или другой письменной информации.

*Пример 1 - "ISO 31" является документом.*

*Пример 2 - "NS 3420" является документом.*

EXPRESS-спецификация:

```
*)  
ENTITY xtdExternalDocument;  
UniqueID :xtdGlobalUniqueID;  
Names :SET [1:?] OF xtdName;  
END_ENTITY;
```

(\*  
Определения атрибутов:

#### **UniqueID**

Глобально уникальный идентификатор для внешнего документа.

#### **Names**

Набор зависящих от языка имен внешнего документа.

### **4.3.15 xtdLanguage**

xtdLanguage представляет язык, на котором выражаются имена, описания, значения и ссылки, включая собственное обозначение на этом языке и на английском языке.

EXPRESS-спецификация:

```
*)  
ENTITY xtdLanguage;  
LanguageNameInEnglish :xtdLabel;  
LanguageNameInSelf :OPTIONAL xtdLabel;  
Comments :OPTIONAL LIST [1:?] OF xtdText;  
UniqueID :xtdGlobalUniqueID;  
END_ENTITY;
```

(\*  
Определения атрибутов:

#### **LanguageNameInEnglish**

Название языка, на котором выражен идентифицирующий описательный атрибут, на английском языке.

Название языка должно интерпретироваться без учета регистра.

Примеры английских наименований языков в соответствии с ИСО 3166-1 [1] приведены ниже.

*Пример 1 - "English" - наименование ИСО 3166-1 для всех типов английского языка.*

*Пример 2 - "German" - английское наименование языка, на котором говорят в Германии и окружающих ее странах.*

#### **LanguageNameInSelf**

Название языка, на котором выражен идентифицирующий описательный атрибут, на этом языке.

*Пример 1 - "Deutsch" - наименование немецкого языка, известное людям, говорящим на этом языке.*

*Пример 2 - "Русский" - название русского языка.*

*Пример 3 - "British English" - наименование формы английского языка, для которой характерна орфография и произношение, используемые на Британских островах и в некоторых других частях света.*

#### **Comments**

Оptionальный список комментариев для используемого языка.

#### **UniqueID**

Глобально уникальный идентификатор языка.

#### **4.3.16 xtdLanguageRepresentation**

xtdLanguageRepresentation - представление языка, используемого для указания имени, описания, значения или ссылки.

EXPRESS-спецификация:

```
*)  
ENTITY xtdLanguageRepresentation  
ABSTRACT SUPERTYPE OF (ONEOF(xtdName, xtdDescription));  
LanguageName :xtdLanguage;  
UniqueID :xtdGlobalUniqueID;  
END_ENTITY;
```

(\*  
Определения атрибутов:

#### **LanguageName**

Название языка, используемого для xtdLanguageRepresentation.

#### **UniqueID**

Глобально уникальный идентификатор для представления языка.

#### **4.3.17 xtdMeasureWithUnit**

xtdMeasureWithUnit - специализация xtdObject, делающая xtdProperty измеримым путем связи единиц измерения со значением.

*Пример 1 - "1 м" - экземпляр xtdMeasureWithUnit с компонентом единицы измерения "м" и компонентом значения "1".*

*Пример 2 - "AB88" - экземпляр xtdMeasureWithUnit с компонентом значения "AB88" и без*

### **компонента единицы измерения.**

#### EXPRESS-спецификация:

```
*)  
ENTITY xtdMeasureWithUnit  
SUBTYPE OF(xtdObject);  
UnitComponent :OPTIONAL xtdUnit;  
ValueDomain :OPTIONAL LIST [1:?] OF xtdValue;  
END_ENTITY;
```

(\*  
Определения атрибутов:

#### **UnitComponent**

Единица измерения, в которой выражается значение.

#### **ValueDomain**

Область значений данной единицы измерения.

#### **4.3.18 xtdName**

xtdName - название объекта на целевом языке проекта. Объект может иметь несколько имен в одном языке, и эти имена не обязательно должны быть уникальными для определенного объекта.

***Пример 1 - "Beam" и "truss" - два английских наименования одного и того же объекта.***

***Пример 2 - "балка" - русское название английского "beam".***

#### EXPRESS-спецификация:

```
*)  
ENTITY xtdName  
SUBTYPE OF(xtdLanguageRepresentation);  
Name :xtdLabel;  
INVERSE  
is_name_of :SET [1:?] OF xtdRoot FOR Names;  
END_ENTITY;
```

(\*  
Определение атрибута:

#### **Name**

Метка, содержащая имя объекта.

#### **4.3.19 xtdNest**

xtdNest - специализация xtdCollection, позволяющая входить в коллекцию только предметам одного типа.

***Пример - "Свойства окна" - это xtdNest, члены которого - исключительно экземпляры xtdProperty.***

#### EXPRESS-спецификация:

```
*)  
ENTITY xtdNest  
SUBTYPE OF(xtdCollection);
```



```
WHERE
WR1 :SIZEOF(QUERY(Result <* SELF.RelatedThings |
NOT(TYPEOF(SELF.RelatingCollection) =
TYPEOF(Result)))) = 0 ;
END_ENTITY;
(*)
Формальное утверждение:
```

**WR1** - Тип коллекции должен всегда совпадать с типом каждого предмета, т.е. коллекция и все входящие в нее предметы должны иметь одинаковый тип.

#### 4.3.20 xtdObject

xtdObject - специализация xtdRoot, являющая собой абстрактный элемент, из которого выводятся xtdProperty, xtdMeasureWithUnit, xtdActor, xtdUnit, xtdSubject, xtdValue and xtdActivity.

EXPRESS-спецификация:

```
*)
ENTITY xtdObject
ABSTRACT SUPERTYPE OF (ONEOF(xtdSubject, xtdActivity, xtdUnit,
xtdProperty, xtdMeasureWithUnit, xtdActor, xtdValue))
SUBTYPE OF(xtdRoot);
END_ENTITY;
(*)
4.3.21 xtdProperty
```

xtdProperty - специализация xtdObject, используемая для качественного или количественного определения xtdObject.

**Пример 1 - "Ширина" - это xtdProperty.**

**Пример 2 - "Ширина двери" - это xtdProperty, используемое для качественного описания "ширины" экземпляра xtdSubject "дверь".**

**Пример 3 - "Теплопередача" - это xtdProperty.**

**Пример 4 - "Цвет" - это xtdProperty.**

**Пример 5 - "Продолжительность" - это xtdProperty.**

**Пример 6 - "Комфорт" - это xtdProperty.**

EXPRESS-спецификация:

```
*)
ENTITY xtdProperty
SUBTYPE OF(xtdObject);
END_ENTITY;
(*)
4.3.22 xtdRelActsUpon
```

xtdRelActsUpon - специализация xtdRelAssociates, представляющая один xtdObject, действующий на один или несколько других xtdObjects.

**Пример 1 - "Колонна поддерживает балку".**

**Пример 2 - "Каменщик кладет кирпичи".**

EXPRESS-спецификация:

```
*)  
ENTITY xtdRelActsUpon  
SUBTYPE OF(xtdRelAssociates);  
WHERE  
WR1 :SIZEOF(QUERY(Result <* RelatedObjects | RelatingObject:=:  
Result)) = 0;  
END_ENTITY;
```

(\*  
Формальное утверждение:

**WR1** - Экземпляр, на который указывает атрибут связующий объект, не должен содержаться в наборе атрибутов связанных объектов.

#### **4.3.23 xtdRelAssignsCollections**

xtdRelAssignsCollections - специализация xtdRelationship, осуществляющая присваивание одного или более экземпляров xtdCollection объекту xtdObject.

EXPRESS-спецификация:

```
*)  
ENTITY xtdRelAssignsCollections  
SUBTYPE OF(xtdRelationship);  
RelatingObject :xtdObject;  
RelatedCollections :SET [1:?] OF xtdCollection;  
END_ENTITY;
```

(\*  
Определения атрибутов:

##### **RelatingObject**

Объект, которому присваиваются коллекции.

##### **RelatedCollections**

Набор коллекций, которые присваиваются объекту.

#### **4.3.24 xtdRelAssignsMeasures**

xtdRelAssignsMeasures - специализация xtdRelationship, осуществляющая присваивание одного или более экземпляров xtdMeasureWithUnit свойству xtdProperty.

EXPRESS-спецификация:

```
*)  
ENTITY xtdRelAssignsMeasures SUBTYPE OF(xtdRelationship);  
RelatingProperty :xtdProperty;  
RelatedMeasures :SET [1:?] OF xtdMeasureWithUnit;  
MethodOfInterpretation :OPTIONAL xtdName;  
END_ENTITY;
```

(\*  
Определения атрибутов:

##### **RelatingProperty**

Свойство, которому присваиваются единицы измерения.

### **RelatedMeasures**

Набор единиц измерения, присваиваемых свойству.

### **MethodOfInterpretation**

Оptionальное пояснение значения в единицах измерения.

*Пример - "Верхняя граница" - это метод интерпретации меры с единицей измерения, представляющий верхнюю границу ограниченного значения.*

### **4.3.25 xtdRelAssignsProperties**

xtdRelAssignsProperties - специализация xtdRelationship, осуществляющая присваивание одного или более экземпляров xtdProperty объекту xtdObject.

*Пример - "высота двери" - это xtdProperty, присвоенное xtdObject "дверь".*

EXPRESS-спецификация:

```
*)  
ENTITY xtdRelAssignsProperties SUBTYPE OF(xtdRelationship);  
RelatedProperties :SET [1:?] OF xtdProperty;  
RelatingObject :xtdObject;  
END_ENTITY;
```

(\*  
Определения атрибутов:

### **RelatedProperties**

Набор свойств, которые присваиваются объекту.

### **RelatingObject**

Объект, которому присваиваются свойства.

### **4.3.26 xtdRelAssignsPropertyWithValues**

xtdRelAssignsPropertyWithValues - специализация xtdRelationship, являющаяся трехсторонней связью, которая осуществляет присваивание одного или нескольких значений из полного набора значений соответствующего свойства относимому объекту.

EXPRESS-спецификация:

```
*)  
ENTITY xtdRelAssignsPropertyWithValues  
SUBTYPE OF(xtdRelationship);  
RelatedProperty :xtdProperty;  
RelatingObject :xtdObject;  
RelatedValues :LIST [1:?] OF UNIQUE xtdValue;  
END_ENTITY;
```

(\*  
Определения атрибутов:

### **RelatedProperty**

Свойство, содержащее полный список значений, из которого выбираются RelatedValues.

### **RelatingObject**

Объект, которому присваивается список значений.

### **RelatedValues**

Список значений, выбранных из полного набора значений относящегося свойства.

#### **4.3.27 xtdRelAssignsUnits**

xtdRelAssignsUnits - специализация xtdRelationship, осуществляющая присваивание xtdMeasureWithUnit одного или более экземпляров xtdUnit.

EXPRESS-спецификация:

```
*)  
ENTITY xtdRelAssignsUnits  
SUBTYPE OF(xtdRelationship);  
RelatingMeasure :xtdMeasureWithUnit;  
RelatedUnits :SET [1:?] OF xtdUnit;  
END_ENTITY;
```

(\*  
Определения атрибутов:

### **RelatingMeasure**

Мера, которой присваиваются единицы измерения.

### **RelatedUnits**

Список единиц измерения, которые присваиваются мере.

#### **4.3.28 xtdRelAssignsValues**

xtdRelAssignsValues - специализация xtdRelationship, осуществляющая присваивание xtdMeasureWithUnit одного или более экземпляров xtdValue.

EXPRESS-спецификация:

```
*)  
ENTITY xtdRelAssignsValues SUBTYPE OF(xtdRelationship);  
RelatingMeasure :xtdMeasureWithUnit;  
RelatedValues :LIST [1:?] OF UNIQUE xtdValue;  
END_ENTITY;
```

(\*  
Определения атрибутов:

### **RelatingMeasure**

Мера, которой присваиваются значения.

### **RelatedValues**

Список единиц измерения, которые присваиваются мере.

#### **4.3.29 xtdRelAssociates**

xtdRelAssociates - специализация xtdRelationship, осуществляющая сопоставление относимого xtdObject и набора относящихся экземпляров xtdObject.

EXPRESS-спецификация:

```
*)
ENTITY xtdRelAssociates
SUPERTYPE OF (ONEOF(xtdRelComposes, xtdRelGroups, xtdRelSpecializes,
xtdRelActsUpon))
SUBTYPE OF(xtdRelationship);
RelatingObject :xtdObject;
RelatedObjects :SET [1:?] OF xtdObject;
WHERE
WR1 :SIZEOF(QUERY(Result <* RelatedObjects | RelatingObject:=:
Result)) = 0;
END_ENTITY;
```

(\*  
Определения атрибутов:

**RelatingObject**

Объект, являющийся целевым для отношения.

**RelatedObjects**

Набор объектов, которые сопоставляются с целевым объектом.

Формальные утверждения:

**WR1** - Экземпляр, на который указывает атрибут связующий объект, не должен содержаться в наборе атрибута связанных объектов.

**4.3.30 xtdRelCollects**

xtdRelCollects - специализация xtdRelationship, представляющая собрание произвольных сущностей в xtdCollection. xtdCollection инстанцируется как xtdNest или xtdBag.

EXPRESS-спецификация:

```
*)
ENTITY xtdRelCollects
SUBTYPE OF(xtdRelationship);
RelatedThings :SET [1:?] OF xtdRoot;
RelatingCollection :xtdCollection;
END_ENTITY;
```

(\*  
Определения атрибутов:

**RelatedThings**

Набор предметов, составляющих коллекцию.

**RelatingCollection**

Коллекция предметов, где каждый экземпляр коллекции представляет собой объект или отношение.

**4.3.31 xtdRelComposes**

xtdRelComposes - специализация xtdRelAssociates, представляющая xtdObject составленным из других экземпляров xtdObject. Отношение композиции может применяться рекурсивно, так что

составной xtdObject может быть частью другого составного xtdObject.

**Пример - "Дверная рама" и "створка двери" - это части (целого) "дверного блока".**

EXPRESS-спецификация:

```
*)
ENTITY xtdRelComposes
SUBTYPE OF(xtdRelAssociates);
WHERE
WR1 :SIZEOF(QUERY(Result <* SELF.RelatedObjects |
NOT(TYPEOF(SELF.RelatingObject) = TYPEOF(Result)))) = 0 ;
WR2 :SIZEOF(QUERY(Result2 <* RelatedObjects | RelatingObject:=
Result2)) = 0;
END_ENTITY;
```

(\*  
Формальные утверждения:

**WR1** - Тип связанного объекта должен всегда совпадать с типом каждого связанного объекта, т.е. экземпляры xtdSubject состояются только из других экземпляров xtdSubject.

**WR2** - Экземпляр, на который указывает атрибут связующий объект, не должен содержаться в наборе атрибута связанных объектов.

#### **4.3.32 xtdRelDocuments**

xtdRelDocuments - специализация xtdRelationship, документирующая объекты путем их отношения к документам, в которых они описываются. Каждый документ может относиться ко многим объектам.

EXPRESS-спецификация:

```
*)
ENTITY xtdRelDocuments
SUBTYPE OF(xtdRelationship);
RelatedObjects :SET [1:?] OF xtdObject;
RelatingDocument :xtdExternalDocument;
END_ENTITY;
```

(\*  
Определения атрибутов:

##### **RelatedObjects**

Набор объектов, которые описываются во внешнем документе.

##### **RelatingDocument**

Внешний документ, который описывает связанные объекты.

#### **4.3.33 xtdRelGroups**

xtdRelGroups - специализация xtdRelAssociates, осуществляющая группирование наборов относящихся экземпляров xtdObject в относимом xtdObject.

EXPRESS-спецификация:

```
*)
ENTITY xtdRelGroups
SUBTYPE OF(xtdRelAssociates);
```

```
WHERE
WR1 :SIZEOF(QUERY(Result <* RelatedObjects | RelatingObject:=:
Result)) = 0;
END_ENTITY;
(*
```

Формальные утверждения:

**WR1** - Экземпляр, на который указывает атрибут связующий объект, не должен содержаться в списке атрибута связанных объектов.

#### 4.3.34 xtdRelSequences

xtdRelSequences - специализация xtdRelationship, описывающая последовательность двух экземпляров xtdActivity. Связывающая xtdActivity предшествует в последовательности связуемой xtdActivity.

EXPRESS-спецификация:

```
*)
ENTITY xtdRelSequences
SUBTYPE OF(xtdRelationship);
RelatingActivity :OPTIONAL xtdActivity;
RelatedActivity :OPTIONAL xtdActivity;
WHERE
WR1 :RelatedActivity:<>: RelatingActivity;
WR2 :EXISTS(RelatingActivity) OR EXISTS(RelatedActivity);
END_ENTITY;
(*
```

Определения атрибутов:

#### **RelatingActivity**

Ссылка на предшествующее действие в последовательности.

#### **RelatedActivity**

Ссылка на последующее действие в последовательности.

Формальные утверждения:

**WR1** - Относимое действие не должно указывать на тот же экземпляр, что и относящееся действие.

**WR2** - Относимое либо относящееся действие должно быть подтверждено.

#### 4.3.35 xtdRelSpecializes

xtdRelSpecializes - специализация xtdRelAssociates, представляющая специализацию супертипа в один или более подтипов, при условии, что подтип более ограничен, чем супертип.

**Пример 1 - "Входная дверь" - это специализация "двери".**

**Пример 2 - "Пространство" - это обобщение "комнаты".**

EXPRESS-спецификация:

```
*)
```

```

ENTITY xtdRelSpecializes
SUBTYPE OF(xtdRelAssociates);
WHERE
WR1 :SIZEOF(QUERY(Result <* RelatedObjects | RelatingObject:=:
Result)) = 0;
WR2 :SIZEOF(QUERY(Result <* SELF.RelatedObjects |
NOT(TYPEOF(SELF.RelatingObject) = TYPEOF(Result)))) = 0 ;
END_ENTITY;
(*)
Формальные утверждения:

```

**WR1** - Экземпляр, на который указывает атрибут связующий объект, не должен содержаться в списке атрибута связанных объектов.

**WR2** - Тип связующего объекта всегда должен совпадать с типом каждого связуемого объекта, т.е. экземпляр xtdSubject может быть только специализацией другого экземпляра xtdSubject.

#### 4.3.36 xtdRelationship

xtdRelationship - специализация xtdRoot. Это абстрактный элемент, который объективирует отношения, позволяющий добавлять к таким сущностям дополнительные атрибуты.

EXPRESS-спецификация:

```

*)
ENTITY xtdRelationship
ABSTRACT SUPERTYPE OF (ONEOF(xtdRelAssociates, xtdRelAssignsProperties,
xtdRelAssignsMeasures, xtdRelCollects,
xtdRelAssignsCollections, xtdRelSequences,
xtdRelDocuments, xtdRelAssignsUnits,
xtdRelAssignsValues, xtdRelAssignsPropertyWithValues))
SUBTYPE OF(xtdRoot);
ViewSelector :OPTIONAL xtdName;
END_ENTITY;
(*)
Определение атрибута:

```

#### ViewSelector

Необязательный атрибут, используемый для уточнения отношения в целях создания различных представлений данных.

**Пример** - Переключатель представлений "IFC 2x2" может использоваться для идентификации отношений, исходящих из модели IFC 2x2.

#### 4.3.37 xtdRoot

xtdRoot - абстрактная сущность, с помощью которой основные сервисы идентификации, присвоения имени, версии, описания и ссылок могут применяться к экземплярам ее подтипов xtdObject, xtdRelationship и xtdCollection.

EXPRESS-спецификация:

```

*)
ENTITY xtdRoot
ABSTRACT SUPERTYPE OF (ONEOF(xtdObject, xtdRelationship,
xtdCollection));
VersionDate :OPTIONAL xtdDate;
VersionID :OPTIONAL xtdVersionID;

```



UniqueID :xtdGlobalUniqueID;  
Descriptions :OPTIONAL SET [1:?] OF xtdDescription;  
Names :SET [1:?] OF xtdName;  
END\_ENTITY;

(\*  
Определения атрибутов:

#### **VersionDate**

Дата последнего исправления объекта справочных данных.

#### **VersionID**

Номер версии справочных данных версии объекта.

#### **UniqueID**

Глобально уникальный идентификатор объекта.

#### **Descriptions**

Оptionальный набор описаний объекта не зависящий от языка.

#### **Names**

Набор названий объекта зависящих от языка.

#### **4.3.38 xtdSubject**

xtdSubject - специализация xtdObject, являющаяся субъектом, который описывается сопоставленными свойствами и действиями. xtdSubject может быть физическим или логическим.

*Пример 1 - "Крыша" - это xtdSubject.*

*Пример 2 - "Вестибюль" - это xtdSubject.*

*Пример 3 - "Система контроля" - это xtdSubject.*

*Пример 4 - "Дорога" - это xtdSubject.*

*Пример 5 - "Аэропорт" - это xtdSubject.*

*Пример 6 - "Программное обеспечение" - это xtdSubject.*

EXPRESS-спецификация:

\*)  
ENTITY xtdSubject  
SUBTYPE OF(xtdObject);  
END\_ENTITY;

#### **4.3.39 xtdUnit**

xtdUnit - специализация xtdObject, представляющая собой шкалу, позволяющую измерить значение.

*Пример 1 - "метр" - xtdUnit, обозначаемый символом "м".*

*Пример 2 - "Вт/м·К" - xtdUnit, используемый для свойства "теплопроводность".*

EXPRESS-спецификация:

```
*)  
ENTITY xtdUnit  
SUBTYPE OF(xtdObject);  
END_ENTITY;
```

(  
**4.3.40 xtdValue**

xtdValue - специализация xtdLanguageRepresentation, содержащая описание значения xtdProperty.

**Пример 1 - "Vertikaal schuivend" - голландское обозначение "вертикального скольжения".**

EXPRESS-спецификация:

```
*)  
ENTITY xtdValue  
SUBTYPE OF(xtdObject);  
LowerTolerance :OPTIONAL xtdText;  
NominalValue :OPTIONAL xtdText;  
UpperTolerance :OPTIONAL xtdText;  
ValueType :OPTIONAL xtdValueTypeEnum;  
ValueRole :OPTIONAL xtdValueRoleEnum;  
ToleranceType :OPTIONAL xtdToleranceTypeEnum;  
END_ENTITY;
```

(  
Определения атрибутов:

**LowerTolerance**

Опциональный нижний допуск значения.

**NominalValue**

Опциональная текстовая строка, содержащая номинальное значение.

**UpperTolerance**

Опциональный верхний допуск значения.

**ValueType**

Опционально выбираемый тип для указания типа значения.

**ValueRole**

Опционально выбираемый тип для указания роли значения.

**ToleranceType**

Опционально выбираемый тип для указания типа допуска для значения.

```
*)  
END_SCHEMA;
```

**4.4 Подробная спецификация EXPRESS<sup>1)</sup>**

---

1) В настоящем стандарте приводится подробная спецификация EXPRESS, представленная в стандарте ИСО 12006-3.

Данная формальная спецификация представлена на языке EXPRESS.

```
SCHEMA ISO_12006_3_VERSION_3;
TYPE xtdDate = STRING;
END_TYPE;
TYPE xtdGlobalUniqueID = STRING;
END_TYPE;
TYPE xtdLabel = STRING;
END_TYPE;
TYPE xtdText = STRING;
END_TYPE;
TYPE xtdToleranceTypeEnum = ENUMERATION OF
(REALVALUE,
PERCENTAGE);
END_TYPE;
TYPE xtdValueRoleEnum = ENUMERATION OF
(NOMINAL,
MAXIMUM,
MINIMUM);
END_TYPE;
TYPE xtdValueTypeEnum = ENUMERATION OF
(XTDSTRING,
XTDNUMBER,
XTDINTEGER,
XTDREAL,
XTDBOOLEAN,
XTDLOGICAL);
END_TYPE;
TYPE xtdVersionID = STRING;
END_TYPE;
ENTITY xtdActivity
SUBTYPE OF(xtdObject);
END_ENTITY;
ENTITY xtdActor
SUBTYPE OF(xtdObject);
END_ENTITY;
ENTITY xtdBag
SUBTYPE OF(xtdCollection);
END_ENTITY;
ENTITY xtdCollection
ABSTRACT SUPERTYPE OF (ONEOF(xtdNest, xtdBag))
SUBTYPE OF(xtdRoot);
END_ENTITY;
ENTITY xtdDescription
SUBTYPE OF(xtdLanguageRepresentation);
Description : xtdText;
INVERSE
is_description_of : SET [1:?] OF xtdRoot FOR Descriptions;
END_ENTITY;
ENTITY xtdExternalDocument;
UniqueID : xtdGlobalUniqueID;
Names : SET [1:?] OF xtdName;
END_ENTITY;
ENTITY xtdLanguage;
```

```

LanguageNameInEnglish : xtdLabel;
LanguageNameInSelf : OPTIONAL xtdLabel;
Comments : OPTIONAL LIST [1:?] OF xtdText;
UniqueID : xtdGlobalUniqueID;
END_ENTITY;
ENTITY xtdLanguageRepresentation
ABSTRACT SUPERTYPE OF (ONEOF(xtdName, xtdDescription));
LanguageName : xtdLanguage;
UniqueID : xtdGlobalUniqueID;
END_ENTITY;
ENTITY xtdMeasureWithUnit
SUBTYPE OF(xtdObject);
UnitComponent : OPTIONAL xtdUnit;
ValueDomain : OPTIONAL LIST [1:?] OF xtdValue;
END_ENTITY;
ENTITY xtdName
SUBTYPE OF(xtdLanguageRepresentation);
Name : xtdLabel;
INVERSE
is_name_of : SET [1:?] OF xtdRoot FOR Names;
END_ENTITY;
ENTITY xtdNest
SUBTYPE OF(xtdCollection);
WHERE
WR1 : SIZEOF(QUERY(Result <* SELF.RelatedThings |
NOT(TYPEOF(SELF.RelatingCollection) = TYPEOF(Result)))) = 0 ;
END_ENTITY;
ENTITY xtdObject
ABSTRACT SUPERTYPE OF (ONEOF(xtdSubject, xtdActivity, xtdUnit, xtdProperty,
xtdMeasureWithUnit, xtdActor, xtdValue))
SUBTYPE OF(xtdRoot);
END_ENTITY;
ENTITY xtdProperty
SUBTYPE OF(xtdObject);
END_ENTITY;
ENTITY xtdRelActsUpon
SUBTYPE OF(xtdRelAssociates);
WHERE
WR1 : SIZEOF(QUERY(Result <* RelatedObjects | RelatingObject:= Result))= 0;
END_ENTITY;
ENTITY xtdRelAssignsCollections
SUBTYPE OF(xtdRelationship) ;
RelatingObject : xtdObject;
RelatedCollections : SET [1:?] OF xtdCollection;
END_ENTITY;
ENTITY xtdRelAssignsMeasures
SUBTYPE OF(xtdRelationship);
RelatingProperty : xtdProperty;
RelatedMeasures : SET [1:?] OF xtdMeasureWithUnit;
MethodOfInterpretation : OPTIONAL xtdName;
END_ENTITY;
ENTITY xtdRelAssignsProperties
SUBTYPE OF(xtdRelationship) ;
RelatedProperties : SET [1:?] OF xtdProperty;
RelatingObject : xtdObject;
END_ENTITY;
ENTITY xtdRelAssignsPropertyWithValues
SUBTYPE OF(xtdRelationship);
RelatedProperty : xtdProperty;

```

```

RelatingObject : xtdObject;
RelatedValues : LIST [1:?] OF UNIQUE xtdValue;
END_ENTITY;
ENTITY xtdRelAssignsUnits
SUBTYPE OF(xtdRelationship);
RelatingMeasure : xtdMeasureWithUnit;
RelatedUnits : SET [1:?] OF xtdUnit;
END_ENTITY;
ENTITY xtdRelAssignsValues
SUBTYPE OF(xtdRelationship);
RelatingMeasure : xtdMeasureWithUnit;
RelatedValues : LIST [1:?] OF UNIQUE xtdValue;
END_ENTITY;
ENTITY xtdRelAssociates
SUPERTYPE OF (ONEOF(xtdRelComposes, xtdRelGroups, xtdRelSpecializes, xtdRelActsUpon))
SUBTYPE OF(xtdRelationship);
RelatingObject : xtdObject;
RelatedObjects : SET [1:?] OF xtdObject;
WHERE
WR1 : SIZEOF(QUERY(Result <* RelatedObjects | RelatingObject:=: Result))= 0;
END_ENTITY;
ENTITY xtdRelationship
ABSTRACT SUPERTYPE OF (ONEOF(xtdRelAssociates, xtdRelAssignsProperties,
xtdRelAssignsMeasures, xtdRelCollects, xtdRelAssignsCollections,
xtdRelSequences, xtdRelDocuments, xtdRelAssignsUnits, xtdRelAssignsValues,
xtdRelAssignsPropertyWithValues))
SUBTYPE OF(xtdRoot);
ViewSelector : OPTIONAL xtdName;
END_ENTITY;
ENTITY xtdRelCollects
SUBTYPE OF(xtdRelationship);
RelatedThings : SET [1:?] OF xtdRoot;
RelatingCollection : xtdCollection;
END_ENTITY;
ENTITY xtdRelComposes
SUBTYPE OF(xtdRelAssociates);
WHERE
WR1 : SIZEOF(QUERY(Result <* SELF.RelatedObjects |
NOT(TYPEOF(SELF.RelatingObject) = TYPEOF(Result)))) = 0 ;
WR2 : SIZEOF(QUERY(Result2 <* RelatedObjects | RelatingObject:=: Result2)) = 0;
END_ENTITY;
ENTITY xtdRelDocuments
SUBTYPE OF(xtdRelationship);
RelatedObjects : SET [1:?] OF xtdObject;
RelatingDocument : xtdExternalDocument;
END_ENTITY;
ENTITY xtdRelGroups
SUBTYPE OF(xtdRelAssociates);
WHERE
WR1 : SIZEOF(QUERY(Result <* RelatedObjects | RelatingObject:=: Result))= 0;
END_ENTITY;
ENTITY xtdRelSequences
SUBTYPE OF(xtdRelationship);
RelatingActivity : OPTIONAL xtdActivity;
RelatedActivity : OPTIONAL xtdActivity;
WHERE
WR1 : RelatedActivity:<>: RelatingActivity;
WR2 : EXISTS(RelatingActivity) OR EXISTS(RelatedActivity);
END_ENTITY;

```

```

ENTITY xtdRelSpecializes
SUBTYPE OF(xtdRelAssociates);
WHERE
WR1 : SIZEOF(QUERY(Result <* RelatedObjects | RelatingObject:=: Result))= 0;
WR2 : SIZEOF(QUERY(Result <* SELF.RelatedObjects |
NOT(TYPEOF(SELF.RelatingObject) = TYPEOF(Result)))) = 0 ;
END_ENTITY;
ENTITY xtdRoot
ABSTRACT SUPERTYPE OF (ONEOF(xtdObject, xtdRelationship, xtdCollection));
VersionDate : OPTIONAL xtdDate;
VersionID : OPTIONAL xtdVersionID;
UniqueID : xtdGlobalUniqueID;
Descriptions : OPTIONAL SET [1:?] OF xtdDescription;
Names : SET [1:?] OF xtdName;
END_ENTITY;
ENTITY xtdSubject
SUBTYPE OF(xtdObject);
END_ENTITY;
ENTITY xtdUnit
SUBTYPE OF(xtdObject);
END_ENTITY;
ENTITY xtdValue
SUBTYPE OF(xtdObject);
LowerTolerance: OPTIONAL xtdText;
NominalValue : OPTIONAL xtdText;
UpperTolerance: OPTIONAL xtdText;
ValueType : OPTIONAL xtdValueTypeEnum;
ValueRole : OPTIONAL xtdValueRoleEnum;
ToleranceType : OPTIONAL xtdToleranceTypeEnum;
END_ENTITY;
END_SCHEMA;

```

Приложение А  
(справочное)

## Соглашение об именах

### А.1 Общие положения

При разработке настоящего стандарта использовались нижеприведенные соглашения об именах. Эти соглашения не применимы при создании содержимого информационной структуры или при ее использовании.

### А.2 Символы, используемые для имен

Для имен используются прописные буквы [A-Z], строчные буквы [a-z] и цифры [0-9].

### А.3 Регистр имен

Имена записываются прописными и строчными буквами как единое имя без пробелов.

Первый символ в каждом слове, как правило, следует после префикса схемы и записывается прописной буквой. Все другие символы, формирующие это же слово, как правило, записываются строчными буквами.

### А.4 Длина имен

Длина имен не ограничена.

### А.5 Префикс xtd

Для идентификации имен, используемых в информационной модели настоящего стандарта, используется префикс "xtd". "xtd" - это аббревиатура для extensible taxonomy definition ("расширяемое классификационное определение").

#### **А.6 Имена элементов (за исключением отношений)**

Именем элемента может служить существительное или комбинация существительных, обозначающих содержание или тип этого объекта.

#### **А.7 Именованное отношение**

Сущности, действующие в модели в качестве отношений, содержат обозначение "Rel" после префикса "xtd" и перед самим именем сущности. Как правило, вставка "Rel" считается словом, к ней применяется правило "Регистр имени". Отношения именуются начиная с связуемых (относимых) объектов (SET [1:?]) и заканчивая связующим (относимым) объектом; например, xtdRelComposes означает, что отношение объединяет части (SET [1:?]) в единое целое.

Именованное отношение - это глагол или глагольная фраза, обозначающие "функцию" отношения.

#### **А.8 Множественное число в имени агрегации**

Имена атрибутов и отношений внутри агрегации выражаются в форме множественного числа.

### Приложение ДА (справочное)

#### **Сведения о соответствии ссылочных международных стандартов национальным стандартам**

Таблица ДА.1

Обозначение ссылочного международного стандарта	Степень соответствия	Обозначение и наименование соответствующего национального стандарта
ISO 10303-11	IDT	<a href="#">ГОСТ Р ИСО 10303-11-2009</a> "Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 11. Методы описания. Справочное руководство по языку EXPRESS"
ISO/IEC 10646	-	*
* Перевод разделов 9, 10 международного стандарта ISO/IEC 10646:2017 приведен в приложении ДБ.		
Примечание - В настоящей таблице использовано следующее условное обозначение степени соответствия стандарта: - IDT - идентичный стандарт.		

### Приложение ДБ (обязательное)

#### **Перевод разделов 9, 10 международного стандарта ISO/IEC 10646:2017**

#### **9 Формы кодирования UCS**

##### **9.1 Общие положения**

Настоящий международный стандарт предоставляет три формы кодирования, выражающие каждое UCS скалярное значение в виде одного уникального кодового набора или последовательности нескольких уникальных кодовых наборов. Они называются UTF-8, UTF-16 и UTF-32 соответственно (от англ. Unicode Transformation Format, 8-bit, 16-bit, 32-bit - формат преобразования Юникода, 8-бит, 16-

бит, 32-бит соответственно).

## 9.2 UTF-8

UTF-8 является формой кодирования UCS, которая присваивает каждому скалярному значению UCS байтовую последовательность из одного-четырех байтов, как определено в таблице 2.

- Символы UCS из "BASIC LATIN" (базового набора латиницы) представлены в UTF-8 в соответствии с ISO/IEC 4873, т.е. в виде отдельных байтов со значениями от 20 до 7E.

- Функции управления в кодовых точках от 0000 до 001F и управляющий символ в кодовой точке 007F представлены без использования набора байтов, установленных в разделе 11 настоящего стандарта, т.е. в виде одиночных байтов со значениями от 00 до 1F и 7F соответственно, в соответствии с ISO/IEC 4873 и с 8-битной структурой ISO/IEC 2022.

- Байт значений от 00 до 7F создается исключительно в кодировке UTF-8 для каждого символа. Это обеспечивает совместимость с существующими системами обработки файлов и подсистемами связи, которые анализируют последовательности единиц кода для байта этих значений.

- Первый байт в кодированном представлении UTF-8 любого символа может быть непосредственно идентифицирован при проверке последовательности единиц кода, по очереди по одному байту, начиная с произвольного местоположения. Он указывает количество байтов (если таковые имеются) в последовательности из нескольких байтов, составляющей кодовое представление этого символа.

Таблица 2 задает битовое распределение для формы кодирования UTF-8, показывая диапазоны скалярных значений UCS, соответствующих одной, двум, трем и четырем последовательностям байтов.

Таблица 2 - Распределение битов UTF-8

Скалярное значение	1-й байт	2-й байт	3-й байт	4-й байт
00000000xxxxxxx	0xxxxxxx			
0000yyyyxxxxxx	110yyyyy	10xxxxxx		
zzzyyyyyxxxxxx	1110zzzz	10yyyyyy	10xxxxxx	
000uuuuuzzzyyyyyxxxxxx	11110uuu	10uuzzzz	10yyyyyy	10xxxxxx

Поскольку суррогатные кодовые точки не являются скалярными значениями UCS, любая последовательность UTF-8, которая иначе сопоставлялась бы с кодовыми точками D800-DFFF, сформирована некорректно.

В таблице 3 перечислены все диапазоны (включительно) байтовых последовательностей, которые правильно сформированы в UTF-8. Любая последовательность UTF-8, которая не соответствует шаблону, перечисленным в таблице 3, является некорректной.

Таблица 3 - Правильно сформированные последовательности байтов UTF-8

Кодовые точки	1-й байт	2-й байт	3-й байт	4-й байт
0000-007F	00-7F			
0080-07FF	C2-DF	80-BF		
0800-0FFF	E0	A0-BF	80-BF	
1000-CFFF	E1-EC	80-BF	80-BF	
D000-D7FF	ED	80-9F	80-BF	
E000-FFFF	EE-EF	80-BF	80-BF	
10000-3FFFF	F0	90-BF	80-BF	80-BF
40000-FFFFFF	F1-F3	80-BF	80-BF	80-BF
100000-10FFFF	F4	80-8F	80-BF	80-BF

Для соблюдения условий корректности, указанных в таблице 9.2, в UTF-8 запрещены следующие





Настоящий международный стандарт определяет семь схем кодирования: UTF-8, UTF-16BE, UTF-16LE, UTF-16, UTF-32BE, UTF-32LE и UTF-32.

## 10.2 UTF-8

Схема кодирования UTF-8 сериализует последовательность единиц кода UTF-8 в том же порядке, в каком чередуются наборы кодов.

При кодировании в UTF-8 подпись превращается в последовательность байт <EF BB BF>. Их использование в начале потока данных UTF-8 не является ни обязательным, ни рекомендуемым, но не влияет на соответствие.

## 10.3 UTF-16BE

Схема кодирования UTF-16BE сериализует кодовую последовательность UTF-16, упорядочивая байты таким образом, что наиболее значимый байт (старший) предшествует менее значимому байту (младшему). Порядок от старшего к младшему.

Примечание - Этот порядок известен, как порядок **big-endian** от англ. - большой конец. Этот порядок соответствует привычному порядку записи арабских цифр, например, число сто двадцать три было бы записано при таком порядке как 123.

В UTF-16BE начальная последовательность байтов <FE FF> интерпретируется как "FEFF ZERO WIDTH NO-BREAK SPACE" (неразрывный пробел нулевой ширины) и не передает значение подписи.

## 10.4 UTF-16LE

Схема кодирования UTF-16LE сериализует последовательность единиц кода UTF-16, упорядочивая байты таким образом, что младший байт предшествует старшему байту (порядок от младшего к старшему).

В UTF-16LE начальная последовательность байтов <FE FF> интерпретируется как "FEFF ZERO WIDTH NO-BREAK SPACE" (неразрывный пробел нулевой ширины) и не передает значение подписи.

## 10.5 UTF-16

Схема кодирования UTF-16 сериализует последовательность единиц кода UTF-16, упорядочивая байты таким образом, что младший байт предшествует или следует за старшим байтом.

В схеме кодирования UTF-16 начальная подпись, читаемая как <FE FF>, указывает, что старший байт предшествует младшему байту, а <FF FE> - наоборот. Подпись не является частью текстовых данных.

При отсутствии подписи порядок байт схемы кодирования UTF-16 состоит в том, что старший байт предшествует младшему байту.

## 10.6 UTF-32BE

Схема кодирования UTF-32BE сериализует последовательность единиц кода UTF-32, упорядочивая байты таким образом, что старшие байты предшествуют младшим байтам (порядок от старшего к младшему).

В UTF-32BE начальная последовательность байт <00 00 FE FF> интерпретируется как FEFF ZERO WIDTH NO-BREAK SPACE (неразрывный пробел нулевой ширины) и не передает значение подписи.

## 10.7 UTF-32LE

Схема кодирования UTF-32LE сериализует последовательность единиц кода UTF-32,

упорядочивая байты таким образом, что младший байт предшествует старшему байту (порядок от младшего к старшему).

В UTF-32LE начальная последовательность байт <FF FE 00 00> интерпретируется как FEFF ZERO WIDTH NO-BREAK SPACE (неразрывный пробел нулевой ширины) и не передает значение подписи.

## 10.8 UTF-32

Схема кодирования UTF-32 сериализует последовательность единиц кода UTF-32, упорядочивая байты таким образом, что младшие байты предшествуют или следуют за старшими байтами.

В схеме кодирования UTF-32 начальная подпись <00 00 FE FF> указывает, что старший байт предшествует младшему байту, а <FF FE 00 00> - наоборот. Подпись не является частью текстовых данных.

При отсутствии подписи порядок байт схемы кодирования UTF-32 состоит в том, что старший байт предшествует младшему байту.

## Библиография

- [1] ISO 3166-1, Country Codes
- [2] ISO 10303-41, Industrial automation systems and integration - Product data representation and exchange - Part 41: Integrated generic resource: Fundamentals of product description and support
- [3] ISO 10303-221, Industrial automation systems and integration - Product data representation and exchange - Part 221: Application protocol: Functional data and their schematic representation for process plants
- [4] ISO/IEC 10646, Information technology - Universal Multiple-Octet Coded Character Set (UCS), Annex D (Technically equivalent to the definitions in the Unicode Standard)
- [5] ISO 12006-2, Building Construction - Organization of information about construction works - Part 2: Framework for classification of information
- [6] ISO 15926-2, Industrial automation systems and integration - Integration of life-cycle data for process plants including oil and gas production facilities - Part 2: Data model
- [7] ISO 16739-1, Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries - Part 1: Data schema
- [8] The Unicode Consortium. The Unicode Standard, Version 4.0. Boston, MA, Addison-Wesley, 2003, ISBN 0-321-18578-1

---

УДК 004.9:006.354

ОКС 91.010.01  
35.240.67  
35.240.01

Ключевые слова: система стандартов, информационное моделирование, здания и сооружения, строительство зданий, основы обмена, объектно-ориентированная информация

---