
**Управление языковыми ресурсами.
Структуры элементов.**

Часть 2.

Декларация системы элементов

Language resource management. – Feature structures –

Part 2:

Feature system declaration

iteh Standards
(<https://standards.iteh.ai>)
Document Preview

ISO 24610-2:2011

<https://standards.iteh.ai/catalog/standards/iso/87dd2075-4d96-4e49-bcb9-43361d03fc99/iso-24610-2-2011>

Ответственность за подготовку русской версии несёт GOST R
(Российская Федерация) в соответствии со статьёй 18.1 Устава ISO



Ссылочный номер
ISO 24610-2:2011(R)

iTeh Standards
(<https://standards.iteh.ai>)
Document Preview

ISO 24610-2:2011

<https://standards.iteh.ai/catalog/standards/iso/87dd2075-4d96-4e49-bcb9-43361d03fc99/iso-24610-2-2011>



ДОКУМЕНТ ЗАЩИЩЁН АВТОРСКИМ ПРАВОМ

© ISO 2011

Все права сохраняются. Если не указано иное, никакую часть настоящей публикации нельзя копировать или использовать в какой-либо форме или каким-либо электронным или механическим способом, включая фотокопии и микрофильмы, без предварительного письменного согласия издателя.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Опубликовано в Швейцарии

Содержание

Страница

Предисловие.....	iv
Введение	v
1 Область применения.....	1
2 Нормативные ссылки	1
3 Термины и определения.....	2
4 Общая структура стандарта	6
5 Базовые понятия	7
5.1 Рассматриваемые типизированные структуры элементов	7
5.2 Типы	8
5.3 Иерархии наследования типов.....	11
5.4 Ограничения для типов	12
5.5 Опциональные (стандартные) значения и недоопределение.....	13
5.6 Категоризация	14
6 Определение формальной правильности и адекватности	16
6.1 Общее описание	16
6.2 О стандарте ISO 24610.....	17
7 Система элементов для грамматики.....	22
7.1 Общие сведения	22
7.2 Выборочные FSD.....	23
8 Декларация системы элементов	27
8.1 Общие сведения	27
8.2 Привязка текста к декларациям систем элементов.....	28
8.3 Общая структура декларации системы элементов.....	29
8.4 Декларации элементов	31
8.5 Ограничения структуры элементов	37
Приложение А (нормативное) Схема XML для структур элементов	40
Приложение В (информативное) Детализированный пример.....	50
Библиография	54

Предисловие

Международная организация по стандартизации (ISO) является всемирной федерацией национальных организаций по стандартизации (комитетов-членов ISO). Разработка международных стандартов обычно осуществляется техническими комитетами ISO. Каждый комитет-член, заинтересованный в деятельности, для которой был создан технический комитет, имеет право быть представленным в этом комитете. Международные правительственные и неправительственные организации, имеющие связь с ISO, также принимают участие в работе. ISO работает в тесном сотрудничестве с Международной электротехнической комиссией (IEC) по всем вопросам стандартизации в области электротехники.

Проекты международных стандартов разрабатываются согласно правилам, приведённым в Директивах ISO/IEC, Часть 2.

Разработка международных стандартов является основной задачей технических комитетов. Проекты международных стандартов, принятые техническими комитетами, рассылаются комитетам-членам на голосование. Для публикации в качестве международного стандарта требуется одобрение не менее 75 % комитетов-членов, принявших участие в голосовании.

Принимается во внимание тот факт, что некоторые из элементов настоящей части стандарта ISO 9735 могут быть объектом патентных прав. ISO не принимает на себя обязательств по определению отдельных или всех таких патентных прав.

ISO 24610-2 был подготовлен Техническим комитетом ISO/TC 37, *Терминология и другие языковые и информационные ресурсы*, Подкомитет SC 4, *Управление языковыми ресурсами*.

В целом серия ISO 24610 состоит из следующих частей, объединённых общим названием *Управление языковыми ресурсами. Структуры элементов*:

— *Часть 1. Представление структуры элементов*

— *Часть 2. Декларация системы элементов*

Введение

ISO 24610 состоит из двух отдельных важных частей.

- Часть 1, *Представление структуры элементов*, посвящена описанию структур, обеспечивающих неформальное, но достаточно явное выражение их характеристик, а также описанию представления структур элементов с использованием языка XML вообще и различных типов таких структур, в частности. В этой части закладываются основы правильного форматирования конструируемых XML-ссылок, обеспечивающих обмен структурами элементов (возможно, с выделением типов) между приложениями.
- Часть 2, *Декларация системы элементов*, предоставляет стандартный метод реализации различных типов структур элементов в языковой среде XML: сначала путём определения множества типов и их иерархии; затем посредством формулирования ограничений, касающихся различных типов, на множестве элементов и их допустимых значений, и, наконец, путём введения множества условий, касающихся надёжности структур элементов в аспекте их использования в конкретных приложениях, - особенно, в целях управления языковыми ресурсами.

Структура элементов – это структура данных общего назначения, которая идентифицирует и группирует отдельные элементы посредством присваивания каждому из них конкретного значения. Благодаря универсальности структур элементов они могут использоваться для представления самых разных типов информации. Существующие связи между различными «порциями» информации и их реализация в языке разметки образуют некоторый метаязык для представления контента лингвистического характера. Более того, подобная реализация позволяет сформировать описание множества элементов и значений, соответствующих конкретным типам и их ограничениям, посредством декларирования системы элементов или с помощью других механизмов языка XML, обсуждаемых в данной части ISO 24610.

Некоторые положения данной части заимствованы из ISO 24610-1:2006 в целях обеспечения полной независимости части 2 от части 1.

[ISO 24610-2:2011](https://standards.iteh.ai/catalog/standards/iso/87dd2075-4d96-4e49-bcb9-43361d03fc99/iso-24610-2-2011)

<https://standards.iteh.ai/catalog/standards/iso/87dd2075-4d96-4e49-bcb9-43361d03fc99/iso-24610-2-2011>

Управление языковыми ресурсами. Структуры элементов.

Часть 2.

Декларация системы элементов

1 Область применения

В данной части ISO 24610 предлагается формат представления, хранения и обмена для структур элементов в прикладных системах, основанных на использовании естественного языка, как для аннотирования, так и для формирования лингвистических данных. Основная цель состоит в том, чтобы предложить такой формат машинной обработки, который позволяет определить иерархию типов и декларировать ограничения, накладываемые на множество спецификаций элементов и на операции со структурами элементов, обеспечивая таким образом средства контроля соответствия каждой структуры элементов их базовой спецификации. Структуры элементов – это важнейшая часть многих формализаций в лингвистике и основополагающий механизм представления информации, используемой или порождаемой в приложениях, связанных с построением языковых систем.

Декларация системы элементов (FSD - feature system declaration) представляет собой вспомогательный файл, относящийся к тексту конкретного типа, в рамках которого используются структурированные элементы. Такая декларация служит четырём основным целям:

- Обеспечивает кодирование, посредством которого могут вводиться и определяться типы и подтипы, образующие основу для конструирования системы элементов;
- Предоставляет механизм, с помощью которого кодировщик может сформировать список имён всех элементов с соответствующими значениями и дать текстовое описание сущности каждого из них;
- Реализует механизм декларирования разных типов ограничений, в соответствии с которыми осуществляется контроль достоверности различных типов структур элементов на основе использования теоретических принципов, установленных логикой выделения типов элементов; этими ограничениями могут задаваться диапазон допустимых значений элемента, разрешённые типы структур элементов или запрет на совместное вхождение в ту или иную структуру определённых пар значений элементов; первоисточником таких ограничений обычно бывает подлежащая моделированию предметная область;
- Предоставляет механизм, посредством которого кодировщик может определять подразумеваемую интерпретацию недоопределённых структур элементов: например, механизм определения значений по умолчанию (литеральных или вычисляемых) для опущенных элементов.

Схема, описанная в данной части ISO 24610, может применяться для документирования любой системы элементов, но предназначена, главным образом, для использования в рамках представлений типизированных структур элементов, определённых в ISO 24610-1. Такие представления задают структуры данных, подчиняющиеся условиям выделения типов и конкретным ограничениям, определяемым с помощью ISO 24610-2. Представления структур элементов по ISO 24610-1 используются также применительно к некоторым элементам, определённым в ISO 24610-2.

2 Нормативные ссылки

Для применения данного документа необходимо обеспечение соответствия приведённым ниже нормативным документам. Применительно к недатированным ссылочным документам (с плавающими ссылками) действующим остаётся самое последнее издание нормативного документа.

ISO 24610-1:2006, *Управление языковыми ресурсами. Структуры элементов. Часть 1: Представление структур элементов*

ISO/IEC 19757-2, *Информационные технологии. Язык определения схемы документа (DSDL). Часть 2. Валидация на основе регулярной грамматики. RELAX NG*

3 Термины и определения

Для целей данного документа используются термины и определения из стандарта ISO 19757-2, а также терминология, приведённая ниже.

3.1
ограничение по допустимости
admissibility constraint
ограничение по разрешённым элементам
feature admissibility constraint
спецификация множества **разрешённых элементов** (3.2) и **допустимых значений элементов** (3.3), ассоциируемая с конкретным **типом** (3.24)

3.2
разрешённый элемент
admissible feature
подходящий элемент
appropriate feature
элемент, для которого соответствующая **структура элементов** (3.14) определённого **типа** (3.24) может нести в себе конкретное **значение** (3.17)

ПРИМЕЧАНИЕ В некоторых интерпретациях этот термин часто приобретает оттенок обязательности, то есть считается, что структуры элементов конкретного типа должны содержать в себе значение для каждого разрешённого элемента. Однако в нашем случае данный термин не предполагает обязательного присутствия элемента.

3.3
разрешённое значение элемента
admissible feature value
допустимое значение
admissible value
ограничение по значениям
value restriction
ограничение по диапазону
range restriction
значение (3.17), которое должно быть отнесено к категории **допустимых элементов** (3.2) в **структурах элементов** (3.14) данного **типа** (3.24)

3.4
атомарный тип
atomic type
пользовательский **тип** (3.24), который не имеет декларируемых или наследуемых **допустимых элементов** (3.2)

3.5
множество с повторяющимися элементами
bag
мультимножество
multiset
триплет, образованный целым числом n , множеством S и функцией отображения целых чисел в диапазоне от 1 до n , в элементы S

ПРИМЕЧАНИЕ Множество с повторяющимися элементами — это промежуточный объект между обычным множеством (как совокупностью неупорядоченных элементов) и списком (где отдельные элементы могут встречаться многократно).

3.6

встроенный элемент

built-in

элемент, не определяемый пользователем, но могущий появиться вместо **структуры элементов** (3.14), например, в качестве **значения элемента** (3.17)

ПРИМЕЧАНИЕ Встроенные элементы могут быть атомарными или составными. К первым относятся численные, строковые, символьные и двоичные элементы; ко вторым — **коллекции** (3.7) и применяемые логические операторы: например, дизъюнкция, отрицание и слияние (5.2.4).

3.7

коллекция

collection

значение элемента (3.17), содержащее совокупность возможных значений, которые представлены в виде списка, обычного множества или **множества с повторяющимися элементами** (3.5)

3.8

ограничение

constraint

компонент спецификации, которая идентифицирует некоторую коллекцию **структур элементов** (3.14) как неадекватную

ПРИМЕЧАНИЕ 1 Все ограничения по своей синтаксической форме имплицитивны, хотя некоторые из них выделяются как ограничения по допустимости. См. **адекватность** (3.31) и 5.4. Все структуры элементов, которые не исключены явным образом как неадекватные, считаются адекватными.

ПРИМЕЧАНИЕ 2 Структура элементов, не идентифицированная таким образом как не соответствующая никакому из ограничений в системе элементов, считается адекватной.

3.9

значение по умолчанию, стандартное значение

default value

значение (3.17), присваиваемое **элементу** (3.12) в том случае, когда оно не определено

ПРИМЕР В датском языке при отсутствии явного указания грамматического рода ему присваивается значение “мужской”.

ПРИМЕЧАНИЕ Структура элементов не может содержать элементов, для которых не указано соответствующее значение.

3.10

пустая структура элементов

empty feature structure

структура элементов (3.14), не содержащая никакой информации

ПРИМЕЧАНИЕ Пустая структура элементов категоризирует все другие структуры элементов.

3.11

расширение

extension

преобразование типа **категоризации** (3.21)

ПРИМЕЧАНИЕ Структура элементов F расширяет G тогда и только тогда, когда G категоризирует F .

3.12

элемент

feature

свойство или аспект объекта, формально представляемые как функция, отображающая объект в его соответствующее **значение** (3.17)

3.13

спецификация элементов

feature specification

связывание **элемента** (3.12) с его **значением** (3.17) в описании структуры элементов

3.14

структура элементов

feature structure

структура записей, которая ставит в соответствие каждой коллекции элементов одно **значение** (3.17)

ПРИМЕЧАНИЕ 1 Каждое значение представляет собой структуру элементов или более простой **встроенный элемент** (3.6), такой как строка.

ПРИМЕЧАНИЕ 2 Структуры элементов частично упорядочены. Минимальными в этом упорядочении являются пустые структуры элементов.

3.15

система элементов

feature system

иерархия типов (3.26), в которой каждый **тип** (3.24) ассоциируется с коллекцией **ограничений по допустимости** (3.1) и **импликативными ограничениями** (3.18)

ПРИМЕЧАНИЕ ср. **декларация типа** (3.25)

3.16

декларация системы элементов

feature system declaration,

FSD

описание конкретной **системы элементов** (3.15)

3.17

значение для элемента

feature value

значение

value

объект или совокупность объектов, характеризующие некоторое свойство другого объекта

3.18

импликативное ограничение

implicational constraint

ограничение типа “если G , то H ,” где G и H — это **структуры элементов** (3.14)

ПРИМЕЧАНИЕ Такое ограничение идентифицирует любую структуру элементов F как неадекватную, когда G категоризирует F , а F и H обычно не имеют адекватного расширения. См. **категоризация** (3.21) и 8.5. Часто ограничение такого вида используется при обращении к импликативным ограничениям, которые одновременно не являются ограничениями по допустимости.

3.19

интерпретация

interpretation

минимально информативное (то есть наиболее общее) **расширение** (3.11) **структуры элементов** (3.14), которое совместимо с множеством ограничений, объявленным в **декларации системы элементов** (3.16)

3.20**частичный порядок****partial order****частично упорядоченное множество****partially ordered set**

множество S , для которого определено отношение $\leq$ на $S \times S$, которое 1) рефлексивно (для всех $s \in S$, $s \leq s$), 2) антисимметрично (для всех $p, q \in S$, если $p \leq q$ и $q \leq p$, то $p = q$), и 3) транзитивно (для всех $p, q, r \in S$, если $p \leq q$ и $q \leq r$, то $p \leq r$)

ПРИМЕЧАНИЕ Множество целых чисел Z частично упорядочено, но дополнительно оно обладает свойством, согласно которому, для каждого $p, q \in Z$, выполняется условие $p \leq q$ или $q \leq p$. Этим свойством обладает не любой частичный порядок. Например, такой частичный порядок, как таксономическая классификация организмов по типам, родам и видам, указанным свойством не обладает; не обязательно обладают этим свойством также иерархии типов. Типизированные структуры элементов системы не имеют этого свойства, если (а) данное свойство присуще иерархии их типов, и (б) иерархия типов состоит из единственного типа либо каждый тип у ограничен присутствием одного-единственного подходящего элемента.

3.21**категоризация****subsumption**

свойство, связывающее две структуры элементов G и F таким образом, что G считается принадлежащей F тогда и только тогда, когда F несёт в себе всю информацию, которую содержит G

ПРИМЕЧАНИЕ Формальное определение представлено ниже, в 5.6.

3.22**подтип****subtype**

тип (3.24), на который распространяются ограничения и соответствующие характеристики, содержащиеся в другом типе

3.23**супертип, надтип****supertype****базовый тип****base type**

тип (3.24), от которого другой тип наследует ограничения и соответствующие элементы

ПРИМЕЧАНИЕ s является подтипом t тогда и только тогда, когда t – супертип s . Каждый тип является подтипом и супертипом самого себя.

3.24**семантический тип****semantic type**

тип, характеризующий выражение, с помощью которого коллекция **структур элементов** (3.14) различается как идентифицируемый и концептуально значимый класс

ПРИМЕЧАНИЕ Как это следует из имени *семантический тип*, типы, о которых идёт речь в данной части ISO 24610, не предназначены для различения структур элементов или их спецификаций по синтаксису.

3.25**декларация типа****type declaration**

информационная структура, декларирующая **супертипы** (3.23), **допустимые элементы** (3.2), **значения допустимых элементов** (3.3), **ограничения по допустимости** (3.1) и **импликативные ограничения** (3.18) для данного **типа** (3.24)

ПРИМЕЧАНИЕ Ограничения, накладываемые на тип в результирующей системе элементов, это ограничения, объявленные в декларации дополнительно к унаследованным от супертипов.

3.26

иерархия типов

type hierarchy

частичный порядок (3.20) на множестве **типов** (3.24)

ПРИМЕЧАНИЯ См. ISO 24610-1:2006, Приложение С, *Наследуемые иерархии типов*.

3.27

типизированная структура элементов

typed feature structure, TFS

структура элементов (3.14), несущая в себе **тип** (3.24)

3.28

типизация

typing

присваивание семантического **типа** (3.24) **встроенному элементу** (3.6) либо **структуре элементов** (3.14), атомарной или составной

ПРИМЕЧАНИЕ Семантические типы в системах элементов частично упорядочены и имеют множественные отношения наследования.

3.29

недоопределение

underspecification

предоставление неполной информации о **значении** (3.17)

ПРИМЕЧАНИЕ Недоопределение обычно категоризирует одно значение из диапазона возможных значений, которые могут быть сведены к единственному значению путём последовательного наложения ограничений. См. **категоризация** (3.21).

3.30

формальная правильность

well-formedness

синтаксическое соответствие представления **структуры элементов** (3.14) и ISO 24610-1

<https://standards.iteh.ai/catalog/standards/iso/87dd2075-4d96-4e49-bcb9-43361d03fc99/iso-24610-2-2011>

3.31

адекватность

validity

соответствие **типизированной структуры элементов** (3.27) действующим **ограничениям** (3.8) конкретной **системы элементов** (3.15)

ПРИМЕЧАНИЕ См. Раздел 6.

4 Общая структура стандарта

Основное содержание настоящего документа отражено в четырёх разделах – 5, 6, 7 и 8.

- В Разделе 5, *Базовые понятия*, рассматривается определение типизированных структур элементов, и вводятся понятия атомарных и составных типов структур элементов, коллекций и прочих операторов, могущих фигурировать в значениях элементов; затем описываются понятия наследуемых типов, иерархий типов, ограничений типов, значений по умолчанию и недоопределения, которые имеют важнейшее значение для конструирования систем элементов.
- В Разделе 6, *Определение формальной правильности и адекватности*, обсуждаются условия отмеченности и достоверности структур элементов.

- Раздел 7, *Система элементов для грамматики*, иллюстрирует способ определения типов с использованием иерархии и ограничений типов, в рамках которых декларируются допустимые элементы и значения для конкретных типов.
- В Разделе 8, *Декларация системы элементов*, показывается, каким образом система элементов может быть декларирована и преобразована в валидатор.

Эта главная часть документа включает в себя два приложения: Приложение А содержит the XML-схему для данной части ISO 24610; Приложение В содержит развёрнутый пример.

5 Базовые понятия

5.1 Рассматриваемые типизированные структуры элементов

Типизированные структуры элементов (TFS) вводятся как базовые записи для управления языковыми ресурсами.

Для получения более подробной информации следует обратиться к ISO 24610-1:2006, 4.7, *Типизированные структуры элементов* и Приложению С, *Типизированные иерархии наследования*.

В данном документе TFS определяется формально как кортеж на конечном множестве элементов **Feat**, который состоит из коллекции X элементов, не входящих в структуру, и иерархии типов **Type** с отношением $\leq$, где **Type** – это конечное множество типов, а отношение $\leq$ определяет выделение подтипов на множестве **Type**.

Структура элементов представляет собой кортеж $\langle Q, \gamma, \theta, \delta \rangle$, в котором:

- Q – множество узлов,
- $\gamma \in Q$ – корневой узел структуры элементов,
- $\theta : Q \rightarrow \mathbf{Type}$ является функцией частичного упорядочения, и
- $\delta : \mathbf{Feat} \times Q \rightarrow Q \cup X$ – функция частичного означивания элементов, такая, что для всех $q \in Q$, существует последовательность элементов $F_1, \dots, F_n$, в которой $\delta[F_1, \dots, \delta(F_1, \gamma) \dots] = q$.

Обозначение $\langle fs \rangle$ показывает узлы. Приведённое выше определение отличается от стандартного, используемого в лингвистике и теории вычислительных систем, тем, что, во-первых, типизация осуществляется частично, а не полностью (то есть типы определяются не для всех структур элементов), и, во-вторых, значения элементов не обязательно должны представлять собой структуры элементов; однако эти значения могут извлекаться из коллекции, отмеченной другими элементами XML, – такими, как строковые, численные, символьные и двоичные (выше им соответствует обозначение X). Следует заметить, что узлы типизируются, тогда как сами элементы – нет.

Приведённое ниже XML-представление структуры элементов считается формально правильным; в нём атрибут “тип” указывается для каждого из двух элементов $\langle fs \rangle$.

ПРИМЕР Типизированная структура элементов:

```
<fs type="word">
  <f name="orth">
    <string>had</string>
  </f>
  <f name="morphoSyntax">
    <fs type="verb">
      <f name="tense">
        <symbol value="past"/>
      </f>
    </f>
  </f>
</fs>
```

```

    <f name="auxiliary">
      <binary value="false"/>
    </f>
  </fs>
</f>
</fs>

```

Имя элемента ORTH обозначает орфографию, то есть общепринятое написание слова или фразы.

Данное XML-представление показывает, каким образом определяются морфосинтаксические характеристики английского слова “had” как не вспомогательного глагола в прошедшем времени.

В альтернативной “матричной” или “AVM” нотации имена типов обычно пишутся строчными буквами, иногда курсивом или текстовым типографским шрифтом; имена элементов пишутся заглавными буквами, а строковые элементы заключаются в кавычки. Двоичные значения отмечаются знаками “плюс” (+) или “минус” (-). В данном документе эти соглашения тоже соблюдаются. Представленная выше структура элементов должна при использовании матричной нотации выглядеть так, как показано на Рисунке 1.

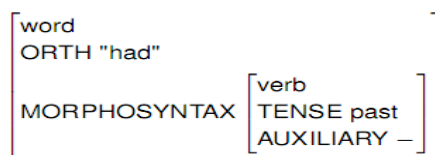


Рисунок 1 — Матричная нотация

5.2 Типы

5.2.1 Атомарные типы

Наряду со структурами с встроенными элементами (<symbol>, <string>, <numeric> и <binary>) могут существовать структуры элементов, имеющие тип, но не имеющие элементов. Такие структуры называются простыми, или атомарными структурами элементов, а типы, которые допускают отсутствие элементов в декларации системы элементов (FSD), именуются атомарными типами.

В результате всегда имеется возможность декларирования новых атомарных типов и использования их вместо вышеупомянутых встроенных элементов для задания простых значений. Например, приведённая выше структура элементов при условии декларирования в FSD дополнительных типов had, past и false могла бы быть представлена так, как показано ниже.

ПРИМЕР Альтернативная формулировка типизированной структуры элементов:

```

<fs type="word">
  <f name="orth">
    <fs type="had"/>
  </f>
  <f name="morphoSyntax">
    <fs type="verb">
      <f name="tense">
        <fs type="past"/>
      </f>
      <f name="auxiliary">
        <binary value="false"/>
      </f>
    </fs>
  </f>
</fs>

```