

Отображение простых типов данных XML Schema в типы языка СИНТЕЗ

Брюхов Д.О., Тюрин И.Н.
Институт Проблем Информатики Российской Академии Наук
e-mail: {brd, turin}@ipi.ac.ru

Аннотация

Статья¹ посвящена построению отображения встроенных типов данных XML Schema в типы языка СИНТЕЗ.

Введение

В лаборатории композиционных методов проектирования информационных систем ИПИ РАН разрабатывается архитектура посредника неоднородных информационных коллекций [3], который позволяет работать с распределенными неоднородными коллекциями данных как с интегрированной коллекцией информации.

Посредник поддерживает процесс систематической регистрации и классификации коллекций, содержит унифицированные онтологические данные и метаинформацию для улучшения обнаружения и композиции существующих коллекций [1].

Регистрация коллекции - это процесс взаимодействия провайдера цифровой коллекции с предметным посредником во время операционной фазы посредника, заключающийся в контекстуализации коллекции в посреднике (согласовании ее понятийного и терминологического контекста с контекстом посредника), представления классов коллекции как материализованных взглядов над классами посредника, генерации адаптеров.

В рамках посредника определяется метаинформация предметной области, которую представляет данный посредник. В качестве канонической модели данных посредника используется язык СИНТЕЗ [2].

Последние несколько лет консорциум W3C развивает стандарт языка разметки XML (eXtended Markup Language) в качестве основного носителя информации в Сети. С помощью данного стандарта строятся словари (DTD или XML Schemas) для передачи более специализированной информации.

XML Schema [4] определяет конкретный словарь XML - конкретный набор элементов разметки (тэгов) и ограничения, связанные с ними. Например, существует XML Schema для определения тэгов HTML, MathML (язык разметки математических выражений), DAML+OIL (язык описания онтологий) и других языков разметки.

В данной статье рассматривается отображение встроенных типов данных XML Schema в типы данных языка СИНТЕЗ. Данное отображение используется при регистрации коллекций для загрузки их спецификаций, представленных в XML Schema, в базу метаинформации посредника.

¹ Данная работа выполняется в рамках гранта РФФИ №01-07-90084.

Система простых типов XML Schema

Простые типы данных XML Schema делятся на две группы: примитивные (primitive) и выводимые (derived). К примитивным типам данных относятся такие типы, как string, boolean, double, decimal и др. Выводимые типы строятся на основе примитивных типов. Ряд выводимых типов является predetermined (например, integer выводится из decimal)

Любой простой тип в XML Schema определяется множеством своих значений и множеством лексических представлений (lexical representation). Операции над типами не определяются. Построение новых простых типов осуществляется с помощью одного из следующих методов:

- выводение по ограничению (deriving by restriction);
- выводение по списку (deriving by list);
- выводение по объединению (deriving by union);

Выведение по ограничению строится с помощью ограничивающих фасетов (constraining facets). Также отметим тот факт, что в XML Schema нет встроенных типов множеств, вместо этого вводятся понятия выведения по списку и объединению.

Приведем примеры построения новых типов:

- по ограничению

```
<xsd:simpleType name="over12">  
  <xsd:restriction base="xsd:decimal">  
    <xsd:minInclusive value="13" />  
  </xsd:restriction>  
</xsd:simpleType>
```

тип over12 выводится на основе примитивного типа decimal заданием ограничения снизу

- по списку

```
<xsd:simpleType name='listOfInteger'>  
  <xsd:list itemType='integer' />  
</xsd:simpleType>
```

тип listOfInteger является списком элементов типа integer

- по объединению

```
<xsd:simpleType name="clothingsize">  
  <xsd:union>  
    <xsd:simpleType>  
      <xsd:restriction base="integer" />  
    </xsd:simpleType>  
    <xsd:simpleType>  
      <xsd:restriction base="string" />  
    </xsd:simpleType>  
  </xsd:union>  
</xsd:simpleType>
```

значением элемента типа clothesize может быть как целое число, так и строка

Система типов языка СИНТЕЗ

В основе объектной модели лежит понятие абстрактного типа данных (АТД), служащее для описания неизменяемых типов данных любой природы. Описание абстрактного типа данных инкапсулирует спецификации атрибутов, ассоциаций и операций типа. Операции типов описываются типом функции. Наряду с АТД в СИНТЕЗе определен набор базовых типов, таких как integer, string, boolean, list, enum и др.

Отображение простых типов данных XML Schema в типы данных языка СИНТЕЗ

Отображение predefined типов

Среди predefined типов XML Schema мы выделяем типы, для которых существуют эквивалентные типы СИНТЕЗа, и для которых не существует эквивалентных типов. Для первой группы ниже приведена таблица соответствия типов:

| Типы XML Schema | Типы СИНТЕЗа |
|-----------------|----------------|
| AnySimpleType | Tbuilt_in |
| String | String |
| Boolean | Boolean |
| Float | Float |
| Double | Double |
| Duration | Interval |
| DateTime | Time |
| Integer | Integer |
| Int | Long |
| UnsignedInt | unsigned long |
| Short | Short |
| UnsignedShort | unsigned short |

Для каждого типа из второй группы строится соответствующий ему АТД СИНТЕЗа. Приведем пример спецификации АТД соответствующего типу anyURI XML Schema.

```
{ xmlAnyURI;  
  in: type;  
  value: string;  
}
```

Отображение типов, выводимых по ограничению

Данные типы XML Schema отображаются в АТД СИНТЕЗа. Поскольку в СИНТЕЗе АТД не могут являться подтипами встроенных типов данных, при отображении мы строим АТД, в атрибуте value которого хранится множество значений отображаемого типа.

Ограничение (restriction) отображается в тип атрибута value, а фасеты отображаются в инварианты для атрибута value. Например:

XML Schema:

```
<xsd:simpleType name="over17">
```

```

    <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="18"/>
    </xsd:restriction>
</xsd:simpleType>

```

СИНТЕЗ:

```

{Over17;
  in: type;
  value: integer;
  metaslot
    inv: {in: predicate, invariant;
          {{all o/Over17 ( o.value>=18 ) }}
    end
}

```

Исключением из этого правила является отображение фасета enumeration, который удобнее отображать с помощью встроенного типа СИНТЕЗа enum:

XML Schema:

```

<xsd:simpleType name="enumerationHeight">
  <xsd:restriction base="string">
    <xsd:enumeration value="short"/>
    <xsd:enumeration value="medium"/>
    <xsd:enumeration value="tall"/>
  </xsd:restriction>
</xsd:simpleType>

```

СИНТЕЗ:

```

{EnumerationHeight;
  in: type;
  value: {enum; enumlist: {'short', 'medium', 'tall' }}
}

```

Отображение типов, выводимых по списку

Данные типы XML Schema отображаются в АДД СИНТЕЗа, в атрибуте value которого хранится множество значений отображаемого типа. Типом атрибута value является тип СИНТЕЗа list. Тип элементов списка (type_of_element) соответствует ограничению по списку в XML Schema (itemType).

XML Schema:

```

<xsd:simpleType name="listOfString">
  <xsd:list itemType='xsd:string'/>
</xsd:simpleType>

```

СИНТЕЗ:

```

{ListOfString;
  in: type;
  value: { list; type_of_element: string }
}

```

Отображение механизма ограничения по объединению

Данные типы XML Schema отображаются в АД СИНТЕЗа, в атрибуте value которого хранится множество значений отображаемого типа. Типом атрибута value является тип СИНТЕЗа union.

XML Schema:

```
<xsd:simpleType name="clothingsize">
  <xsd:union>
    <xsd:simpleType>
      <xsd:restriction base='integer' />
    </xsd:simpleType>
    <xsd:simpleType>
      <xsd:restriction base='string' />
    </xsd:simpleType>
  </xsd:union>
</xsd:simpleType>
```

СИНТЕЗ:

```
{Clothingsize;
  in: type;
  value: {union;
    type_of_label: string;
    label1: integer;
    label2: string;
  }
}
```

Заключение

В данной статье представлены правила отображения типов данных XML Schema в СИНТЕЗ.

Литература

- [1] Briukhov, D.O., Kalinichenko, L.A., Skvortsov, N.A. Information sources registration at a subject mediator as compositional development. In Proceedings of the Fifth East European Symposium on Advances in Databases and Information Systems (ADBIS'01), Springer-Verlag, 2001, pp. 70-83.
- [2] Kalinichenko L.A. SYNTHESIS: the language for description, design and programming of the heterogeneous interoperable information resource environment, Moscow, 1995.
- [3] Kalinichenko L.A., Briukhov D.O., Skvortsov N.A., Zakharov V.N. Infrastructure of the subject mediating environment aiming at semantic interoperability of heterogeneous digital library collections. 2nd Russian Conference "DIGITAL LIBRARIES: Advanced Methods and Technologies, Digital Collections", September 26-28, 2000, Protvino
- [4] XML Schema Language Part 2: Datatypes. <http://www.w3.org/TR/2001/PR-xmlschema-2-20010330/>. W3C Proposed Recommendation. March, 2001.