# Лекция 7. Язык XML

Александр Смаль

**CS центр**
1 апреля 2013
Санкт-Петербург

# SGML

- 1969 году в IBM разработан язык GML (Generalized Markup Language).
- В 80-е на основе GML разрабатывается язык SGML (Standard Generalized Markup Language) — стандартный обобщённый язык разметки, метаязык, на котором можно определять язык разметки для документов.
- HTML и XML произошли от SGML. HTML — это приложение SGML, а XML — это подмножество SGML, разработанное для упрощения процесса машинного разбора документа.
- Другими приложениями SGML являются SGML Docbook (документирование) и «Z Format» (типография и документирование).

# Структура SGML

- SGML-декларация — определяет, какие символы и ограничители могут появляться в приложении;
- Document Type Definition — определяет синтаксис конструкций разметки. DTD может включать дополнительные определения, такие, как символьные ссылки-мнемоники;
- Спецификация семантики, относится к разметке — также даёт ограничения синтаксиса, которые не могут быть выражены внутри DTD;
- Содержимое SGML-документа — по крайней мере, должен быть корневой элемент.

# Примеры

DTD:

```
<!ELEMENT lines (line*)
<!ELEMENT line O - (#PCDATA)>
<!ENTITY   line-tagc  ‘‘</line>’’>
<!SHORTREF one-line ‘‘&#RE;&#RS;’’ line-tagc>
<!USEMAP   one-line line>
```

SGML:

```
<lines>
first line
second line
</lines>
```

```
<lines>
<line>first line</line>
<line>second line</line>
</lines>
```

# XML

- XML (eXtensible Markup Language) — расширяемый язык разметки, рекомендованный Консорциумом Всемирной паутины (W3C).
- XML разрабатывался как
  - язык с простым формальным синтаксисом,
  - удобный для создания и обработки документов программами,
  - одновременно удобный для чтения и создания документов человеком,
  - нацеленный на использование в Интернете.

# Устройство XML

- Определены следующие замены

| | |
|---|---|
| < | < |
| > | > |
| & | & |
| ' | ' |
| " | " |

- Секция CDATA не является логической единицей текста. Внутри секции могут находиться любые символьные данные без замен.

```
<![CDATA[ any text < > &   ]]>.
```

# Структура

- Пролог

```
<?xml version="1.1" encoding='UTF-8' ?>
```

- Комментарии: `<!-- comments -->`
- Элемент — структурная единица. Границы элементов представлены начальным и конечным тегами.

```
<element attr1="value1" attr2="value2">
        ... contents ...
</element>
<element attr1="value2" attr2="value4" />
```

- Обязательно наличие корневого элемента.

# DTD

Существует два основных способа описания структуры XML: DTD и XSD. DTD (Document Type Definition) — аналогично описанию SGML.

```
<!ELEMENT people_list (person*)>
<!ELEMENT person (name,birthdate?,gender?,socialsecuritynumber?)>
<!ELEMENT name (#PCDATA) >
<!ELEMENT birthdate (#PCDATA) >
<!ELEMENT gender (#PCDATA) >
<!ELEMENT socialsecuritynumber (#PCDATA) >

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE people_list SYSTEM "example.dtd">
<people_list>
    <person>
        <name>Fred Bloggs</name>
        <birthdate>27/11/2008</birthdate>
        <gender>Male</gender>
        <socialsecuritynumber>1234567890</socialsecuritynumber>
    </person>
</people_list>
```

# XSD

XML Schema — язык описания структуры XML-документа. Спецификация XML Schema является рекомендацией W3C.

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="country" type="country"/>
  <xs:complexType name="country">
    <xs:sequence>
      <xs:element name="name" type="xs:string"/>
      <xs:element name="population" type="xs:decimal"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

```
<?xml version="1.0" encoding="utf-8"?>
<country>
    <name>France</name>
    <population>59.7</population>
</country>
```

# XSLT

XSLT (eXtensible Stylesheet Language Transformations) — язык преобразования XML-документов. Спецификация XSLT входит в состав XSL и является рекомендацией W3C.

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
                                                    version="1.0">
    <xsl:output method="xml" indent="yes"/>
    <xsl:template match="persons">
        <transform>
            <xsl:apply-templates/>
        </transform>
    </xsl:template>
    <xsl:template match="person">
        <record>
            <xsl:apply-templates select="@*|*"/>
        </record>
    </xsl:template>
    ...
    <xsl:template match="surname"/>
</xsl:stylesheet>
```

# XPath

XPath (XML Path Language) — язык запросов к элементам XML-документа.

```
<html>
 <body>
    <div>AAAA
      <span>BBBB</span>
    </div>
    <div>CCCC</div>
    <div>DDDD
      <span class="text">EEEE</span>
      <span class="text">FFFF</span>
      <span>GGGG</span>
    </div>
  </body>
</html>
```

Запрос `/html/body/*/span[@class]` соответствует строкам с `EEEE` и `FFFF`.

# XQuery

XQuery — язык запросов, разработанный для обработки данных в формате XML. XQuery использует XML как свою модель данных.

```
<html><head/><body>
 {
   for $act in doc("hamlet.xml")//ACT
   let $speakers := distinct-values($act//SPEAKER)
   return
     <span>
       <h1>{ $act/TITLE/text() }</h1>
       <ul>
       {
         for $speaker in $speakers
         return <li>{ $speaker }</li>
       }
       </ul>
     </span>
 }
 </body>
</html>
```

# Способы разбора XML

Два основных подхода:

- DOM (Document Object Model)
  - Естественное соответствие древовидной структуры документа и его объектной модели.
  - Естественно представлять
  - Необходимо прочитать весь документ в память.
- SAX (Simple API for XML)
  - Событийный подход к разбору XML.
  - Можно обрабатывать только некоторую часть документа.
  - Позволяет не хранить весь документ в памяти.

# Языки основанные на XML

- XHTML — версия HTML, отвечающая синтаксическим требованиям XML.
- SOAP — протокол передачи данных, использующий для сообщений формат XML.
- FB2 — формат описания книг, базирующийся на XML.
- WSDL — формат описания сервисов.
- RSS и Atom — семейство XML-форматов, предназначенных для описания лент новостей, анонсов статей, изменений в блогах.
- OpenDocument Format, Office Open XML — форматы файлов для офисных документов (Open Office и MS Office).

# Альтернативы XML: YAML

YAML (YAML Ain't Markup Language, Yet Another Markup Language) — язык разметки с облегчёнными синтаксисом.

- краток и понятен;
- очень выразительный и расширяемый;
- допускает простой потоковый интерфейс;
- использует структуры данных, родные для языков программирования;
- легко реализуется, возможно, слишком легко;
- использует цельную модель данных. Нет исключений — нет беспорядка.

# Альтернативы XML: Пример YAML

```
receipt:     Oz-Ware Purchase Invoice
date:        2007-08-06
customer:
    given:   Dorothy
    family:  Gale

items:
    - part_no:   A4786
      descrip:   Water Bucket (Filled)
      price:     1.47
      quantity:  4

    - part_no:   E1628
      descrip:   High Heeled "Ruby" Slippers
      size:      8
      price:     100.27
      quantity:  1
```

# Альтернативы XML: JSON

JSON (англ. JavaScript Object Notation) — текстовый формат обмена данными, основанный на JavaScript.
JSON является подмножеством языка YAML (спецификация YAML 1.2), т.е. «любой файл в формате JSON является корректным файлом в формате YAML».

```
{   "firstName": "John",
    "lastName": "Smith",
    "age": 25,
    "address": {
        "streetAddress": "21 2nd Street",
        "city": "New York",
        "state": "NY",
        "postalCode": 10021 },
    "phoneNumbers": [
        {   "type": "home", "number": "212 555-1234" },
        {   "type": "fax", "number": "646 555-4567" }
    ]
}
```

# Спасибо за внимание!