Глава 16
XML & JAVA

XML (*Extensible Markup Language* – расширяемый язык разметки) – рекомендован W3C как язык разметки, представляющий свод общих синтаксических правил. XML предназначен для обмена структурированной информацией с внешними системами. Формат для хранения должен быть эффективным, оптимальным с точки зрения потребляемых ресурсов (памяти, и др.). Такой формат должен позволять быстро извлекать полностью или частично хранимые в этом формате данные и быстро производить базовые операции над этими данными.

XML является упрощённым подмножеством языка SGML. На основе XML разрабатываются более специализированные стандарты обмена информацией (общие или в рамках организации, проекта), например XHTML, SOAP, RSS, MathML.

Основная идея XML – это текстовое представление с помощью тегов, структурированных в виде дерева данных. Древовидная структура хорошо описывает бизнес-объекты, конфигурацию, структуры данных и т.п. Данные в таком формате легко могут быть как построены, так и разобраны на любой системе с использованием любой технологии – для этого нужно лишь уметь работать с текстовыми документами. С другой стороны, механизм **namespace**, различная интерпретация структуры XML документа (триплеты RDF, microformat) и существование смешанного содержания (mixed content) часто превращают XML в многослойную структуру, в которой отсутствует древовидная организация (разве что на уровне синтаксиса).

Почти все современные технологии стандартно поддерживают работу с XML. Кроме того, такое представление данных удобочитаемо (human-readable). Если нужен тег для представления имени, его можно создать:

<name>Java SE 6</name> или **<name/>**.

Далее представлены примеры неправильных написаний тегов:

```
<?xml version="1.0"?>
<book>
   <title>title</title>
</book>
<book/>
```

Каждый XML-документ должен содержать только один корневой элемент (root element или document element). В примере есть два корневых элемента, один из которых пустой. В отличие от файла XML, файл HTML может иметь несколько корневых элементов и не обязательно <HTML>.

```
<?xml version="1.0"?>
<book>
   <caption>C++
</book>
   </caption>
```

Тег должен закрываться в том же теге, в котором был открыт. В данном случае это **caption**. В HTML этого правила не существует.

```
<?xml version="1.0"?>
<book>
   <author>Petrov
</book>
```

Любой открывающий тег должен иметь закрывающий. Если тег не имеет содержимого, можно использовать конструкцию вида **<author/>**. В HTML есть возможность не закрывать теги, и браузер определяет стили по открывающемуся тегу

Наименования тегов являются чувствительные к регистру (case-sensitive), т.е. например теги, **<author>**, **<Author>**, **<AuToR>** будут совершенно разными при работе с XML:

```
<author>Petrov</Author>
```

Программа-анализатор просто не найдет завершающий тег и выдаст ошибку. Язык HTML нетребователен к регистру.

Все атрибуты тегов должны быть заключены либо в одинарные, либо в двойные кавычки:

```
<book dateOfIssue="09/09/2007" title='JAVA in Belarus'/>
```

В HTML разрешено записывать значение атрибута без кавычек.

Например: **<FORM method=POST action=index.jsp>**

Пусть существует XML-документ с данными о студентах:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE students SYSTEM "students.dtd">
<students>
   <student login="mit" faculty="mmf">
      <name>Mitar Alex</name>
      <telephone>2456474</telephone>
      <address>
         <country>Belarus</country>
         <city>Minsk</city>
         <street>Kalinovsky 45</street>
      </address>
   </student>
   <student login="pus" faculty="mmf">
      <name>Pashkun Alex</name>
      <telephone>3453789</telephone>
      <address>
         <country>Belarus</country>
         <city>Brest</city>
         <street>Knorina 56</street>
      </address>
   </student>
</students>
```

Каждый документ начинается декларацией – строкой, указывающей как минимум версию стандарта XML. В качестве других атрибутов могут быть указаны кодировка символов и внешние связи.

После декларации в XML-документе могут располагаться ссылки на документы, определяющие структуру текущего документа и собственно XML-элементы (теги), которые могут иметь атрибуты и содержимое. Открывающий тег состоит из имени элемента, например **<city>**. Закрывающий тег состоит из того же имени, но перед именем добавляется символ '**/**', например **</city>**. Содержимым элемента (content) называется всё, что расположено между открывающим и закрывающим тегами, включая текст и другие (вложенные) элементы.

Инструкции по обработке

XML-документ может содержать инструкции по обработке, которые используются для передачи информации в работающее с ним приложение. Инструкция по обработке может содержать любые символы, находиться в любом месте XML документа и должна быть заключены между **<?** и **?>** и начинаться с идентификатора, называемого **target** (цель).

Например:

```
<?xml-stylesheet type="text/xsl" href="book.xsl"?>
```

Эта инструкция по обработке сообщает браузеру, что для данного документа необходимо применить стилевую таблицу (stylesheet) **book.xsl**.

Комментарии

Для написания комментариев в XML следует заключать их, как и в HTML, между **<!--** и **-->** . Комментарии можно размещать в любом месте документа, но не внутри других комментариев:

```
<!-- комментарий <!-- Неправильный -->  -->
```

Внутри значений атрибутов:

```
<book title="BLR<!-- Неправильный комментарий -->"/>
```

Внутри тегов:

```
<book <!-- Неправильный комментарий -->/>
```

Указатели

Текстовые блоки XML-документа не могут содержать символов, которые служат в написании самого XML: **<**, **>**, **&**.

```
<description>
      в текстовых блоках нельзя использовать символы <,>,&
</description>
```

В таких случаях используются ссылки (указатели) на символы, которые должны быть заключены между символами **&** и **;** .

Особо распространенными указателями являются:

< – символ **<**;

> – символ **>**;

& – символ **&**;

' – символ апострофа **'**;

" – символ двойной кавычки **"**.

Таким образом, пример правильно будет выглядеть так:

```
<description>
      в текстовых блоках нельзя использовать символы
      &lt;, &gt;, &amp;
</description>
```

Раздел CDATA

Если необходимо включить в XML-документ данные (в качестве содержимого элемента), которые содержат символы '**<**', '**>**', '**&**', ' ' ' и ' " ', чтобы не заменять их на соответствующие определения, можно все эти данные включить в раздел **CDATA**. Раздел **CDATA** начинается со строки "**<[CDATA[**", а заканчивается "**]]>**", при этом между ними эти строки не должны употребляться. Объявить раздел **CDATA** можно, например, так:

```
<data><[CDATA[ 5 < 7 ]]></data>
```

Корректность XML-документа определяют следующие два компонента:

- синтаксическая корректность (well-formed): то есть соблюдение всех синтаксических правил XML;
- действительность (valid): то есть данные соответствуют некоторому набору правил, определённых пользователем; правила определяют структуру и формат данных в XML. Валидность XML документа определяется наличием DTD или XML-схемы XSD и соблюдением правил, которые там приведены.

DTD

Для описания структуры XML-документа используется язык описания DTD (Document Type Definition). В настоящее время DTD практически не используется и повсеместно замещается XSD. DTD может встречаться в достаточно старых приложениях, использующих XML и, как правило, требующих нововведений (upgrade).

DTD определяет, какие теги (элементы) могут использоваться в XML-документе, как эти элементы связаны между собой (например, указывать на то, что элемент **<student>** включает дочерние элементы **<name>**, **<telephone>** и **<address>**), какие атрибуты имеет тот или иной элемент.

Это позволяет наложить четкие ограничения на совокупность используемых элементов, их структуру, вложенность.

Наличие DTD для XML-документа не является обязательным, поскольку возможна обработка XML и без наличия DTD, однако в этом случае отсутствует средство контроля действительности (validness) XML-документа, то есть правильности построения его структуры.

Чтобы сформировать DTD, можно создать либо отдельный файл и описать в нем структуру документа, либо включить DTD-описание непосредственно в документ XML.

В первом случае в документ XML помещается ссылка на файл DTD:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<! DOCTYPE students SYSTEM "students.dtd">
```

Во втором случае описание элемента помещается в XML-документ:

```
<?xml version="1.0" ?>
<! DOCTYPE student [
<!ELEMENT student (name, telephone, address)>
<!--
далее идет описание элементов name, telephone, address
-->
]>
```