

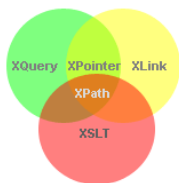
# XML Path Language

Муравьёв С.К.

Национальный исследовательский ядерный университет «МИФИ»  
Кафедра №36 «Информационные системы и технологии»

1 ноября 2013

# Что такое XPath?



- XPath позволяет определять части XML документа
- XPath использует выражения для навигации по XML документам
- XPath предоставляет набор стандартных функций
- XPath является важнейшим элементом XSLT
- XPath рекомендован W3C

# Выражения XPath

Выражения XPath используются для выбора единичных узлов или их множеств в документе XML. Эти выражения очень похожи на обычные пути поиска в файловой системе.

```
/universities/university/faculty/sector/department
```

# Стандартные функции XPath

- XPath включает в себя более ста встроенных функций.
- Существуют функции для работы со строковыми и числовыми значениями, сравнения времени и дат, управления узлами, последовательностями, и т.п.

```
concat(//firstname/text(), '_ ', //lastname/text())
```

- Описание стандартных функций можно найти по адресу:

```
http://www.w3schools.com/xpath/xpath\_functions.asp
```

# Использование XPath в XSLT

- XPath играет ключевую роль в стандарте XSLT. Без знания XPath невозможно создавать XSLT документы.
- На базе выражений XPath строятся языки XQuery и XPointer.
- XQuery 1.0 и XPath 2.0 имеют единую модель данных, а также общие функции и операторы.

```
1 <xsl:stylesheet version="1.0">
2   <xsl:template match="//student">
3     <p><xsl:value-of select="@name"/></p>
4   </xsl:template>
5 </xsl:stylesheet>
```

XPath оперирует узлами следующих типов:

- 1 документ (document)
- 2 элемент (element)
- 3 атрибут (attribute)
- 4 текст (text)
- 5 програнство имён (namespace)
- 6 комментарий (comment)
- 7 инструкция обработки (processing-instruction)

# Отношения узлов

- XML документы представляют собой иерархии узлов.
- В зависимости от взаимного расположения узлов в иерархии, между ними возникают следующие виды отношений:
  - родитель (parent)
  - ребёнок (children)
  - брат (sibling)
  - предок (ancestor)
  - потомок (descendant)
- Самый верхний узел в иерархии называется корневым элементом.

# Пример

- элемент book является родителем для элементов title, author, year, и price
- элементы title, author, year, и price являются детьми элемента book
- элементы title, author, year, и price являются братьями
- элемент bookstore является предком для узлов title, author, year, и price
- узлы title, author, year, и price являются потоками узла bookstore

```
1 <bookstore>
2   <book>
3     <title>Harry Potter</title>
4     <author>J K. Rowling</author>
5     <year>2005</year>
6     <price>29.99</price>
7   </book>
8 </bookstore>
```



# Выбор узлов

Выражения XPath позволяют осуществить выбор узлов в XML документе.

Выражение	Описание
имя	Выбор всех узлов с указанным именем
/	Выбор относительно корневого элемента
//	Выбор узлов во всём документе
.	Выбор текущего узла
..	Выбор родителя текущего узла
@	Выбор атрибута

## Примеры выражений для выбора узлов

Выражение	Описание
bookstore	Выбор всех узлов с именем "bookstore".
/bookstore	Выбор корневого узла bookstore. <b>Внимание:</b> пути, начинающиеся с /, всегда являются абсолютными.
bookstore/book	Выбор всех узлов с именем "book", которые являются детьми узла bookstore.
//book	Выбор всех узлов с именем "book" на всех уровнях документа.
bookstore//book	Выбор всех потомков узла bookstore с именем "book".
//@lang	Выбор всех атрибутов с именем "lang".

# Предикаты

Предикаты используются для нахождения конкретного узла или узлов, содержащих требуемое значение.

Выражение	Описание
<code>/bookstore/book[1]</code>	Выбор первого узла с именем "book", являющегося ребёнком узла bookstore.
<code>/bookstore/book[last()]</code>	Выбор последнего узла с именем "book", являющегося ребёнком узла bookstore.
<code>/bookstore/book[last()-1]</code>	Выбор предпоследнего узла с именем "book", являющегося ребёнком узла bookstore.
<code>/bookstore/book[position()&lt;3]</code>	Выбор двух первых узлов с именем "book", являющихся ребёнком узла bookstore.

# Предикаты

Выражение	Описание
<code>//title[@lang]</code>	Выбор всех элементов с именем "title", имеющих атрибут lang.
<code>//title[@lang='eng']</code>	Выбор всех элементов с именем "title", имеющих атрибут lang, который равен 'eng'.
<code>/bookstore/book[price&gt;35.00]</code>	Выбор всех узлов book, являющихся детьми узла bookstore, у которых есть дочерний элемент price, значение которого больше 35.00.
<code>/bookstore/book[price&gt;35.00]/title</code>	Выбор всех элементов с именем "title", чей родительский элемент удовлетворяет условию из предыдущего примера.

# Выбор произвольных узлов

XPath позволяет использовать спецсимволы для выбора произвольных узлов.

Выражение	Описание
*	Соответствует любому элементу.
*@	Соответствует любому атрибуту.
node()	Соответствует узлу любого типа.

## Примеры

Выражение	Описание
/bookstore/*	Выбор всех дочерних элементов узла bookstore.
//*	Выбор всех элементов документа.
//title[@*]	Выбор всех элементов с именем "title", имеющих какие-либо атрибуты.

# Выбор по нескольким условиям

При помощи оператора **|** можно осуществлять выбор по нескольким условиям.

Выражение	Описание
<code>//book/title   //book/price</code>	Выбор всех элементов <code>title</code> и <code>price</code> , являющихся детьми узла <code>book</code> .
<code>//title   //price</code>	Выбор всех элементов <code>title</code> и <code>price</code> .
<code>/bookstore/book/title   //price</code>	Выбор всех узлов <code>title</code> , являющихся детьми узла <code>book</code> и предками узла <code>bookstore</code> , и всех элементов <code>price</code> .

## Оси (Axes)

Оси позволяют определить наборы узлов по отношению к текущему.

Выражение	Описание
ancestor	Соответствует всем вышестоящим узлам относительно текущего.
ancestor-or-self	Соответствует всем вышестоящим узлам или текущему узлу.
attribute	Соответствует всем атрибутам текущего узла.
child	Соответствует всем детям текущего узла.
descendant	Соответствует всем нижестоящим узлам относительно текущего.
descendant-or-self	Соответствует всем нижестоящим узлам или текущему узлу.

## Оси (Axes)

Выражение	Описание
following	Соответствует любому узлу документа, расположенному после закрывающего тега текущего узла.
following-sibling	Соответствует всем братьям, следующим за текущим узлом.
namespace	Соответствует пространствам имён текущего узла.
parent	Соответствует родителю текущего узла.
preceding	Соответствует всем узлам документа, идущим перед текущим узлом, кроме атрибутов, пространств имён и предков текущего узла.
preceding-sibling	Соответствует всем братьям, идущим передтекущим узлом.
self	Соответствует текущему узлу.



# Пути поиска

- Пути поиска могут быть абсолютными и относительными (абсолютные пути начинаются с символа /).
- Пути формируются из одного или нескольких шагов, разделённых символом / (шаг/шаг/...).
- Каждый шаг включает в себя:
  - ось (определяет иерархические отношения между искомыми узлами и текущим узлом)
  - критерий выбора узла (указывает на узел в пределах оси)
  - ноль или больше предикатов (для уточнения перечня выбранных узлов)
- В общем случае шаги поиска имеют следующий синтаксис:  
ось : критерий [предикат]

## Примеры использования осей

Выражение	Описание
<code>child::book</code>	Выбирает все узлы <code>book</code> , которые являются детьми для текущего.
<code>attribute::lang</code>	Выбирает атрибут <code>lang</code> текущего узла.
<code>child::*</code>	Выбирает все дочерние элементы текущего узла.
<code>attribute::*</code>	Выбирает все атрибуты текущего узла.
<code>child::text()</code>	Выбирает все текстовые узлы, которые являются детьми для текущего.
<code>child::node()</code>	Выбирает все узлы, которые являются детьми для текущего.

# Примеры использования осей

Выражение	Описание
<code>descendant::book</code>	Выбирает все узлы с именем "book", являющиеся потомками текущего.
<code>ancestor::book</code>	Выбирает все узлы с именем "book", являющиеся предками текущего.
<code>ancestor-or-self::book</code>	Выбирает все узлы с именем "book", являющиеся предками текущего, и сам текущий узел.
<code>child::* / child::price</code>	Выбирает все узлы с именем "price", являющиеся внуками текущего.

# Операторы

Оператор	Описание	Пример	Результат
	Объединение двух множеств узлов	//book   //cd	Набор всех узлов book и всех cd
+	Сложение	6 + 4	10
-	Вычитание	6 - 4	2
*	Умножение	6 * 4	24
div	Деление	8 div 4	2
=	Равно	price = 9.6	true, если price равно 9.6 false, если price не равно 9.6
!=	Не равно	price != 9.6	false, если price равно 9.6 true, если price не равно 9.6

# Операторы

Оператор	Описание	Пример	Результат
<	Меньше	price < 9.6	true, price = 9.5 false, price = 9.8
<=	Меньше или равно	price <= 9.6	true, price = 9.6 false, price = 9.8
>	Больше	price > 9.6	true, price = 9.9 false, price = 4.8
=>	Больше или равно	price => 9.6	true, price = 9.6 false, price = 4.8
or	Логическое ИЛИ	price=9.80 or price=9.70	true, price = 9.8 false, price = 4.8
and	Логическое И	price>9.00 and price<9.90	true, price = 9.1 false, price = 9.9
mod	Деление по модулю	5 mod 2	1

# Тестовое приложение для работы с XPath

```
1  #!/usr/bin/env python
2  import sys
3  import libxml2
4
5  def open(xml_file):
6      doc = libxml2.parseFile(xml_file)
7      ctxt = doc.xpathNewContext()
8      students = ctxt.xpathEval("//student")
9      print students
10     ctxt.xpathFreeContext()
11     doc.freeDoc()
12
13 def main(argv):
14     if len(argv) != 2:
15         sys.stderr.write("Usage: %s xml_file" % (argv[0],))
16     else:
17         open(argv[1])
18
19 if __name__ == '__main__':
20     main(sys.argv)
```

- Вывести перечень кафедр факультета КиБ
- Вывести перечень студентов, поступивших после указанного года
- Вывести перечень групп, в которых есть студент с заданной фамилией
- Определить число студентов в заданной группе
- Определить общее число студентов на заданной кафедре
- Вывести факультет, к которому относится кафедра №36
- Вывести полное имя заданной группы
- Найти факультет с наименьшим и наибольшим количеством студентов