Web технологии 3

Язык запросов к элементам XML-документа
XML Path Language (XPath)

- не XML
- ссылка на отдельную часть XML документа
- выбор по положению, относительному положению, типу, содержимому и другим критериям
- XML документ — дерево => XPath – это язык для выбора узлов и наборов узлов этого дерева
- Работает с конечным документом => нельзя отличить генерируемые конструкции
- Активное использование в xsl преобразованиях

Типы узлов

- Корневой узел (весь XML)
- Узлы элементов
- Текстовые узлы
- Узлы атрибутов
- Узлы комментариев
- Узлы инструкций обработки
- Узлы пространств имен

Корневой маршрут

- Маршрут до корневого узла "/"
- Выбор всего дерева документа
- Абсолютный маршрут поиска

Маршруты к дочерним элементам

- Простейший маршрут — одиночное имя элемента
- Выбор дочерних элементов с указанным именем
- Относительный маршрут — выбор зависит от контекстного элемента

Маршруты к атрибутам

- Выбор атрибута элемента — @имя_атрибута
- Относительный маршрут — выбор зависит от контекстного элемента

Маршруты к тексту и комментариям

- Специальные функции — text() и comment()
- любой текстовый узел или комментарий относительно контекстного элемента

Пример

```
<people>
<person born="1912" died="1954" id="p342">
<name>Алан Тьюринг</name>
<!-- Существовало ли понятие «специалист по информатике» во времена Тьюринга? -->
<profession>математик</profession>
<profession>криптограф</profession>
<homepage xlink:href="http://www.turing.org.uk/"/>
</person>
</people>
```

Маршрут "profession"
<profession>математик</profession>
<profession>криптограф</profession>

Маршрут "@xlink:href"
http://www.turing.org.uk/

Маршрут "person/comment()"
<!-- Существовало ли понятие «специалист по информатике» во времена Тьюринга? →

Маршрут "name/text()"
Алан Тьюринг

Подстановочные выражения

- Три подстановочных выражения: *, node() и @*
- Звездочка * соответствует любому узлу элемента
- Звездочка не соответствует атрибутам, текстовым узлам, комментариям или узлам инструкций обработки
- Перед звездочкой может присутствовать префикс пространства имен (svg:*)
- node() – соответствует всем узлам: элементов, текста, атрибутов, инструкций обработки, пространств имен и комментариев
- Выражение @* соответствует всем узлам атрибутов
- Может быть с префиксом (@xlink:*)

Составные выражения

- возможность выбора более одного типа элементов или атрибутов
- Объединение нескольких маршрутов (|)

  first_name|middle_initial|last_name

  @id|@xlink:type

  |@

Сложные маршруты

- Объединение простых маршрутов
- / - перемещение вниз по иерархии
- . - текущий узел
- .. - родительский узел
- // - потомки контекстного узла

  /people/person/name/first_name/text()

  person/@id

  //@id/..

Предикаты

- Выражение Xpath ссылается на произвольное число элементов
- Ограничение выборки на каждом уровне
- Пример: элементы с профессией физик

  //person[profession="физик"]
- Поиск элемента с ID = p4567

  //person[@id="p4567"]
- Операторы сравнения <, >, >=, <= и !=
- Экранирование при использовании в XML

  <xsl:apply templates select="//person[@born<=1976]"/>

- логические операторы and и or

  //person[@born<=1920 and @born>=1910]
- Выражения конвертируются в логический тип:
  - набор узлов истинен если не пуст
  - строка истинна если не пуста
  - число истинно если равно позиции элемента

  //name[middle_initial] — элементы с дочерним элементом middle_initial
- любой шаг может иметь предикат

  /people/person[@born < 1950]/name[first_name="Алан"]

Полные маршруты поиска

- До этого были сокращенные маршруты поиска
- Каждый шаг в маршруте имеет две обязательные части: ось и критерий узла, а также одну необязательную часть – предикаты
- Ось - направление перемещения от контекстного узла
- Критерий узла - какие узлы по этой оси следует включать
- предикаты - дополнительное отсеивание
- ось и критерий узла разделяются двумя двоеточиями ::

  people/person/@id

  child::people/child::person/attribute::id

Основные оси поиска

- Поддерживаются сокращенным синтаксисом
  - дочерние узлы (child) - /
  - родительские узлы (parent) - ..
  - собственная ось (self) - .
  - ось атрибутов (attribute) - @
  - ось потомков с включением контекстного узла (descendant-or-self) - //

Дополнительные оси поиска

- Ось предков (ancestor)

  Все узлы элементов, содержащие контекстный узел

- Ось следующих одноуровневых узлов (following-sibling)

  следующие за контекстным узлом и содержащиеся в том же узле родительского элемента

- Ось предыдущих одноуровневых узлов (preceding-sibling)

  предшествующие контекстному узлу и содержащиеся в том же узле родительского элемента

- Ось следующих узлов (following)

  следующие после контекстного узла

- Ось предыдущих узлов (preceding)

  предшествующие началу контекстного узла
- Ось пространств имен (namespace)

  Все пространства имен в области действия контекстного узла, объявленные либо в контекстном узле, либо в одном из его предков
- Ось потомков (descendant)

  Все потомки контекстного узла
- Ось предков, включая контекстный узел (ancestor or self)

  Все предки контекстного узла, включая сам контекстный узел

Общие выражения XPath

- Числовые значения
  - восемь байт с плавающей запятой
  - пять базовых операций +, -, *, div (деление), mod (остаток)

    @id div 10
- Строки
  - последовательность символов Unicode
  - заключаются в одинарные или двойные кавычки
  - если Xpath в XML, то учитываются ограничения XML
  - операторы сравнения = и !=

- Логические значения
  - функции XPath: true() и false()
  - Операторы сравнения =, !=, <, >, >= и <=
  - Инверсия — функция not()

Функции XPath

- Возвращают значения одного из 4-х типов
  - логическое значение
  - числовое значение
  - набор узлов
  - строка
- нестрогая типизация параметров
- преобразование данных по возможности
- функции без параметров, с одним параметром, с двумя и с неограниченным числом

Функции для наборов узлов

- информация о наборах узлов – упорядоченных коллекциях узлов XPath
- position() - положение текущего узла в контекстном списке
- last() - количество узлов в контекстном наборе
- count() - количество узлов своего аргумента

  count(//name) - количество элементов name
- id() - поиск узлов по атрибутам типа ID (разделены пробелами)
- local_name(), namespace_uri() и name() - локальное имя, URI пространства и полное имя узла

Строковые функции

- string(a) - преобразует аргумент в строку
- starts_with(a,b) — Истина если "a" начинается с "b"
- contains(a,b) — Истина если "a" содержит "b"
- substring_before(a,b) — подстрока из "a" до "b"
- substring_after(a,b) — подстрока из "a" после "b"
- substring(a,b,c) — подстрока из "a" начиная с позиции "b" и заканчивая позицией "c"
- string_length(a) - длина строки
- normalize_space(a) — нормализация и обрезка пробельных символов

Числовые функции

- number(a) — преобазование "a" в число
- round(a) - аргумент, округленный до ближайшего целого
- floor(a) - наименьшее целое, большее или равное
- ceiling(a) - максимальное целое, меньшее или равное
- sum() - суммирует значения набора узлов