Xpath
Xlink Xpointer Xquery

Sources:
http://www.brics.dk/~amoeller/XML
http://www.w3schools.com
Overlapping domains

- XQuery
- XPointer
- XLink
- XPath
- XSLT
XPath

- XPath is a syntax for defining parts of an XML document
- XPath uses path expressions to navigate in XML documents
- XPath contains a library of standard functions
- XPath is a major element in XSLT
- XPath is a W3C Standard
Terminology

• Element
• Attribute
• text,
• namespace,
• processing-instruction,
• comment,
• document (root) nodes
The most useful path expressions:

- `nodename` Selects all child nodes of the named node
- `/` Selects from the root node
- `//` Selects nodes in the document from the current node that match the selection no matter where they are
- `.` Selects the current node
- `..` Selects the parent of the current node
- `@` Selects attributes
Path wildcards can be used to select unknown XML elements.

- * Matches any element node
- @* Matches any attribute node
- node() Matches any node of any kind
Axis: a node-set relative to the current node.

<table>
<thead>
<tr>
<th>AxisName</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>ancestor</td>
<td>Selects all ancestors (parent, grandparent, etc.) of the current node</td>
</tr>
<tr>
<td>ancestor-or-self</td>
<td>Selects all ancestors (parent, grandparent, etc.) of the current node and the current node itself</td>
</tr>
<tr>
<td>attribute</td>
<td>Selects all attributes of the current node</td>
</tr>
<tr>
<td>child</td>
<td>Selects all children of the current node</td>
</tr>
<tr>
<td>descendant</td>
<td>Selects all descendants (children, grandchildren, etc.) of the current node</td>
</tr>
<tr>
<td>descendant-or-self</td>
<td>Selects all descendants (children, grandchildren, etc.) of the current node and the current node itself</td>
</tr>
<tr>
<td>following</td>
<td>Selects everything in the document after the closing tag of the current node</td>
</tr>
<tr>
<td>following-sibling</td>
<td>Selects all siblings after the current node</td>
</tr>
<tr>
<td>namespace</td>
<td>Selects all namespace nodes of the current node</td>
</tr>
<tr>
<td>parent</td>
<td>Selects the parent of the current node</td>
</tr>
<tr>
<td>preceding</td>
<td>Selects everything in the document that is before the start tag of the current node</td>
</tr>
<tr>
<td>preceding-sibling</td>
<td>Selects all siblings before the current node</td>
</tr>
<tr>
<td>self</td>
<td>Selects the current node</td>
</tr>
<tr>
<td>Operator</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>`</td>
<td>`</td>
</tr>
<tr>
<td><code>+</code></td>
<td>Addition</td>
</tr>
<tr>
<td><code>-</code></td>
<td>Subtraction</td>
</tr>
<tr>
<td><code>*</code></td>
<td>Multiplication</td>
</tr>
<tr>
<td><code>div</code></td>
<td>Division</td>
</tr>
<tr>
<td><code>=</code></td>
<td>Equal</td>
</tr>
<tr>
<td><code>!=</code></td>
<td>Not equal</td>
</tr>
<tr>
<td><code>&lt;</code></td>
<td>Less than</td>
</tr>
<tr>
<td><code>&lt;=</code></td>
<td>Less than or equal to</td>
</tr>
<tr>
<td><code>&gt;</code></td>
<td>Greater than</td>
</tr>
<tr>
<td><code>&gt;=</code></td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td><code>or</code></td>
<td>or</td>
</tr>
<tr>
<td><code>and</code></td>
<td>and</td>
</tr>
<tr>
<td><code>mod</code></td>
<td>Modulus (division remainder)</td>
</tr>
</tbody>
</table>
Xpath functions

- See
  http://www.w3schools.com/xpath/xpath_functions.asp
XQuery

XQuery engines and APIs

• XQuery API for Java

• Xquery engines
  - [http://www.sqlsummit.com/XQueryProv.htm#Engine](http://www.sqlsummit.com/XQueryProv.htm#Engine)
• XLink defines a standard way of creating hyperlinks in XML documents.

• XPointer allows the hyperlinks to point to more specific parts (fragments) in the XML document.
XPointer

URI

http://www.foo.org/bar.xml#xpointer(article/section[position()<=5])

XPointer expression

XPointer fragment identifier

* an extension of XPath which is used by XLink to locate remote link resources
* relative addressing: allows links to places with no anchors
* flexible and robust: XPointer/XPath expressions often survive changes in the target document
* can point to substrings in character data and to whole tree fragments
Xlink: the HTML link model

Construction of a hyperlink:
`<a name="here">` is placed at the destination
`<a href="URL#here">` is placed at the source
**HTML link problems**

* in HTML, links are recognized by element names (a, img, ..)
  - we want a generic XML solution
* the "semantics" of a link is defined in the HTML specification
  - we want to identify abstract semantic features, e.g. link actuation

# Limitations:
* an anchor must be placed at every link destination (problem with read-only documents)
  - we want to express relative locations (XPointer!)
* the link definition must be at the same location as the link source (outbound)
  - we want inbound and third-party links
* only individual nodes can be linked to
  - we want links to whole tree fragments
* a link always has one source and one destination
  - we want links with multiple sources and destinations
XLink linking model
<?xml version="1.0"?>

<homepages xmlns:xlink="http://www.w3.org/1999/xlink">
  <homepage xlink:type="simple"
    xlink:href="http://www.w3schools.com">Visit W3Schools</homepage>
  <homepage xlink:type="simple"
    xlink:href="http://www.w3.org">Visit W3C</homepage>
</homepages>
simple vs extended

<mylink xlink:type="simple" xlink:href="..." xlink:show="..." .../>

<mylink xlink:type="extended">
   <myresource xlink:type="resource" xlink:role="local"/>
   <myresource xlink:type="locator" xlink:role="remote" xlink:href="..."/>
   <myarc xlink:type="arc" xlink:from="local" xlink:to="remote" xlink:show="..."/>
</mylink>
Using two remote resources

<mylink xmlns:xlink="http://www.w3.org/1999/xlink" xlink:type="extended">
  <myresource xlink:type="locator" xlink:href="students.xml#Fred" xlink:label="student"/>
  <myresource xlink:type="locator" xlink:href="teachers.xml#Joe" xlink:label="teacher"/>
  <myarc xlink:type="arc" xlink:from="student" xlink:to="teacher"/>
</mylink>
Possible uses

A powerful example application of general XLinks:

Using third-party links and a smart browser, a group of people can annotate Web pages with "post-it notes" for discussion – without having write access to the pages. They simply need to agree on a set of URIs to XLink link bases defining the annotations. The smart XLink-aware browser lets them select parts of the Web pages (as XPointer ranges), comment the parts by creating XLInks to a small XHTML documents, view each other's comments, place comments on comments, and perhaps also aid in structuring the comments.
Simple with more features:

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>

<bookstore xmlns:xlink="http://www.w3.org/1999/xlink">
  <book title="Harry Potter">
      As his fifth year at Hogwarts School of Witchcraft and Wizardry approaches, 15–year–old Harry Potter is.......  
    </description>
  </book>
</bookstore>
```