

Lecture 0

Xlink

Sang Shin
Java™ Technology Evangelist
sang.shin@sun.com

**(You can use this material in any way you want,
but if you can drop me an email when you do,
that will be greatly appreciated.)**



Topics

- Simple links
- Extended links
 - ◆ Locator
 - ◆ Arc
 - ◆ Resource

XLink Overview

- Attribute-based syntax for attaching links to XML documents
- Can be
 - ◆ Simple link
 - ◆ Bi-directional link
 - ◆ Multi-directional link
- Terms
 - ◆ Connection
 - ◆ Source, Starting resource
 - ◆ Target, Ending resource

XLink Usage Examples

- Describe tables of contents or indexes
- Connect textual emanations to the text they describe
- Indicate possible paths through online courses(sang)

XLink and Namespace

- XLink is fully namespace-aware
- “xlink” as customary namespace prefix
- Standard URI
 - ◆ <http://www.w3.org/1999/xlink>

Simple Links

- One-way connection between two resources
 - ◆ Starting resource
 - ◆ Ending resource
- Starting resource
 - ◆ Must be XML element

Simple Links

- Ending resource
 - ◆ Represented by URI
 - Does not have to be URL
 - urn:isbn:12344545
 - ◆ Examples
 - XML document
 - a particular element in an XML document
 - a group of elements in an XML document
 - a span of text in an XML document
 - non-XML document: MPEG movie

Example

```
<novel xmlns:xlink="http://www.w3.org/1999/xlink"
      xlink:type="simple"
      xlink:href="ftp://archive.org/pub/etext/etext93/wizoz10.txt">
  <title>The wonderful Wizard of Oz</title>
  <author>L. Frank Baum</author>
  <year>1990</year>
</novel>
```

xlink:type

- Tells what kind of link it is
- Possible values
 - ◆ simple
 - ◆ extended
 - ◆ locator
 - ◆ arc
 - ◆ title
 - ◆ resource

xlink:href

- Identifies the resource that is linked to
- URI form
- Absolute or relative URL
- Does not have to be URL

Applications and XLink

- Most browsers of today do not support XLink
- Different applications interpret a set of XLinks **differently**

xlink:show

- Suggests the **display behavior**
- Possible values
 - ◆ new
 - ◆ replace
 - ◆ embed
 - ◆ other
 - ◆ none

xlink:actuate

- Suggests link **actuation behavior**
- Possible values
 - ◆ onLoad
 - ◆ onRequest
 - ◆ other
 - ◆ none

Example

```
<novel xlink:type="simple"
  xlink:href="ftp://archive.org/pub/etext/etext93/wizoz10.txt"
  xlink:actuate="onRequest" xlink:show="replace">
  <title>The wonderful Wizard of Oz</title>
  <author>L. Frank Baum</author>
  <year>1990</year>
</novel>
```

- Suggestion to the browser: Display the contents of the URI in a new window when a user clicks on the link

Example

```
<image xlink:type="simple"  
  xlink:actuate="onLoad" xlink:show="embed"  
  xlink:href="http://www.turing.org.uk/turing/pi1/bus.jpg"  
  width="152" height="345" />
```

- Suggestion to the browser: Embed the image within the current page as soon as the current page is loaded

xlink:title and xlink:role

- Specifies **meaning of connection** between resources
- *xlink:title*
 - ◆ Short text description of remote resource
- *xlink:role*
 - ◆ Contains URI pointing to a longer description of the remote resource

Example

```
<novel xlink:type="simple"
  xlink:href="ftp://archive.org/pub/etext/etext93/wizoz10.txt"
  xlink:actuate="onRequest" xlink:show="replace"
  xlink:title="The complete text of the novel"
  xlink:role="http://promo.net/pg/" >
  <title>The wonderful Wizard of Oz</title>
  <author>L. Frank Baum</author>
  <year>1990</year>
</novel>
```

Extended Links

- *xlink:type="extended"*
- Collection of resources and paths between those resources
- Each path connects exactly two resources
- Directed, labeled graph
 - ◆ paths are arcs
 - ◆ documents are vertices
 - ◆ labels are URIs

Extended Links

- No obvious analogy exists
 - ◆ How application treat them
 - ◆ How to present them in UI
- Contain
 - ◆ Local resources
 - Represented by a *resource*
 - ◆ Remote resources
 - Represented by a *locator*
 - ◆ Arcs between resources

Locators

- Represents remote resource
- *xlink:type="locator"* attribute
- Accompanying attributes
 - ◆ *xlink:label*
 - Serves as element's ID
 - Does not have to be unique
 - ◆ *xlink:title*
 - Short description of the remote resource
 - ◆ *xlink:role*
 - URI pointing to longer description

Example

```
<novel xlink:type="extended">
  <title>The wonderful Wizard of Oz</title>
  <author>L. Frank Baum</author>
  <year>1990</year>
  <edition xlink:type="locator" xlink:href="urn:isbn:123"
           xlink:label="ISBN123"
           xlink:title="William Morrow"
           xlink:role="http://www.xxx.com/" >
  <edition xlink:type="locator" xlink:href="urn:isbn:456"
           xlink:label="ISBN456"
           xlink:title="Oxford"
           xlink:role="http://www.osf.com/">
</novel>
```

Arcs

- Paths between resources
 - ◆ Among local and remote resources
- *xlink:xtype="arc"* attribute
- Accompanying attributes
 - ◆ *xlink:from="<label>"* - link's source
 - ◆ *xlink:to="<label>"* - link's target
 - ◆ *xlink:title*
 - ◆ *xlink:role*
- Application behavior not defined



Example

```
<series xlink:type="extended" xmlns:xlink="http://www.w3.org/1999/xlink">
  <author>L. Frank Baum</author>

  <!-- locator elements -->
  <novel xlink:type="locator" xlink:label="oz1"
        xlink:href="ftp://archive.org/pub/etext/etext93/wizoz10.txt">
    <title>The Wonderful Wizard of Oz</title>
    <year>1900</year>
  </novel>
  <novel xlink:type="locator" xlink:label="oz2"
        xlink:href="ftp://archive.org/pub/etext/etext93/ozland10.txt">
    <title>The Marvelous Land of Oz</title>
    <year>1904</year>
  </novel>
  <novel xlink:type="locator" xlink:label="oz3"
        xlink:href="ftp://archive.org/pub/etext/etext93/wizoz10.txt">
    <title>Ozma of Oz</title>
    <year>1907</year>
  </novel>

  <!-- arcs -->
  <next    xlink:type="arc" xlink:from="oz1" xlink:to="oz2" />
  <next    xlink:type="arc" xlink:from="oz2" xlink:to="oz3" />
  <previous xlink:type="arc" xlink:from="oz2" xlink:to="oz1" />
  <previous xlink:type="arc" xlink:from="oz3" xlink:to="oz2" />

</series>
```

Multiple Arcs

- Multiple elements share same label
- If *xlink:to* is omitted
 - ◆ All resources in the extended link are target resources
- If *xlink:from* is omitted
 - ◆ All resources in the extended link are source resources

Example

```
<book xlink:type="extended" xmlns:xlink="http://www.w3.org/1999/xlink">

  <author>L. Frank Baum</author>
  <title>The Wonderful Wizard of Oz</title>

  <!-- locator elements -->
  <edition xlink:type="locator" xlink:href="urn:isbn:0192839306"
    xlink:title="Oxford University Press"
    xlink:role="http://www.oup-usa.org/"
    xlink:label="ISBN0192839306"/>

  <store xlink:type="locator"
    xlink:href="http://www.amazon.com/exec/obidos/ASIN/0192839306"
    xlink:label="buy">Amazon</store>

  <store xlink:type="locator" xlink:href=
    "http://www1.fatbrain.com/asp/bookinfo/bookinfo.asp?theisbn=0192839306"
    xlink:label="buy">Fatbrain</store>

  <store xlink:type="locator" xlink:href=
    "http://shop.bn.com/booksearch/isbninquiry.asp?isbn=0192839306"
    xlink:label="buy">Barnes & Noble</store>

  <!-- arcs -->
  <purchase xlink:type="arc" xlink:from="ISBN0192839306" xlink:to="buy" />

</book>
```

Local Resources

- Represents data in the XML document
- *xlink:type="resource"* attribute
- Accompanying attributes
 - ◆ *xlink:label*
 - ◆ *xlink:role*
 - ◆ *xlink:title*

Summary

- Simple links
- Extended links
 - ◆ Locator
 - ◆ Arc
 - ◆ Resource