XML Namespaces

Lecture 4, 6/30/03
Overview

In this lecture, we will cover:

- Problems with basic X ML
- Solutions to this problem
An Example

<artistic-work>
  <book>
    <title>The Hitchhiker’s Guide to the Galaxy</title>
    <author>Douglas Adams</author>
  </book>
</artistic-work>
What is the problem with this document?

```html
<html>
<head>
  <title>Favorite Books</title>
</head>
<body>
  <artistic-work>
    <book>
      <title>The Hitchhiker's Guide to the Galaxy</title>
      <author>Douglas Adams</author>
    </book>
  </artistic-work>
</body>
</html>
```
How can we solve this problem?

- Introduce ‘scoping’ rules for element name.
- W3C came up with ‘Namespaces’
- Think of Java classes and fully qualified names: `System.out.println("hello");`
What does one look like?

- Namespace qualified names look just like regular names, but are prefixed with a \texttt{xmlns}

\begin{verbatim}
<art:title>The Hitchhiker’s Guide to the Galaxy</art:title>
<html:title>Favorite Books</html:title>
\end{verbatim}
Problems with prefixes?

- Are there any problems with using prefixes? What are they?
- Anyone can create a prefix, so we still have the same problem we had before.
Solutions: URLs and URNs

- We must specify a URL or URN that uniquely identifies a namespace prefix.

  `<artistic-work
    xmlns:art="http://ww.cs.uoregon.edu/classes/cis399xml/art">`

- The ‘art’ prefix will be recognized for elements at this level and below.
Namespace syntax

- Namespace declarations are case sensitive, just like elements and attributes.

\[
<\text{art:artistic-work xmlns:art="art.org"}>
<\text{art:artistic-work xmlns:art="ART.ORG"}>
\text{are different!}
\]
Default Namespace

- We can also set a default namespace with the following:

```xml
<artistic-work xmlns="art.org/ns">
</artistic-work>
```
What about attributes?

- Do we want namespaces for attributes?

- We can declare attribute name spaces just the same!

```xml
<art xmlns:historic="http://history.org/ns">
  <painting historic:old="true">Birth of Venus</painting>
</art>
```
Namespaces and DTDs

- DTDs were designed before namespaces (SGML)
- You can specify the namespace attribute for default namespaces like this:

```xml
<!ATTLIST artistic-work xmlns CDATA #FIXED "http://art.org/ns">
```
DTDs and Namespaces cont...

- Since the colon is part of a valid XML name, you can use namespaces in your DTD directly

```xml
<!ELEMENT art:artistic-work ((book | painting)*)>
```
Problems with DTDs

- However, namespaces are interpreted on the fly
- Validating parsers will match namespaces with URIs.

<artistic-work xmlns:art="http://art.org/ns">
<artistic-work xmlns:aw="http://art.org/ns">
are the same, but not according to a DTD!
Exercises

- Create a checkbook xml document
- Use the namespace “check” for checks with the URL “http://www.bankofamerica.com”
- Use the namespace “deposits” for deposits, with the URL “http://www.oregoncommunitycu.org”