Grid Computing

XML

Paul A. Farrell
Fall 2006

Including material from Amy Apon, James McCartney, Arkansas U.

International Summer School on Grid Computing 2006
Ischia (Naples) July 9-21, 2006

What is XML

- XML stands for extensible markup language
- It is a hierarchical data description language
- It is a sub set of SGML a general document markup language
- XML can define both presentation style and give information about content.
- XML relies on custom documents defining the meaning of tags.
- Differences from HTML
  - HTML is a presentation markup language – provides no information about content.
  - There is only one standard definition of all of the tags used in HTML.

Tags, elements, and attributes.

```
<address>
  <name>
    <title>Mrs.</title>
    <first-name>
      Mary
    </first-name>
    <last-name>
      McGoon
    </last-name>
  </name>
  <street>
    1401 Main Street
  </street>
  <city state="NC" >Anytown</city>
  <postal-code>
    34829
  </postal-code>
</address>
```

A. A tag is a value between the less than (<) and greater than (>) angle brackets.

B. An element includes the starting and ending tag, and everything between the two. This includes other (child) elements.

C. An attribute is a name value pair that is located in the opening tag of an element

Valid and well formed XML

- A correct XML document must be both valid and well formed.
- Well formed means that the syntax must be correct and all tags must close correctly (eg <…> </…>).
- Valid means that the document must conform to some XML definition (a DTD or Schema).

What is a Schema?

- A schema is the definition of the meaning of each of the tags within a XML document.
- *Analogy: A HTML style sheet can be seen as a limited schema which only specifies the presentational style of HTML which refers to it.*
  - *Example: in HTML the tag pre-defined. In XML you would need to define this in the context of your document.*
- A schema can 'inherit' from another and extend it. (*analogous to extending a class in JAVA*)
- For example the basic tags which allow you to write schema are defined in :
  http://www.w3.org/2001/XMLSchema

Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs=http://www.w3.org/2001/XMLSchema
xmlns="document" >
<xs:element name = "DOCUMENT">
        <xs:element name="CUSTOMER"> </xs:element>
</xs:element>
</xs:schema>
```

Simple schema saved as order.xsd

```
<?xml version="1.0"?>
<DOCUMENT xmlns="document"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
Xsi:schemaLocation="order.xsd">
<DOCUMENT>
        <CUSTOMER>sam smith</CUSTOMER>
        <CUSTOMER>sam smith</CUSTOMER>
</DOCUMENT>
```

XML document derived from schema.

XML Structure

- General Structure of an XML Document:

XML Rules

- An XML document must contain a single root element.
- Elements can't overlap (jump levels).
- End tags are required (at least <tag />).
- Element names are case sensitive.
- Attributes must have a value and the values must be quoted.

Parts and Pieces

- Declaration – a parser flag.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
```

- Comments – between <!-- and -->
- Processing Instructions –

```
<?cocoon-process type="sql"?>
```

- Entities – begin with an ampersand, end in semi-colon.

```
<!ENTITY myent "My Custom Entity"!>
&lt;    &gt;    &amp;    &quot;    &apos;
```


Namespaces in XML

- Schema require namespaces.
- A namespace is the domain of possible names for an entity within a document.
- Normally a single namespace is defined for a document. In this case fully qualified names are not required.

Common namespace prefixes

xsi	http://www.w3c.org/2000/10/XMLSchema_instance *namespace governing XMLSchema instances*
xsd	http://www.w3c.org/2000/10/XMLSchema *namespace of schema governing XMLSchema (.xsd) files*
tns	by convention this refers to "this" document *refers to the current XML document*
wsdl	http://schemas.xmlsoap.org/wsdl/ *WSDL namespace*
soap	http://schema.xmlsoap.org/wsdl/soap/ *WSDL SOAP binding namespace*

Using namespaces in XML

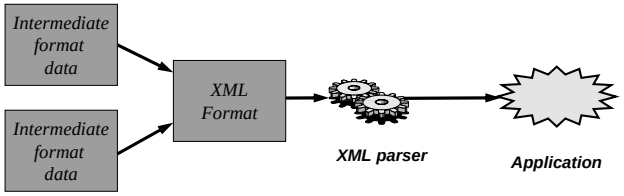
- To fully qualify a namespace in XML write the namespace:tag name. eg.
  <my_namespace:tag> </my_namespace:tag>
- In a globally declared single namespace the qualifier may be omitted.
- More than one namespace:
  <my_namespace:tag> </my_namespace:tag>
  <your_namespace:tag> </your_namespace:tag>
  can co-exist if correctly qualified.

Why namespaces in XML?

- A namespace is used to ensure that a tag (variable) has a unique name and can be referred to unambiguously.
- Namespaces protect variables from being inappropriately accessed – encapsulation.
- This makes sure that when you access a variable correctly it has the expected value.

Using XML for Data Exchange

- XML can help, because its (standard) notation can be analysed by off the shelf XML parsers



Using XML as Metadata

- XML metadata provides information about the structure and meaning of any data
- XML metadata can be used to perform more intelligent web searches for goods or information
- Cross-site searches are difficult (depends on metadata info in pages)
- XML metadata is more self-describing and meaningful, for example ...
  - Search for all plays written by William Shakespeare
  - Rather than every web site that mentions him!

Using XML for Remote Procedure Calls

- XML used to exchange data between Software Components
- Simple Object Access Protocol – SOAP
  - A lightweight protocol for exchange of information in a decentralised, distributed environment
  - Web-Sites expose interfaces for interrogation
- Universal Description, Discovery and Integration – UDDI
  - Integrating business services
  - 'Yellow/White Pages'

References

- http://www.xml.com – O'Reilly Network
- http://www.xml.org - OASIS
- http://www.106.ibm.com/developerworks/edu/x_dw_xmlintro_i.html IBM tutorial on XML, Doug Tidwell
- XML, The Complete Reference. Williamson, 2001 McGraw-Hill

Summary

- XML is a language that provides
  - A mark upspecification for creating self descriptive data
  - A platform and application independent data format
  - A way to perform validation on the structure of data
  - A syntax that can be understood by computers and humans
  - The way to advance web applications used for electronic commerce.