

Validating textual values

Act **Act 1** **Scene 5** Scene

Another part of the platform.

Enter GHOST and HAMLET

HAMLET

Where wilt thou lead me? speak; I'll go no further.

GHOST

Mark me.

HAMLET

I will.

Speech

29/10/2001

AIJA & VSCL Joint Conference

7

Example XML

Attribute

<Transcript Date="21 June '94">

<Front/>

<Body>

Name Value

<Witness Name="Jill Shively">

Element

<Examination Name="Ms. Clark">

<Question>Miss Shively, directing your attention to the date of June 12, 1994...</Question>

Text

<Answer>Yes</Answer>

</Examination>

</Witness></Body></Transcript>

29/10/2001

AIJA & VSCL Joint Conference

8

What you can specify with a DTD

- Element and attribute names you will allow
- Which attributes are allowed in which elements
- Which elements are allowed in which other elements
- Restrictions on the values taken by attributes
- Where text may occur

29/10/2001

AIJA & VSCL Joint Conference

9

Preparing to write a DTD for transcripts

- Key question: What conventions are used?
 - All transcripts have front matter, a body, and end matter
 - All transcript bodies contain a sequence of witnesses
 - Each witness is examined, cross-examined, re-examined, etc
 - Each examination involves a series of question and answer pairs

29/10/2001

AIJA & VSCL Joint Conference

10

What elements do I want?

- Transcript
- Front
- Body
- End
- Witness
- Examination
- Question
- Answer

29/10/2001

AIJA & VSCL Joint Conference

11

Specifying elements in a DTD

<!ELEMENT Transcript (Front, Body, End?)>

<!ELEMENT Front (#PCDATA)>

<!ELEMENT Body (Witness+)>

<!ELEMENT Witness (Examination+)>

<!ELEMENT Examination (Question, Answer)+>

<!ELEMENT Question (#PCDATA)>

#PCDATA = Parsed Character DATA
i.e. strings, not markup

29/10/2001

AIJA & VSCL Joint Conference

12

Validating XML against DTD

```

<!ELEMENT Witness
  (Examination+)>
<!ELEMENT Examination
  (Question, Answer)+>
<!ELEMENT Question
  (#PCDATA)>
<Witness Name="Jill
  Shively">
  <Examination Name="Ms.
  Clark">
    <Question>Miss Shively,
    directing your attention
    to the date of June 12,
    1994...</Question>
    <Answer>Yes</Answer>
  </Examination>
</Witness>

```

29/10/2001

AIJA & VSCL Joint Conference

13

What attributes do I want?

- Transcripts have dates
- Transcripts have locations
- Witnesses have names
- Examiners have names
- Questions and answers have times

29/10/2001

AIJA & VSCL Joint Conference

14

Specifying attributes in a DTD

```

<!ATTLIST Transcript
  Date CDATA #REQUIRED
  Location CDATA #IMPLIED>
<!ATTLIST Witness
  Name CDATA #REQUIRED>

```

CDATA = Character DATA
i.e. strings, not markup

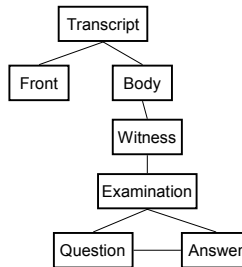
29/10/2001

AIJA & VSCL Joint Conference

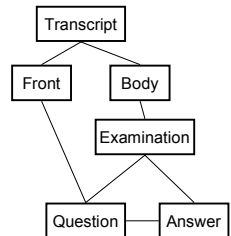
15

DTD limitation: Tree shape

- Right



- Wrong



29/10/2001

AIJA & VSCL Joint Conference

16

Alternative trees

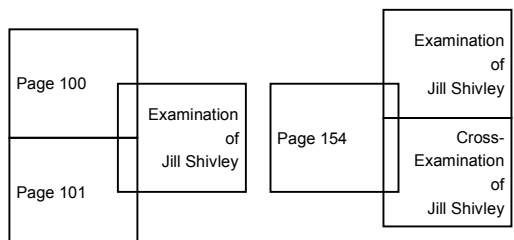
- Transcript
 - Front
 - Body
 - Examination
 - Witness
 - » Question
 - » Answer
 - » Question
 - » Answer
 - Witness
 - ...
 - End
- Transcript
 - Front
 - Body
 - Page 4291
 - Line 1
 - Line 2
 - Line 3
 - ...
 - Page 4292
 - Line 1
 - ...
 - End

29/10/2001

AIJA & VSCL Joint Conference

17

Conflicting trees



- Page cannot go inside Examination
- Examination cannot go inside page

29/10/2001

AIJA & VSCL Joint Conference

18

Why XML Schemas instead of DTDs

- XML Schemas offer
 - More detailed specification of types
 - DTD offers strings, ID references, only
 - Better cardinality control
 - An XML Schemas namespace
- XML Schemas are written in XML

“Over time, many people have complained to the W3C about the complexity of DTDs and have asked for something simpler.

“W3C listened, assigned a committee to work on the problem, and came up with a solution that is much more complex than DTDs ever were.”

Steven Holzner, Inside XML.

Example schema

```
<element name="Transcript">
  <complexType>
    <sequence>
      <element ref="In:Front" minOccurs="1"
                maxOccurs="1"/>
      <element ref="In:Body"/>
    </sequence>
    <attribute name="Date" type="date"/>
    <attribute name="Location" type="In:locationType"/>
  </complexType>
</element>
```

Example: Custom types in Schema

```
<simpleType name="locationType">
  <restriction base="string">
    <enumeration value="Supreme Court">
    <enumeration value="Bankruptcy Court">
    <enumeration value="Circuit Court">
  </restriction>
</simpleType>
```

Built-in data types for Schemas

- boolean e.g. true, false
- decimal e.g. 1.23
- positiveinteger e.g. 1000
- date e.g. 2001-10-26
- anyURI e.g. http://www.livenote.com
- int, short, byte
- ...plus 36 others

Conclusions

- We need DTDs or Schemas (or something)
 - They are the foundation for data exchange, e-commerce, etc especially for 'loosely connected' partners
- XML Schemas are stable enough to use
 - They reached Recommendation status in May
 - Reasonable tools are emerging
- The current pressing problem: the politics of schema making
 - *“Schemas reflect the institutions that produce them.”*
Kendall Grant Clark [3]

Further reading

1. Learning Schemas
 - *XML Schema Tutorial*, and *Best Practices*
 - Roger L. Costello
 - www.xFront.com
2. The official documentation
 - *XML Schema Part 0: Primer*
 - World Wide Web Consortium
 - www.w3.org/TR/xmlschema-0
3. Political implications of schemas
 - *The Politics of Schemas*
 - Kendall Grant Clark
 - www.xml.com/pub/a/2001/01/31/politics.html
4. Extensive reference of Schema (and XML) news
 - *The XML Cover Pages*
 - Robin Cover
 - xml.coverpages.org/schemas.html