XML and Semantic Web Technologies

II. XML / 3. XML Namespaces

Lars Schmidt-Thieme

Information Systems and Machine Learning Lab (ISMLL)
Institute of Economics and Information Systems
& Institute of Computer Science
University of Hildesheim
http://www.ismll.uni-hildesheim.de

DTD Integration / Example (1/9)

```
1 <?xml version="1.1"?>
2 <!DOCTYPE article SYSTEM "article.dtd">
3 <article>
4   <title>What others say</title>
5   A short overview of basic and most important XML technologies
6   is given in ...
7
8   Also useful is ...
9 </article>
```

Figure 1: Sample article document.

```
1 <!ELEMENT article (#PCDATA | strong | em | title)* >
2 <!ELEMENT strong (#PCDATA)>
3 <!ELEMENT em (#PCDATA)>
4 <!ELEMENT title (#PCDATA | strong | em)* >
```

Figure 2: Sample article DTD `article-separat.dtd`.

DTD Integration / Example (2/9)

```
1 <?xml version="1.1"?>
2 <!DOCTYPE book SYSTEM "books.dtd">
3 <book>
4   <author><fn>Erik T.</fn><sn>Ray</sn></author>
5   <title>Learning XML</title>
6   <year edition="2">2003</year>
7 </book>
```

Figure 3: Sample book document.

```
1 <!ELEMENT books (book*)>
2 <!ELEMENT book (author+, title, year)>
3 <!ELEMENT author (fn, sn)>
4 <!ELEMENT fn (#PCDATA)>
5 <!ELEMENT sn (#PCDATA)>
6 <!ELEMENT title (#PCDATA)>
7 <!ELEMENT year (#PCDATA)>
8 <!ATTLIST year
9   edition CDATA #IMPLIED>
```

Figure 4: Sample book DTD `books.dtd`.

DTD Integration / Example (3/9)

```
1 <?xml version="1.1"?>
2 <!DOCTYPE article SYSTEM "article.dtd" [
3   <!ENTITY % books SYSTEM "books.dtd">
4   %books;
5 ]>
6 <article>
7   <title>What others say</title>
8   A short overview of basic and most important XML technologies
9   is given in
10  <book>
11    <author><fn>Erik T.</fn><sn>Ray</sn></author>
12    <title>Learning XML</title>
13    <year edition="2">2003</year>
14  </book>
15  Also useful is ...
16 </article>
```

Figure 5: Combined article and book document.

DTD Integration / Example (4/9)

```
1 <?xml version="1.1"?>
2 <!DOCTYPE article SYSTEM "article-any-atitle.dtd" [
3   <!ENTITY % books SYSTEM "books.dtd">
4   %books;
5 ]>
6 <article>
7   <articletitle>What others say</articletitle>
```

Figure 6: Combined article and book document (first 7 lines).

```
1 <!ELEMENT article ANY >
2 <!ELEMENT strong (#PCDATA)>
3 <!ELEMENT em (#PCDATA)>
4 <!ELEMENT articletitle (#PCDATA | strong | em)* >
```

Figure 7: Modified article DTD `article-any-atitle.dtd`.

DTD Integration / Example (5/9)

This is a cumbersome approach as

- original DTDs have to be modified: element names (booktitle) and element content models (article),
- documents have to be partly recoded (title → booktitle),
- every position where elements from another DTD should be allowed,
  - either the content model has to be changed to `ANY`
    (which is much too lax)
  - or the specific elements of the other DTD have to be included
    (which affords customization for each DTD to allow elements from).

What we want to say is

- "<title>What others say</title>" (line 7) belongs to article DTD,
- "<title>Learning XML</title>" (line 12) belongs to book DTD,

i.e., attach DTDs to element names.

XML Namespaces

For "mixing vocabularies" XML Namespaces have been designed. They provide mechanisms for

- marking elements and attributes with namespaces and
- validating documents with elements and attributes from different namespaces (mostly in conjunction with XML Schema)

version: Namespaces in XML 1.1 (W3C Recommendation, 2nd ed., 2006/08/16)

A namespace is identified by an (absolute) IRI reference.

Expanded name: pair of

- namespace IRI (**namespace name**) and
- **local name**.

Declaration of Namespace Prefixes

Namespace attribute to declare namespace prefixes:

⟨*NamespaceAtt*⟩ := (`xmlns` | `xmlns :` ⟨*NCName*⟩) `=` `"` ⟨*IRI*⟩ `"`

⟨*NCName*⟩ = non-colonized name (i.e., without ":"s).

Scope: element it is attribute of.

Without prefix defines **default namespace**.

Implicitly declared prefixes:

- xml: `http://www.w3.org/XML/1998/namespace`
- xmlns: `http://www.w3.org/2000/xmlns/`

Namespace Usage

Qualified name (⟨*QName*⟩): name subject to namespace interpretation (maybe prefixed, maybe unprefixed).

⟨*QName*⟩ := NCName | (⟨*NamespacePrefix*⟩ : ⟨*NCName*⟩)

A prefix associates the name of an element or attribute with a namespace.

Default namespace applies

- to the element it is attribute of (if it is unprefixed) and
- to all nested elements (unless they are prefixed or the default namespace is overwritten).
- but not to unprefixed attributes.

DTD Integration / Example (6/9)

```
1 <?xml version="1.1"?>
2 <article xmlns="http://www.cgnm.de/xml/article.dtd"
3         xmlns:bk="http://www.cgnm.de/xml/books.dtd">
4   <title>What others say</title>
5   A short overview of basic and most important XML technologies
6   is given in
7   <bk:book>
8     <bk:author><bk:fn>Erik T.</bk:fn><bk:sn>Ray</bk:sn></bk:author>
9     <bk:title>Learning XML</bk:title>
10    <bk:year edition="2">2003</bk:year>
11  </bk:book>
12  Also useful is ...
13 </article>
```

Figure 8: Namespaces are used to differentiate elements from different DTDs (default namespace and prefix).

DTD Integration / Example (7/9)

```
<?xml version="1.1"?>
<article xmlns="http://www.cgnm.de/xml/article.dtd">
  <title>What others say</title>
  A short overview of basic and most important XML technologies
  is given in
  <book xmlns="http://www.cgnm.de/xml/books.dtd">
    <author><fn>Erik T.</fn><sn>Ray</sn></author>
    <title>Learning XML</title>
    <year edition="2">2003</year>
  </book>
  Also useful is ...
</article>
```

Figure 9: Namespaces are used to differentiate elements from different DTDs (overwritten default namespace).

DTD Integration / Example (8/9)

```
<?xml version="1.1"?>
<!DOCTYPE article SYSTEM "article-any.dtd" [
  <!ENTITY % books SYSTEM "books-bk.dtd">
  %books;
]>
<article xmlns="http://www.cgnm.de/xml/article.dtd"
         xmlns:bk="http://www.cgnm.de/xml/books.dtd">
  <title>What others say</title>
  A short overview of basic and most important XML technologies
  is given in
  <bk:book>
    <bk:author><bk:fn>Erik T.</bk:fn><bk:sn>Ray</bk:sn></bk:author>
    <bk:title>Learning XML</bk:title>
    <bk:year edition="2">2003</bk:year>
  </bk:book>
  Also useful is ...
</article>
```

Figure 10: Combined article and book document.

DTD Integration / Example (9/9)

```
<!ELEMENT article ANY >
<!ATTLIST article
   xmlns    CDATA #IMPLIED
   xmlns:bk CDATA #IMPLIED>
<!ELEMENT strong (#PCDATA)>
<!ELEMENT em (#PCDATA)>
<!ELEMENT title (#PCDATA | strong | em)* >
```

Figure 11: Modified article DTD `article-any.dtd` for use of pseudo-namespaces in DTDs.

```
<!ELEMENT bk:books (bk:book*)>
<!ELEMENT bk:book (bk:author+, bk:title, bk:year)>
<!ELEMENT bk:author (bk:fn, bk:sn)>
<!ELEMENT bk:fn (#PCDATA)>
<!ELEMENT bk:sn (#PCDATA)>
<!ELEMENT bk:title (#PCDATA)>
<!ELEMENT bk:year (#PCDATA)>
<!ATTLIST bk:year
  edition CDATA #IMPLIED>
```

Figure 12: Modified books DTD `books-bk.dtd` for use of pseudo-namespaces in DTDs.

Summary

- XML namespaces are used to distinguish names of elements and attributes from different schemata / DTDs.
- XML namespaces are declared using the attribute xmlns:⟨prefix⟩.
- XML namespaces declarations are valid starting from the element they are declared in.
- Elements and attributes are assigned to a namespace
  - either by using qualified names ⟨prefix⟩:⟨localname⟩ or
  - the default namespace declared by the attribute xmlns.
- DTDs are not namespace-aware.