Advanced Databases

XML : eXtensible Markup Language

Prof. dr. J. Paredaens
mmv. M. Mampaey
TU/e

Table of Contents

References

I. XML

[1] www.w3.org/TR/xmlschema-0

[2] www.w3.org/TR/xmlschema-1

[3] www.w3.org/TR/xpath

[4] P.M. Lewis, A. Bernstein, M. Kifer, Databases and Transaction Processing, Addison Wesley, Chapter 17, 2002

[5] D. Chamberlin, XQuery, a query language for XML, Web, June 2003

[6] www.w3.org/TR/xquery

[7] www.w3.org/XML/Query

References

[8] www.w3.org/TR/xslt20

[9] Draper et al, XQuery 1.0 and XPath 2.0 Formal Semantics, www.w3.org/TR/xquery-semantics/, 2003

[10] M. Brundage, XQuery, Add. Wesley, 2004

[11] J. McGovern, P. Bothner, K. Cagle, J. Linn, V. Nagarajan, XQuery, Sams Publ., 2004

[12] H. Katz, XQuery from the Experts, Add. Wesley, 2004

1. Introduction to XML [4]

- Web data for human consumption
  - HTML (Cfr. next slide)
  - Self describing: attribute names are included
    - but not explicitly separated from data values
- Web data for machine consumption
  - Characteristics of Semistructured Data:
    - object-like
    - schemaless
    - self-describing
  - XML
    - (optional) structure descr.: DTD, XML Schema

HTML-document

```
<html>
  <head><Title>Student List</Title></head>
  <body>
    <h1>ListName: Students</h1>
    <dl>
      <dt>Name: Jan Vijs
        <dd>Id: 11
        <dd>Address:
          <ul>
          <li>Number: 123
          <li>Street: Turnstreet
          </ul>
      <dt>Name: Jan De Moor
        <dd>Id: 66
        <dd>Address:
          <ul>
          <li>Number: 4
          <li>Street: Hole Rd
          </ul>
    </dl>
  </body>
</html>
```

- Why is XML important?
  - simple open non-proprietary widely accepted data exchange format
- XML is like HTML but
  - no fixed set of tags
    - X = “extensible”
  - no fixed semantics (c.q. representation) of tags
    - representation determined by separate ‘stylesheet’
    - semantics determined by application
  - no fixed structure
    - user-defined schemas

XML-document – Running example 1

```
<?xml version ="1.0"?>
<PersonList Type="Student" Date="2004-12-12">
  <Title Value="Student List"/>
  <Contents>
    <Person>
      <Name>Jan Vijs</Name>
      <Id>11</Id>
      <Address>
        <Number>123</Number>
        <Street>Turnstreet</Street>
      </Address>
    </Person>
    <Person>
      <Id>66</Id>
      <Address>
        <Number>4</Number>
        <Street>Hole Rd</Street>
      </Address>
    </Person>
  </Contents>
</PersonList>
```

- Global structure
  - First line is mandatory;
  - Tags are chosen by author;
  - Opening tag must have a matching closing tag;
    `<a><b></b><c></c></a>`
  - Only one root element `PersonList`;
  - `<a> … </a>`; **a** is the name of the element, content, child, descendant, parent, ancestor, sibling;
  - `<PersonList Type="Student">` **Type** is name of the attribute of element **PersonList**; the value of the attribute is **"Student"** ; all attribute values must be quoted;

  - empty elements:
    ```
    <Title Value="Student List"> </Title>
    <Title Value="Student List"/>
    ```
  - processing instruction: `<? . . . ?>`
  - comment: `<!-- here we go -->`
  - mixed data-text:

```
<Address>
  Jan lives in <Street> Q Street </Street> number
  <Number>123</Number>
</Address>
```

  - elements are ordered:

```
<Address>                    <Address>
  <Number>123</Number>         <Street> Q Street </Street>
  <Street> Q Street </Street>  <Number>123</Number>
</Address>                   </Address>
```

    are different
  - weak facilities for constraints