

# Accessing, querying and transforming XML documents

Πληροφοριακά Συστήματα Διαδικτύου

# Addressing and Querying XML Documents

- In relational databases, parts of a database can be selected and retrieved using SQL
  - Same necessary for XML documents
  - **Query languages**: XQuery, XQL, XML-QL
- The central concept of XML query languages is a **path expression**
  - Specifies how a node or a set of nodes, in the tree representation of the XML document can be reached

# XPath

- XPath is core for XML query languages
- Language for addressing parts of an XML document.
  - It operates on the tree data model of XML
  - It has a non-XML syntax

# Types of Path Expressions

- **Absolute** (starting at the root of the tree)
  - Syntactically they begin with the symbol /
  - It refers to the root of the document (situated one level above the root element of the document)
- **Relative** to a context node

# Example of Path Expression in XPath

- Address all **author** elements. (Here **//** says that we should consider all elements in the document and check whether they are of type **author**)

**//author**

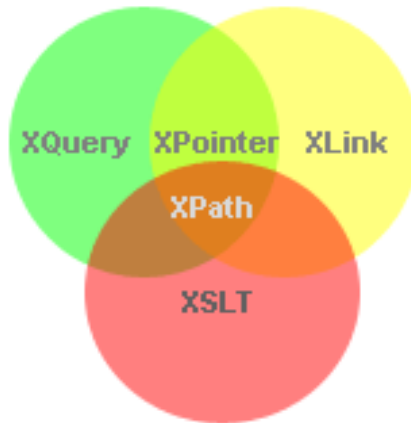
- Address all **author** elements. (Addresses all **author** elements that are children of the **library** element node, which resides immediately below the root)

**//library/author**

- Address all **title** attribute nodes within **book** elements anywhere in the document, which have the value “Artificial Intelligence”

**//book/@title=“Artificial Intelligence”**

# XQuery



- XQuery is **the** language for querying XML data
- XQuery is to XML what SQL is for databases
- XQuery is built on XPath expressions
- XQuery is a W3C Recommendation

# XQuery Example

- for \$x in doc("books.xml")/bookstore/book  
where \$x/price>30  
order by \$x/title  
return \$x/title

# Displaying XML Documents

- It is possible to use **CSS** (Cascading Style Sheets) to format (add display information) to an XML document.
- However, formatting XML with CSS is not the most common method.
- **XSLT** (eXtensible Stylesheet Language Transformations) is the recommended style sheet language of XML and is far more sophisticated than CSS.
- XSLT can be used to transform XML into HTML, before it is displayed by a browser



# XSL Transformations (XSLT)

- XSLT specifies rules with which an input XML document is transformed to
  - another XML document
  - an HTML document
  - plain text
- The output document may use the same DTD or schema, or when applications that use different DTDs or schemas need to communicate
- XSLT can be used independently of the formatting language
- Move data and metadata from one XML representation to another

# XSLT Transformation into HTML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/author">
    <html>
      <head><title>An author</title></head>
      <body bgcolor="white">
        <b><xsl:value-of select="name"/></b><br />
        <xsl:value-of select="affiliation"/><br />
        <i><xsl:value-of select="email"/></i>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```

## Example input:

```
<author>
  <name>Marios Pitikakis</name>
  <affiliation>University of Thessaly</affiliation>
  <email>mpitikak@inf.uth.gr</email>
</author>
```

## Example output:

```
<html>
  <head><title>An author</title></head>
  <body bgcolor="white">
    <b>Marios Pitikakis</b><br>
    University of Thessaly<br>
    <i>mpitikak@inf.uth.gr</i>
  </body>
</html>
```

# Observations About XSLT

- XSLT documents are XML documents
- The XSLT document defines a **template**
  - In this case an HTML document, with some placeholders for content to be inserted
- **xsl:value-of** retrieves the value of an element and copies it into the output document
  - It places some content into the template