SDXML VT2024
Models and languages for semi-structured data and XML

XML-based languages
Usages of XML and SSD

Usages of XML and SSD

nikos dimitrakas
nikos@dsv.su.se
08-161295

Corresponding reading
Chapter 1, 4, 16, 18 of the course book
Section 31.4 of Database Systems (Connolly, Begg) 6th edition (section 30.4 in 5th edition)

XML-based languages

- **Definition of structure**
- **Definition of semantics**

- **XML basic rules**
  - **Alphabet, Vocabulary**
- **XML Schema (or DTD)**
  - **Grammar, Syntax**
- **XML Schema annotations (for humans)**
  - **Semantics, Meaning**

Definition of structure
defn of semantics

XML basic rules
alphabet,vocabulary

XML schema
- grammar,syntax

XML schema annotations
-semantics,meaning

XML-based languages - examples

- DC (Dublin Core)
- MathML
- CML (Chemical Markup Language)
- RecipeML
- ODF (Open Document Format)
- OOXML (Open Office XML)
- SVG (Scalable Vector Graphics)
- WSDL (Web Services Description Language)
- RSS (Really Simple Syndication)
- RDF and RDFS (Resource Description Framework)
  - RDF XML (RDF is much more than an XML-based language)
- And many, many more

DC
MathML
CML
RecipeML
ODF
OOXML
SVG
WSDL
RSS

DC (Dublin Core)

- Catalog metadata for books, articles, etc.
  - Close relation to RDF
- 15 elements
  - Title, Subject, Description, Type, Source, Relation, Coverage, Creator, Publisher, Contributor, Rights, Date, Format, Identifier, Language
- Many more metadata terms
  - Redefine the 15 elements

```
<book xmlns:dc="http://purl.org/dc/elements/1.1/">
  <dc:title>The blue sea</dc:title>
  <dc:description>Book with pictures about the ocean</dc:description>
  <dc:publisher>Great Books Publishing</dc:publisher>
  <!-- etc -->
</book>
```

MathML

- **Representation of mathematical equations**
  - **Visual representation**
  - **Semantic representation**

x^3+5

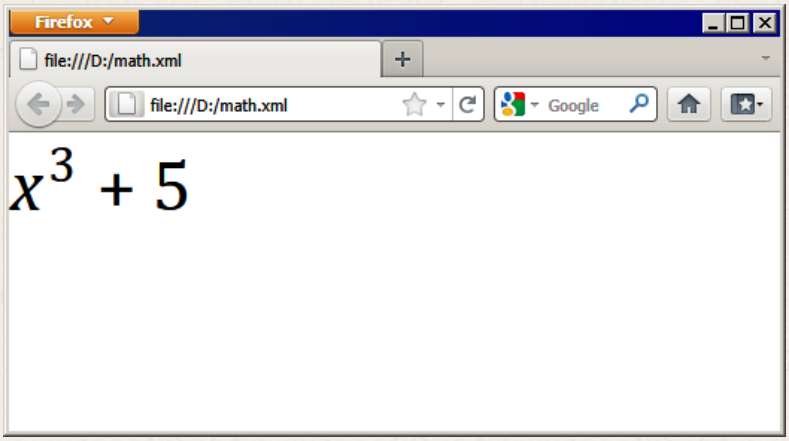
```
<math>
 <mrow>
  <msup>
   <mi>x</mi>
   <mn>3</mn>
  </msup>
  <mo>+</mo>
  <mn>5</mn>
 </mrow>
</math>
```

```
<math>
 <apply>
  <plus />
  <apply>
   <power />
   <ci>x</ci>
   <cn>3</cn>
  </apply>
  <cn>5</cn>
 </apply>
</math>
```


MathML

- **Built-in support in certain browsers:**

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<math xmlns="http://www.w3.org/1998/Math/MathML">
 <mrow>
  <msup>
   <mi>x</mi>
   <mn>3</mn>
  </msup>
  <mo>+</mo>
  <mn>5</mn>
 </mrow>
</math>
```



CML (Chemical Markup Language)

- Representation of chemical formulas
  - molecules, reactions, etc
- Elements
  - atom, atomParity, atomSet, atomType
  - electron, molecule, particle
  - action, reaction, reactant
  - angle, bond, crystal, dimension
  - formula, isotope, spectrum
  - etc.
- Namespace
  - http://www.xml-cml.org/schema

RecipeML

- Describes recipies
  - ingredients
  - process
- Elements
  - recipe
  - ingredients and ing
  - directions and step

```
<recipeml>
  <recipe>
    <head>
      <title>Chocolate Milk</title>
    </head>
    <ingredients>
      <ing>
        <amt><qty>1</qty><unit>cup</unit></amt>
        <item>milk</item>
      </ing>
      <ing>
        <amt><qty>2</qty><unit>tsp.</unit></amt>
        <item>chocolate powder</item>
      </ing>
    </ingredients>
    <directions>
      <step>Pour the milk in a cup</step>
      <step>Add the chocolate powder</step>
      <step>Mix until powder dissolves</step>
    </directions>
  </recipe>
</recipeml>
```

ODF (Open Document Format)

- For representing
  - documents
  - spreadsheets
  - presentations
  - graphs, diagrams, tables
- OASIS standard and ISO standard
- Uses MathML
- The latest version has a formal relationship to RDF and DC

ODF - for representing documents, spreadsheets,
presentations, graphs, diagrams, tables

Uses MathML
latest version - formal relationship to RDF,DC

OASIS standard and ISO standard
Uses MathML

OOXML (Open Office XML)

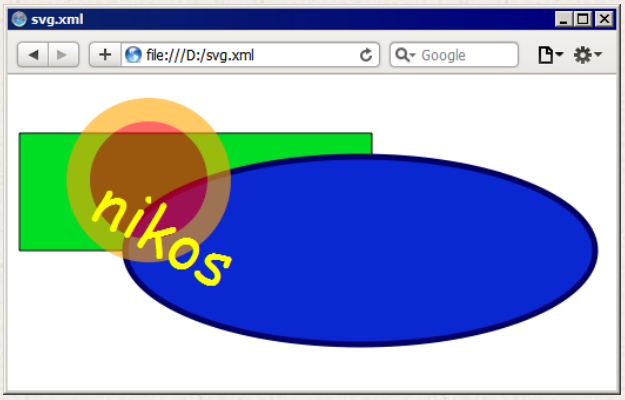
- Microsoft's counterpart to ODF
  - Became standard after much lobbying
- Comprised of many smaller languages
  - WordprocessingML
  - SpreadsheetML
  - PresentationML
  - OfficeMathML
  - DrawingML

WordProcessingML
SpreadsheetML
PresentationML
OfficeMathML
DrawingML (WSPOD)

OOXML - Microsoft's counterpart to ODF
Became standard after much lobbying

SVG (Scalable Vector Graphics)

- Represents 2D graphics
- Supports
  - vector graphics
  - images
  - text
  - interactivity
  - animation
- Supported in most web browsers
  - Some may require a plugin




```
<svg width="100%" height="100%" version="1.1" xmlns="http://www.w3.org/2000/svg">
  <rect x="10" y="50" width="300" height="100" style="fill:rgb(0,222,35); stroke-width:1; stroke:rgb(0,0,0)"/>
  <ellipse cx="300" cy="150" rx="200" ry="80" style="fill:rgb(10,40,210); stroke:rgb(0,0,100); stroke-width:5"/>
  <circle cx="120" cy="90" r="60" stroke="orange" stroke-width="20" fill="red" opacity="0.6"/>
  <text x="120" y="70" font-family="Comic Sans MS" font-size="54" fill="yellow" transform="rotate(30)">nikos</text>
</svg>
```


WSDL (Web Services Description Language)

- Used to describe Web Services
  - defines data types
  - defines calls (operations with input and output)
  - defines details about protocols
- Used together with SOAP

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
 <wsdl:types>
  <complexType name="DTCustomer">
   <sequence><element name="customerName" nillable="true" type="xsd:string"/>
   <element name="userName" nillable="true" type="xsd:string"/></sequence>
  </complexType>
 </wsdl:types>
 <wsdl:message name="loginRequest">
  <wsdl:part name="in0" type="xsd:string"/>
  <wsdl:part name="in1" type="xsd:string"/>
 </wsdl:message>
 <wsdl:message name="loginResponse">
  <wsdl:part name="loginReturn" type="DTCustomer"/>
 </wsdl:message>
 <wsdl:portType name="B2BService">
  <wsdl:operation name="login" parameterOrder="in0 in1">
   <wsdl:input message="loginRequest" name="loginRequest"/>
   <wsdl:output message="loginResponse" name="loginResponse"/>
  </wsdl:operation>
 </wsdl:portType>
 <wsdl:service name="B2BServiceService">
  <wsdl:port name="B2BWebService">
   <wsdlsoap:address location="http://iv1023.kth.se/services/B2BWebService"/>
  </wsdl:port>
 </wsdl:service>
</wsdl:definitions>
```

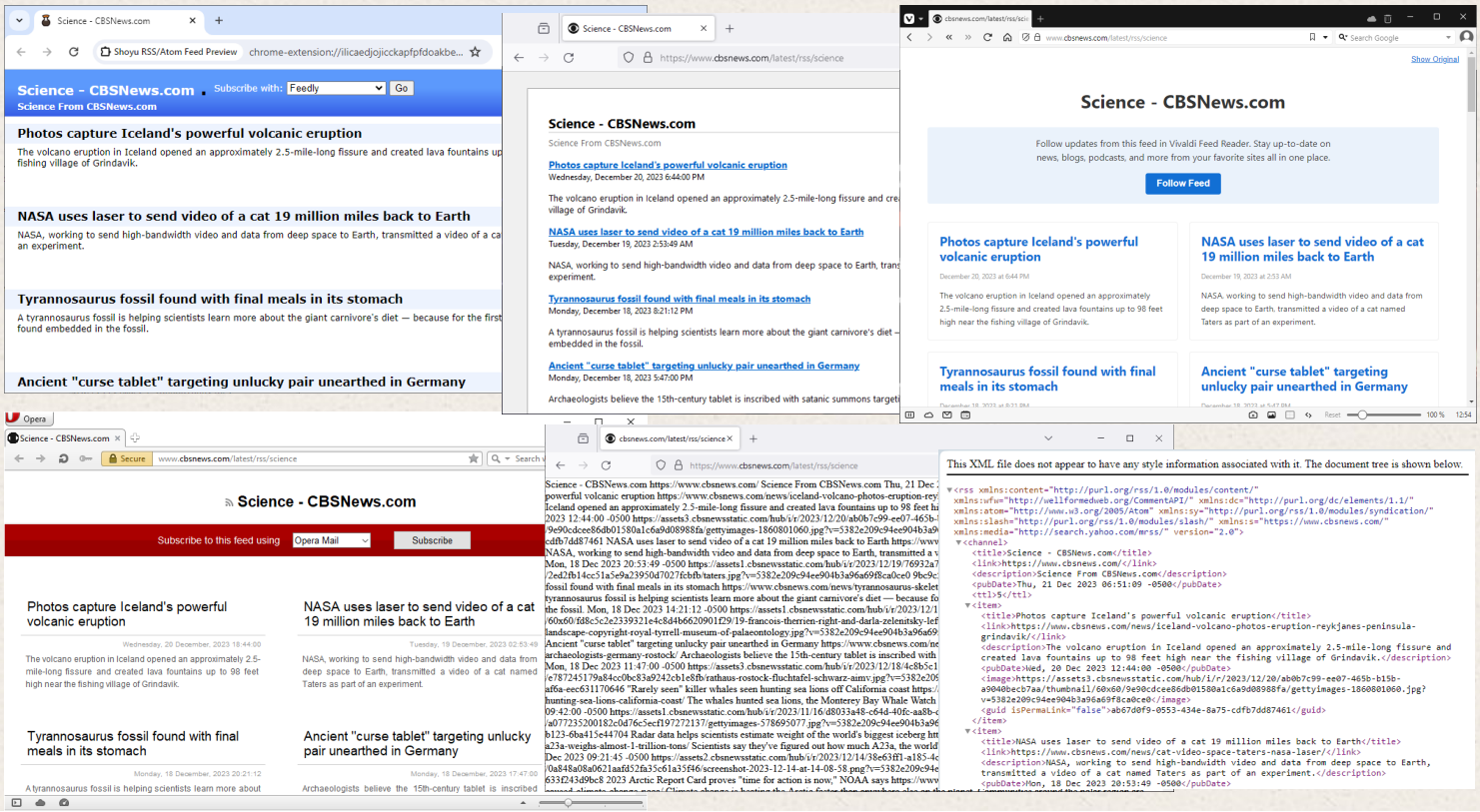
RSS (Really Simple Syndication)

- Used mostly for news feeds, blogs, ads, etc.
- Earlier "RDF Site Summary" with a relationship to RDF
- AKA "Rich Site Summary"
- Often combined with DC

```
<rss version="2.0">
  <channel>
    <link>http://www.cbsnews.com/</link>
    <title>Breaking News: CBS News</title>
    <item>
      <title>Stocks turn lower as optimism about jobs fades</title>
      <link>http://feeds.cbsnews.com/~r/CBSNewsMain/~3/WdSytMQjwds/main20088553.shtml</link>
      <description>Day after worst single-day drop since 2008, Dow jumps on strong hiring report - then quickly loses
       gains</description>
      <pubDate>Fri, 05 Aug 2011 07:47:45 EDT</pubDate>
    </item>
    <item>
      <title>Syrian troops fire on defiant protesters; 8 dead</title>
      <link>http://feeds.cbsnews.com/~r/CBSNewsMain/~3/5eg3GhnUoG0/main20088559.shtml</link>
      <description>Regime's security forces open fire on tens of thousands across nation calling for downfall of President
       Bashar Assad</description>
      <pubDate>Fri, 05 Aug 2011 07:58:53 EDT</pubDate>
    </item>
  </channel>
</rss>
```


RSS (Really Simple Syndication)

- Built-in support in some browsers
  - Example: https://www.cbsnews.com/latest/rss/science



RDF (Resource Description Framework)

- **Used to describe resources (anything in the universe)**
  - **Identify a resource uniquely with a URI**
  - **State something about the resource**
- **RDF Statements**
  - **A subject (the resource)**
  - **A predicate (a property, similar to attributes in other terminologies)**
  - **An object (the value, may be a resource)**
- **RDF representations**
  - **Graph**
  - **XML (so "RDF/XML" is an XML-based language)**
- **RDF Schema (RDFS)**
  - **For defining rules**
  - **Also an XML-based language**
  - **An RDFS document is an RDF document**

RDF statements
a subject(the resource)
a predicate
an object

RDF Representations
Graph
XML

RDF - used to describe resources
Identify resource uniquely with URI
State something about the resource

RDF/XML example

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:sdxml="sdxml/predicates/">
  <rdf:Description about="sdxml/res/p1">
    <sdxml:name>Maria</sdxml:name>
    <sdxml:hasAsBoss rdf:resource="sdxml/res/p2" />
  </rdf:Description>
  <rdf:Description about="sdxml/res/p2">
    <sdxml:name>Lisa</sdxml:name>
  </rdf:Description>
<rdf:RDF>
```

RDF, DC and other languages

- Everything can be considered to be a resource
  - RSS channel, item
  - MathML
  - CML
  - RecipeML
  - ODF documents
- All resources may have DC metadata
  - An ODF document may have a dc:creator, dc:date
  - An rss:channel and rss:item may have dc:date, dc:language, dc:title
  - etc.

Open Data

- Availability
  - free to use
  - in a standard format
  - CSV
  - XML
  - JSON
  - RDF

Usages of XML and JSON

- Open data
  - RDF
  - ODF
  - OOXML
- Metadata in files
- APIs (Application Programming Interface)
  - XML
  - JSON
- Protocols
- Configurations
  - Programming languages
  - Software
- Data transfer
  - QTI (Question & Test Interoperability) (supports MathML)

Metadata in Files

- PDF
  - Adobe metadata with RDF and DC (XML)
- Try to open a pdf in Notepad++

APIs

- Input and output as
  - JSON
  - XML
- REST (Representational State Transfer)
  - Combining HTTP with XML or JSON
  - GET, POST, PUT, DELETE
- SOAP (Simple Object Access Protocol)
  - Sending messages as XML
  - Built-in support for error handling

REST API - Example

- GET
- JSON as response

GET HTTP://sdxml.dsv.su.se/api/course/345 accept: application/json

Response:
HTTP 200 (success, found it) (404 if not found, and no body)
Body

```
{
  "id" : 245, "code" : "DB101",
  "nameEN" : "Database methodology",
  "nameSV" : "Databasmetodik"
}
```

REST API - Example

- GET
- XML as response

GET HTTP://sdxml.dsv.su.se/api/course/345 accept: application/xml

Response:
HTTP 200 (success, found it) (404 if not found, and no body)
Body

```
<course>
  <id>245</id>
  <code>DB101</code>
  <nameEN>Database methodology</nameEN>
  <nameSV>Databasmetodik</nameSV>
</course>
```


REST API - Example

- PUT (update)
- XML in the request body

PUT HTTP://sdxml.dsv.su.se/api/course/345
Content-Type: application/xml
Body

```
<course>
  <id>245</id>
  <code>DB101</code>
  <nameEN>Database methodology for beginners</nameEN>
  <nameSV>Databasmetodik för nybörjare</nameSV>
</course>
```

Response: HTTP 200 (success) or other code

REST API - Example

- POST (create)
- With JSON in the request body

```
POST HTTP://sdxml.dsv.su.se/api/course
Content-Type: application/json
Body
{
  "code" : "SDXML",
  "nameEN" : "Semi-structured data and XML",
  "nameSV" : "Semistrukturerade data och XML"
}
```

Response: HTTP 200 (success) or other code

Perhaps more as Body or Header

SOAP Message - Example

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <getCourseInfoRequest>
      <code>SDXML</code>
    </getCourseRequest>
  </soap:Body>
</soap:Envelope>

<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  <soap:Body>
    <getCourseInfoResponse>
      <code>SDXML</code>
      <nameEN>Semi-structured data and XML</nameEN>
      <nameSV>Semistrukturerade data och XML</nameSV>
    </getCourseResponse>
  </soap:Body>
</soap:Envelope>
```

Configurations

- Software like
  - Web servers, FTP servers
    - » Tomcat
    - » Filezila
  - Operating systems
    - » Windows
    - » Linux
  - Desktop application
    - » VLC
    - » Notepad++
    - » NetBeans
    - » Firefox
- Pretty much everywhere!

Programming

- Java technologies
  - Maven
  - Spring
  - Hibernate
  - Jasper
  - Struts
  - Seam
  - JavaFX (FXML)
  - JUnit

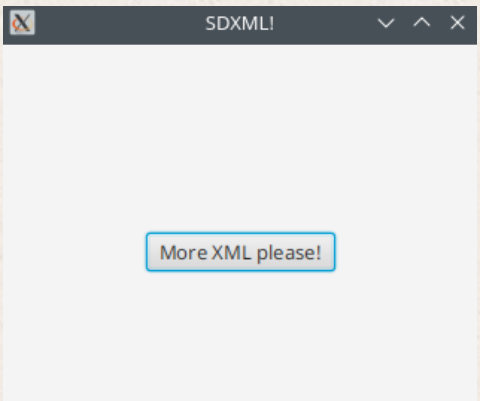
Maven

- Dependencies
- Building

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
                             https://maven.apache.org/xsd/maven-4.0.0.xsd">
    <dependencies>
        ...
    </dependencies>
    <build>
        ...
    </build>
</project>
```


JavaFX with FXML

- Java GUI




```
<?xml version="1.0" encoding="UTF-8"?>
<VBox alignment="CENTER" spacing="20.0" xmlns:fx="http://javafx.com/fxml"
      fx:controller="sdxml.Controller">
    <padding>
        <Insets bottom="20.0" left="20.0" right="20.0" top="20.0"/>
    </padding>
    <Label fx:id="myText"/>
    <Button text="More XML please!" onAction="#onButtonClick"/>
</VBox>
```

Spring

- Beans
- Injections
- Transactions
- Security

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans">
        <bean id="myBean" class="sdxml.MyBean" >
            <property name="name" value="XML"/>
            <property name="status" value="5"/>
        </bean>
</beans>
```


Hibernate

- ORM (Object Relational Mapping)
- Classes to tables and columns

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-mapping>
<hibernate-mapping>
   <class name="sdxml.Person" table="Pesron">
      <id name="id" column="PersonID" type="int">
         <generator class="native"/>
      </id>
      <property name="fname" column="Firstname" type="string" not-null="true"/>
      <property name="lname" column="Lastname" type="string" not-null="true"/>
      <property name="alive" column="Alive" type="boolean"/>
   </class>
</hibernate-mapping>
```

Struts

- Configuration of web application
  - actions
  - validations
  - …
- JSPs (Java server pages)

```
<?xml version="1.0" encoding="UTF-8" ?>
<struts>
   <package name="sdxml" extends="default" namespace="/sdxml">
      <action name="start" class="sdxml.StartAction" method="start">
         <result name="success">great.jsp</result>
         <result name="error">sorry.jsp</result>
      </action>
   </package>
</struts>
```


What to do next

- Quiz about Semi-structured data, XML-based languages, usages (Quiz 4)