

6. XML and JSON

Fundamentals of Data Management

Dr Charaka Palansuriya

- Why do we need XML and JSON?
- What is XML
 - Data Validation with Schemas
 - XML parsers
 - Querying XML documents
 - Where XML documents are used
- What is JSON
 - Data Validation with Schemas
 - JSON parsers and querying
- Summary

Why do we need XML and JSON?

- Mainly used for platform and language neutral data exchange and storage
 - Both text based
- Many web services support both XML and JSON based request and responses
- Both used for storing application data
 - Application configuration
 - Persistent objects
 - Log files
- Lets look at each one in turn

- eXtensible Markup Language
 - A mark up language derived from SGML
 - Helps to both structure the data and give meaning to the data that other application can use
 - i.e., XML provides a metalanguage
 - Extensible
 - E.g., new user-defined elements can be added to the existing ones
 - Used to define open standards
 - SOAP
 - WSDL
- A core technology behind web 2.0, AJAX and many web services

- An application configuration XML file

```
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://www.netbeans.org/ns/project/1">
  <type>org.netbeans.modules.java.j2seproject</type>
  <configuration>
    <data xmlns="http://www.netbeans.org/ns/j2se-project/3">
      <name>BioDare_GUI</name>
      <source-roots>
        <root id="src.dir"/>
      </source-roots>
      <test-roots>
        <root id="test.src.dir"/>
      </test-roots>
    </data>
  </configuration>
</project>
```

- Declaration
 - **<?xml version="1.0" encoding="UTF-8"?>**
 - Useful to a parser
 - optional
- Tags
 - Text in between < and >
- Elements
 - Start tag and an end tag defines an element
 - Could be self contains, e.g., <self/>
 - XML document contains one root element

- **Attributes**
 - Name-value pairs that provide additional information about an element.
 - Can use either single or double quotes to encode values
 - Each attribute name is unique within the same element
- **Comments**
 - Text that appears between `<!--` and `--!>`
 - Parsers ignore comments

- XML Schemas are used to validate XML documents
 - Describes the valid structure and content for a XML document
- Two schema languages are commonly used
 - Document Type Definition (DTD)
 - W3C XML Schemas
- Industries and organisations have standardised schemas to exchange data
 - Banking
 - Life sciences
 - Health care
 - Etc.


```
<?xml version="1.0" encoding="UTF-8"?>

<xsd:schema xmlns= "http://www.mycompany.com/ns/orders"
             xmlns:xsd="http://www.w3.org/2001/XMLSchema"
             targetNamespace="http://www.mycompany.com/ns/orders">

  <xsd:element name="order" type="orderType">

    <xsd:complexType name="orderType">
      <xsd:sequence>
        <xsd:element name="item" type="xsd:string" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>

  </xsd:element>

</xsd:schema>
```

- An example XML document that adheres to the schema defined in the previous slide

```
<?xml version="1.0" encoding="UTF-8"?>

<co:co xmlns:co= "http://www.mycompany.com/ns/orders"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://www.mycompany.com/ns/orders
                            http://www.mycompany.com/schema/co.xsd">

  <order>
    <item> kettle</item>
    <item> toaster </item>
  </order>

</co:co>
```

- DOM
 - “Document Object Model”
 - A parser that reads entire XML document and then generates a tree data structure representing the XML components
 - Not suitable for reading very large XML documents
- SAX
 - “Simple API for XML”
 - A push parser (a.k.a, event based parsing)
 - Parser sends notifications to the application as it encounters various XML components in the document.
 - Uses SAX parser event call backs to implement application logic
 - Suitable for reading very large XML documents

- XPath
 - Allows navigating an XML document
 - Allows navigating to a specific node
 - Nodes: elements, attributes, text, comments, etc.
 - For example:
 - `/order/item[1]`
 - Selects the first “item” element from an “order” element
 - It’s a W3C standard
- XQuery
 - Allows querying an XML document
 - Uses XPath expressions
 - E.g., `doc(“orders.xml”)/order/item[price<12]`
 - Find all the “item” elements under the “order” element where “price” element is less than 12
 - A W3C standard

- XML can be use to store an application configuration and state
- An example: Java Architecture for XML Binding (JAXB)
 - Allows processing of XML data within Java without having to use XML parsers explicitly
 - Allows Java classes to be stored as XML representations
 - Can define a schema and generate corresponding classes or vice versa
 - Can be used to persist in memory objects
 - i.e., preserve state
 - Can be used to store an application configuration

- Many Web Services are described using WSDL
- WSDL
 - Web Services Description Language
 - Operations and data input/output to a web service are described using XML
 - Can generate a web service application skeleton using a WSDL
- SOAP
 - Simple Object Access Protocol
 - An XML based communication protocol
 - Allows platform neutral communication between different applications
 - Usually over http
 - Typically, WSDL based Web Services communicate using SOAP messages

- RDF (Resource Description Framework) are written in XML
 - RDF/XML documents can be validated from
 - <http://www.w3.org/RDF/Validator/>

```
<?xml version="1.0"?>
```

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
```

```
  <rdf:Description rdf:about="http://www.w3.org/">
    <dc:title>World Wide Web Consortium</dc:title>
  </rdf:Description>
```

```
</rdf:RDF>
```

- JavaScript Object Notation
 - A light weight data exchange format
 - Much easier for a human to read and write than XML
 - An example JSON document:

```
{
  "CustomerID": 1,
  "Name": "David Hudson",
  "Address": "23 Lygon St",
  "Products": [
    {
      "Item": "Kettle",
      "Price": 10
    },
    {
      "Item": "Toaster",
      "Price": 12
    }
  ]
}
```

- Originated from JavaScript (as the name suggests)
 - But JSON is a language independent data format
 - JSON documents have .json extension
- Supported in virtually all languages due to its popularity
 - Python, Java, C++, etc.
 - Generally, reading and writing JSON is much more straightforward than XML
- Also Used for persisting (storing) application objects
 - i.e., can be used to store a state of an application
- Commonly used in client and web server data exchanges

- Similar to XML Schema, a JSON schema describes valid content for application/domain specific JSON documents
 - Can be used for industry specific data exchanges
- IETF (internet) draft status
- Language support exists for JSON Schema
 - E.g.,
 - there is an implementation of JSON schema validation for Python, <https://pypi.python.org/pypi/jsonschema>
 - Java, <https://github.com/fge/json-schema-validator>
 - Support for other languages too
 - see <http://json-schema.org/implementations>

- An example from <http://json-schema.org/examples.html>

```
{
  "title": "Example Schema",
  "type": "object",
  "properties": {
    "firstName": {
      "type": "string"
    },
    "lastName": {
      "type": "string"
    },
    "age": {
      "description": "Age in years",
      "type": "integer",
      "minimum": 0
    }
  },
  "required": ["firstName", "lastName"]
}
```

- Language specific parsers
 - Java API for JSON processing provides
 - Object Model API (similar to DOM for XML)
 - A Streaming API (Similar to SAX for XML)
 - Python provides a JSON library (from v2.6 onwards)
- No standard query language for JSON
 - Usually use a language specific access mechanism
 - Some effort has gone into this area:
 - E.g, JsonPath, JsonQuery

- XML and JSON allow platform and language independent data exchange
- Both data formats can be used to store application data
- XML Schema facilitates validating XML documents
 - A W3C standard
- XML is used to define Web Services and write RDF documents
- Language support for JSON Schema exist
 - Only a IETF draft at the moment
- JSON is becoming increasingly popular due to its light weight nature
 - Both human and machine friendly

- Extensible Markup Language (XML), <http://www.w3.org/XML/>
- JSON, <http://json.org/>
- JSON Basics, <http://www.elated.com/articles/json-basics/>
- JSON Schema, <http://json-schema.org/>