



# Introduction to the XML Family and XML Access Languages

Brian Matthews

- **Brian Matthews**
  - Information Management Group, e-Science Centre, CCLRC
  - Deputy, W3C Office for the UK and Ireland
- **Set the context for the day**
- **Provide some background**
  - Describe the landscape and history of XML Access languages
  - Set some questions for the day

- In the beginning there was SGML
- With an access language DSSSL
  - Document Style Semantics and Specification Language (ISO/IEC 10179:1996)
  - Used extensible Scheme (Lisp-like) syntax
  - transformations as well as style
  - transform one SGML form to another

```
(element H1
  (make paragraph
    font-size: (* 1.5 (inherited-
font-size))
    space-before: 2.5em
    space-after: 1.5em
    escapement-space-after: 0.1em
    font-weight: 'bold (uppercase)
  ))
(element H2
  (make paragraph
    font-size: (* 1.3 (inherited-
font-size))
    space-before: 1.5em
    space-after: 1em
    font-weight: 'bold))
(element H3
  (make paragraph
    font-size: (* 1.3 (inherited-
font-size))
    space-before: 1em
    space-after: 0.25em
    font-weight: 'bold))
```

- Leading the Web to its full potential
- Founded by Web inventor Tim Berners-Lee in 1994
  - Engineering the Standards that Make the Web Work.
  - From a Web of Documents ...
  - ... towards a Web of data and services that is:
    - Meaning-ful
    - on Everything
    - for Everyone
  - ... and Interoperable, Trustworthy, Evolvable, Decentralized
- Now with 419 members worldwide
- 3 hosts and 16 offices world wide

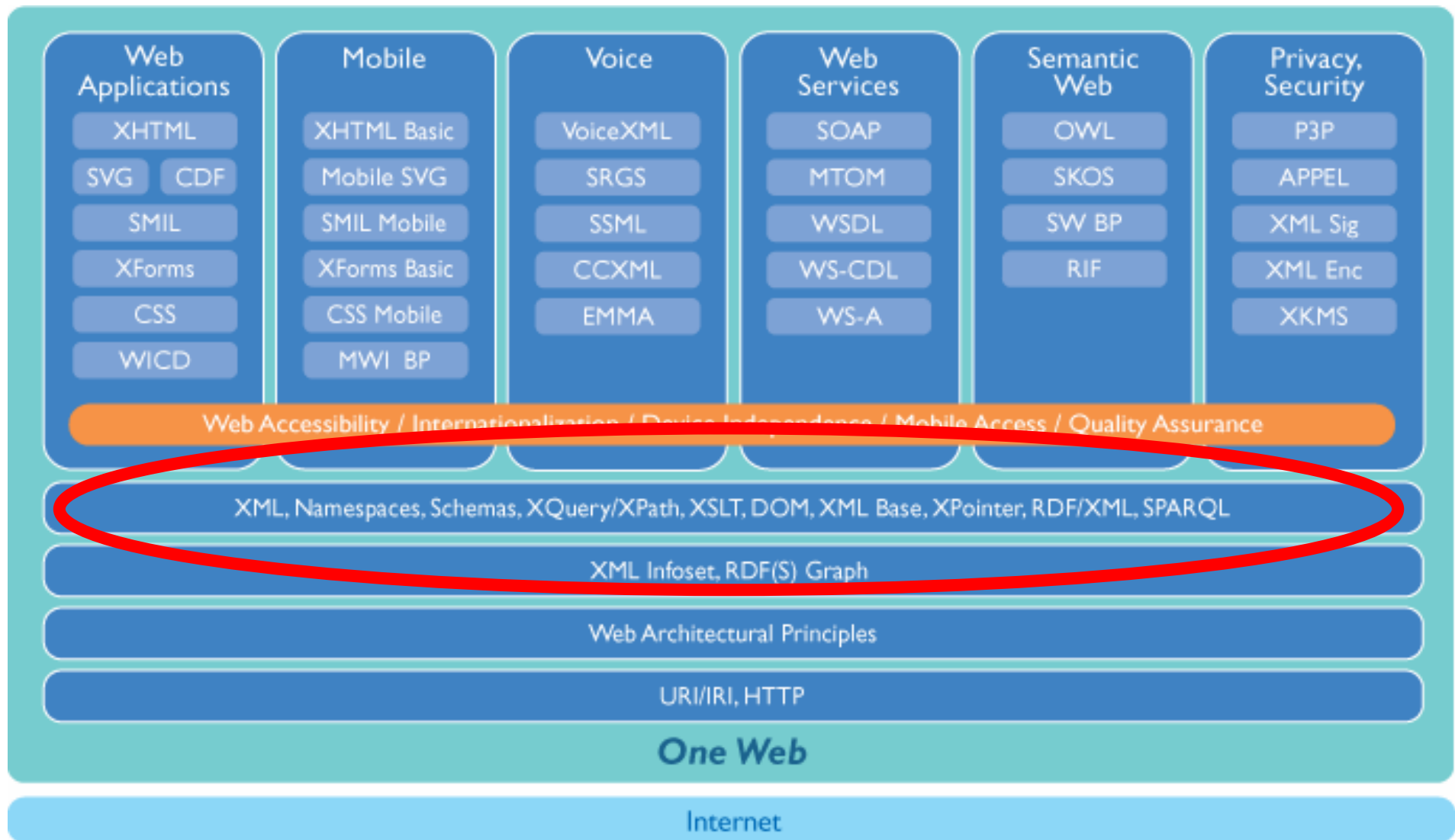
- **Web of Meaning**
  - Data, information, knowledge
  - Text, graphics, animation, sound,
  - Security, privacy, trust
- **Web on Everything**
  - Small to large displays. Fixed to mobile devices. Hi to low bandwidth.
  - Computers to appliances to cars, etc.
- **Web for Everyone**
  - Language, culture, location
  - User capabilities
- **Interoperability**
  - Especially at its foundation, Web's components must work together

- Over 80 Recommendations



# Today's Topic

- XML a core part of this architecture



- Developed XML language 1996-98
  - XML 1.0 – 10 February 1998
- Immediately recognised the need for further recommendations:
  - Infoset
  - Namespaces
  - Schema Languages
  - Style Sheet Languages
  - Query Languages
  - DOM



- In particular a set of XML Access Languages have emerged
  - XPath
  - XSL (XSLT, XSL-FO)
  - XQuery
  - DOM (not covered today)
- Languages which:
  - access
  - manipulate
  - transform
  - query
- Also RDF provides another query language – SPARQL.

- XPath – simplest data access language
- Designed as a utility for other specs
  - “XPath is a language for addressing parts of an XML document”
- But came to be used as an simple query language in its own right.
  - XPath 1.0 – November 1999
    - <http://www.w3.org/TR/1999/REC-xpath-19991116>
  - XPath 2.0 - Candidate Rec., June 2006
    - Now a large subset of XQuery!
    - <http://www.w3.org/TR/2006/CR-xpath20-20060608/>
  - XPath 2.0 Full-Text – Working draft (with XQ-FT)
    - <http://www.w3.org/TR/2006/WD-xquery-full-text-20060501/>
- Now under the joint development of XSL and XQuery

- A simple expression to address into an XML document tree:

```
/bib/book/author[1]
```

– Returns sequence of all first authors

- Can address across axes:

```
following::Book[Year="2006"]
```

- XPath 2.0 introduces some expressions from XQuery

```
for $x in /bib/book return $x/price * 1.48
```

- XSL - Extensible Stylesheet Language
- A language to XML as DSSSL is to SGML
- Styling and Transformation
  - XSL Transformations 1.0 – November 1999
  - also XML-FO for formatting
- XSLT transforms an old XML document into a new one
  - Uses XPath as match expression in templates
- Now on Version 2.0 – in Candidate Recommendation
  - <http://www.w3.org/TR/xslt20/>
- Can be used for querying as well as transformation

```
<xsl:template match="book" >
  <xsl:if test="publisher='Addison Wesley' and year >
    2003">
    <title><xsl:value-of select="@title" /></title>
  </xsl:if>
</xsl:template>
```

```
<xsl:template match="bib">
  <titles>
    <xsl:apply-templates />
  </titles>
</xsl:template>
```

- XML Query Workshop – 5 Dec 1998
- XML Query Language
  - “XML Database” query language
- Now a set of Candidate Recommendations for 1.0
  - see <http://www.w3.org/XML/Query/>
- Working in conjunction with the XSL WG.
- Can be seen as an extended XPath 2.0.
  - “functional programming” semantics
  - emphasis on “human readable” syntax
  - FLWOR expressions
  - function definitions
  - Select statements and Joins
- Can transform as well as query

- List the titles of books published by Addison Wesley after 2003.

```
for $b in doc("books.xml")//book
where $b/publisher = "Addison Wesley"
    and $b/year > "2003"
return $b/@title
```

- List each publisher and the average price of its books.

```
for $p in
distinct-values(doc("books.xml")//publisher)
let $a := avg(doc("books.xml")/book[publisher =
    $p]/price)
return <publisher>
    <name> { $p } </name> ,
    <avgprice> { $a } </avgprice>
</publisher>
```

- Part of the Semantic Web effort
  - RDF Data Access Working Group
  - Based on previous languages: RDQL, Squish
- Access to data described using the Resource Description Framework (RDF)
  - Runs queries over Stores of Triples
- Now in Candidate Recommendation
  - <http://www.w3.org/2001/sw/DataAccess/>
  - A Query Language
  - A protocol for passing queries
  - Encoding results into XML
- A very different language – but designed for the Web

```
PREFIX dc: http://purl.org/dc/elements/1.1/
PREFIX ns: <http://example.org/ns#>
SELECT ?title ?price
WHERE {
    ?x ns:price ?price .
    FILTER (?price < 30) .
    ?x dc:title ?title .
}
```

- [XML Query Requirements](#)
- [XML Query Use Cases](#)
- [XQuery 1.0 and XPath 2.0 Data Model](#)
- [XSLT 2.0 and XQuery 1.0 Serialization](#)
- [XQuery 1.0 and XPath 2.0 Formal Semantics](#)
- [XQuery 1.0: An XML Query Language](#)
- [XML Syntax for XQuery 1.0 \(XQueryX\)](#)
- [XQuery 1.0 and XPath 2.0 Functions and Operators](#)
- [XPath Requirements Version 2.0](#)
- [XML Path Language \(XPath\) 2.0](#)
- [XML Query and XPath Full-Text Requirements](#)
- [XML Query and XPath Full-Text Use Cases](#)
- [XQuery 1.0 and XPath 2.0 Full-Text](#)
- [XQuery Update Facility Requirements](#)
- [XQuery Update Facility](#)
- [XQuery Update Facility Use Cases](#)
- [Building a Tokenizer for XPath or XQuery](#)
- [XSL Transformations \(XSLT\) Version 2.0](#) (by the XSL Working Group)
- [SPARQL Query Language for RDF](#) 6 Apr 2006
- [SPARQL Protocol for RDF](#) W3C Candidate Recommendation 6 Apr 2006
- [SPARQL Query Results XML Format](#)

This survey leads us to a number of questions:

- Why a number of access languages?
- What are their relative strengths and weaknesses?
- Which one to use in what situations?
- How do they work together?
- Where are they going?

The rest of the day should help us address these questions