



Distributed Computing

Lesson 16: XML

Thomas Weise · 汤卫思

tweise@hfu.edu.cn · <http://www.it-weise.de>

Hefei University, South Campus 2
Faculty of Computer Science and Technology
Institute of Applied Optimization
230601 Shushan District, Hefei, Anhui, China
Econ. & Tech. Devel. Zone, Jinxiu Dadao 99

合肥学院 南艳湖校区/南2区
计算机科学与技术系
应用优化研究所
中国 安徽省 合肥市 蜀山区 230601
经济技术开发区 锦绣大道99号

1 Introduction

2 Structure



website

- Learn about the XML format as the basis for many modern distributed applications

- Many data formats are binary and cannot be interpreted by human beings.

[illegible]

- Many data formats are binary and cannot be interpreted by human beings.
 - This is good when we absolutely need the the highest possible performance

- Many data formats are binary and cannot be interpreted by human beings.
 - This is good when we absolutely need the the highest possible performance
 - But it will make documentation, maintenance, and finding of bugs very hard

- Many data formats are binary and cannot be interpreted by human beings.
 - This is good when we absolutely need the the highest possible performance
 - But it will make documentation, maintenance, and finding of bugs very hard
 - Plus: How about the future, e.g., 30 years from now...

- Many data formats are binary and cannot be interpreted by human beings.
 - This is good when we absolutely need the the highest possible performance
 - But it will make documentation, maintenance, and finding of bugs very hard
 - Plus: How about the future, e.g., 30 years from now. . .
- Other data formats are text-based and human readable, but specialized

Listing: Some Quotations from the Universal Library (\LaTeX , .tex^[2, 3])

```
\begin{document}
  \title{Some Quotations from the Universal Library}
  ...
  \section{Famous Quotes}
  \subsection{By William I}
  \textbf{\cite{Sonnet XVIII}{shakespeare-sonnets-1609}}
  \begin{verse}
    Shall I compare thee to a summer's day?\\
    Thou art more lovely and more temperate. \\
    Rough winds do shake the darling buds of May, \\
    And summer's lease hath all too short a date. \\
    Sometime too hot the eye of heaven shines, \\
    And often is his gold complexion dimmed. \\
    ...
    \quad So long as men can breathe, or eyes can see,\\
    \quad So long live this, and this gives life to thee. \\
  \end{verse}
  ...
  \bibliographystyle{abbrv}
  ...
\end{document}
```

Listing: Some Quotations from the Universal Library (.html)

```
<html>
  <head>
    <title>Some Quotations from the Universal Library</title>
  </head>
  <body>
    <b><font face="Arial" size=5><p>Some Quotations from the Universal
      Library</p></font>
    <i><font face="Arial"><p>1 Famous Quotes</p></b></i><p>1.1 By William
      I</p>
    <b><p>[2, Sonnet XVIII]</p></b>
    <p>Shall I compare thee to a summer's day?</p>
    <p>Thou art more lovely and more temperate.</p>
    <p>Rough winds do shake the darling buds of May,</p>
    <p>And summer's lease hath all too short a date.</p>
    <p>Sometime too hot the eye of heaven shines,</p>
    <p>And often is his gold complexion dimmed.</p>
    ...
  </body>
</html>
```

- Many data formats are binary and cannot be interpreted by human beings.
 - This is good when we absolutely need the the highest possible performance
 - But it will make documentation, maintenance, and finding of bugs very hard
 - Plus: How about the future, e.g., 30 years from now. . .
- Other data formats are text-based and human readable, but specialized
- We want: General, human readable data format that can be specialized for different tasks

- Many data formats are binary and cannot be interpreted by human beings.
 - This is good when we absolutely need the the highest possible performance
 - But it will make documentation, maintenance, and finding of bugs very hard
 - Plus: How about the future, e.g., 30 years from now. . .
- Other data formats are text-based and human readable, but specialized
- We want: General, human readable data format that can be specialized for different tasks
 - very compatible, future save, many different purposes

- Many data formats are binary and cannot be interpreted by human beings.
 - This is good when we absolutely need the the highest possible performance
 - But it will make documentation, maintenance, and finding of bugs very hard
 - Plus: How about the future, e.g., 30 years from now. . .
- Other data formats are text-based and human readable, but specialized
- We want: General, human readable data format that can be specialized for different tasks
 - very compatible, future save, many different purposes
 - This is **X**ML!

Listing: Some Quotations from the Universal Library (.xml)

```

<?xml version="1.0"?>
<universal_library>
  <books>
    <book><title>Some Quotations from the Universal Library</title>
      <section><title>Famous Quotes</title>
        <subsection><title>By William I</title>
          <quote bibref="shakespeare-sonnets-1609">
            <title>Sonnet XVIII</title>
            <verse>
              <line>Shall I compare thee to a summer's day?</line>
              <line>Thou art more lovely and more temperate.</line>
              <line>Rough winds do shake the darling buds of May,</line>
            </verse>
            ...
          </quote>
        </subsection>
      </section>
    </book> ...
  </books>
</universal_library>

```

- XML is an “Extensible Markup Language” ^[4, 5]

- XML is an “Extensible Markup Language” ^[4, 5]
- a meta language, i.e., a language to describe a language

- XML is an “Extensible Markup Language” [4, 5]
- a meta language, i.e., a language to describe a language
- “HTML [6, 7] minus presentation plus own tags”

- XML is an “Extensible Markup Language” [4, 5]
- a meta language, i.e., a language to describe a language
- “HTML [6, 7] minus presentation plus own tags”
- a semi-structured data model (data model part of text)

- XML is an “Extensible Markup Language” [4, 5]
- a meta language, i.e., a language to describe a language
- “HTML [6, 7] minus presentation plus own tags”
- a semi-structured data model (data model part of text)
- a self-describing exchange format

- Origins in HTML ^[6, 7] and SGML (ISO Standard, complex, 1986 ^[8])

- Origins in HTML ^[6, 7] and SGML (ISO Standard, complex, 1986 ^[8])
- XML is W3C-Standard ^[4, 5] containing XML syntax + Document Type Definition (DTDs)

- Origins in HTML ^[6, 7] and SGML (ISO Standard, complex, 1986 ^[8])
- XML is W3C-Standard ^[4, 5] containing XML syntax + Document Type Definition (DTDs)
- XML offers a subset of the SGML function set

- Origins in HTML ^[6, 7] and SGML (ISO Standard, complex, 1986 ^[8])
- XML is W3C-Standard ^[4, 5] containing XML syntax + Document Type Definition (DTDs)
- XML offers a subset of the SGML function set
- HTML is an application of SGML, a language produced on basis of SGML

- Origins in HTML ^[6, 7] and SGML (ISO Standard, complex, 1986 ^[8])
- XML is W3C-Standard ^[4, 5] containing XML syntax + Document Type Definition (DTDs)
- XML offers a subset of the SGML function set
- HTML is an application of SGML, a language produced on basis of SGML
- XHTML ^[7] is an application of XML

Listing: HTML/XHTML: defines for the text looks like

```
<h1>Bibliography</h1>
<p><i>Foundations of DBs</i>, Abiteboul, Hull, Vianu
<br>Addison-Wesley, 1995
<p><i>Logics for DBs and ISs</i>, Chomicki, Saake, eds.
<br>Kluwer, 1998
```

HTML: fixed set of tags describe visual appearance and document structure

Listing: XML: defines what the text consists of

```
<bibliography>
<book><title>Foundations of DBs</title>
<author>Abiteboul</author>
<author>Hull</author>
<author>Vianu</author>
<publisher>Addison-Wesley</publisher>
...
</book>
<book>... <editor>Chomicki</editor>... </book> ...
</bibliography>
```

XML: set of tags can be freely defined. only semantics represented

- XML can easily be parsed

- XML can easily be parsed
 - includes parse tree in the data

- XML can easily be parsed
 - includes parse tree in the data
 - structure and semantic can easily be reconstructed

- XML can easily be parsed
 - includes parse tree in the data
 - structure and semantic can easily be reconstructed
- Validity criteria can be defined with a meta-language (e.g., DTD, XSD ^[9–14])

- XML can easily be parsed
 - includes parse tree in the data
 - structure and semantic can easily be reconstructed
- Validity criteria can be defined with a meta-language (e.g., DTD, XSD ^[9–14])
- XML separates presentation from contents

- XML can easily be parsed
 - includes parse tree in the data
 - structure and semantic can easily be reconstructed
- Validity criteria can be defined with a meta-language (e.g., DTD, XSD ^[9–14])
- XML separates presentation from contents
 - presentation via style sheets ^[15, 16]

- XML can easily be parsed
 - includes parse tree in the data
 - structure and semantic can easily be reconstructed
- Validity criteria can be defined with a meta-language (e.g., DTD, XSD ^[9–14])
- XML separates presentation from contents
 - presentation via style sheets ^[15, 16]
 - different style sheets can be used for different output formats (HTML, PDF, ...)

Different perspectives on XML

Different perspectives on XML:

- Documents:
 - XML allows for markup of text parts and adds semantics and structure to the text

Different perspectives on XML:

- Documents:
 - XML allows for markup of text parts and adds semantics and structure to the text
- Data bases

Different perspectives on XML:

- Documents:
 - XML allows for markup of text parts and adds semantics and structure to the text
- Data bases:
 - XML is the most prominent example for semi-structured data models (i.e., the data model is part of the data)

Different perspectives on XML:

- Documents:
 - XML allows for markup of text parts and adds semantics and structure to the text
- Data bases:
 - XML is the most prominent example for semi-structured data models (i.e., the data model is part of the data)
 - XML covers the spectrum between unstructured and fully structured data

- XML is the basic format used in enterprise distributed computing system

- XML is the basic format used in enterprise distributed computing system
 - Web Services^[17] use SOAP^[18, 19] – an XML-based standard

- XML is the basic format used in enterprise distributed computing system
 - Web Services^[17] use SOAP^[18, 19] – an XML-based standard
 - Services are described using WSDL^[20] – an XML-based standard

- XML is the basic format used in enterprise distributed computing system
 - Web Services^[17] use SOAP^[18, 19] – an XML-based standard
 - Services are described using WSDL^[20] – an XML-based standard
 - QoS criteria of services are expressed in WSLA^[21] – an XML-based standard

- XML is the basic format used in enterprise distributed computing system
 - Web Services^[17] use SOAP^[18, 19] – an XML-based standard
 - Services are described using WSDL^[20] – an XML-based standard
 - QoS criteria of services are expressed in WSLA^[21] – an XML-based standard
 - Services can be composed using WS-BPEL^[22–24] or WS-CDL^[25] – both XML-based standards

- XML is the basic format used in enterprise distributed computing system
 - Web Services^[17] use SOAP^[18, 19] – an XML-based standard
 - Services are described using WSDL^[20] – an XML-based standard
 - QoS criteria of services are expressed in WSLA^[21] – an XML-based standard
 - Services can be composed using WS-BPEL^[22–24] or WS-CDL^[25] – both XML-based standards
 - XHTML^[7] is an XML format
 - ...

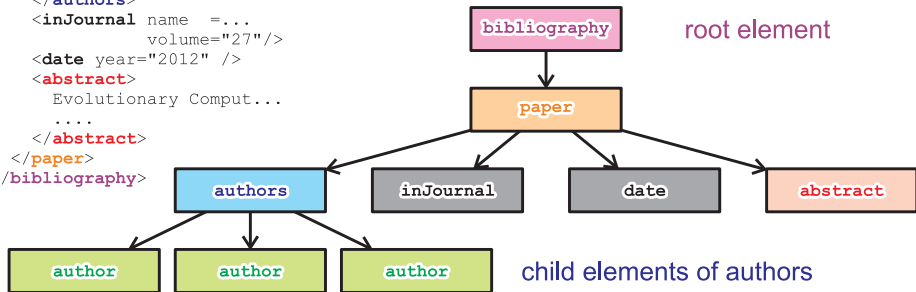
- XML is the basic format used in enterprise distributed computing system
 - Web Services^[17] use SOAP^[18, 19] – an XML-based standard
 - Services are described using WSDL^[20] – an XML-based standard
 - QoS criteria of services are expressed in WSLA^[21] – an XML-based standard
 - Services can be composed using WS-BPEL^[22–24] or WS-CDL^[25] – both XML-based standards
 - XHTML^[7] is an XML format
 - ...
- ... but more about this in a later lesson

- Tags allow for the structured representation of contents using XML

```
<?xml version="1.0"?>
<bibliography>
  <paper title="Evolutionary Optimization: Pitfalls and Booby Traps"
        type="article">
    <authors>
      <author firstName="Thomas"  familyName="Weise" />
      <author firstName="Raymond" familyName="Chiong" />
      <author firstName="Tang"     familyName="Ke"   />
    </authors>
    <inJournal name="Journal of Computer Science and Technology"
              volume="27"/>
    <date year="2012" />
    <abstract>
      Evolutionary Computation (EC),...
      ....
    </abstract>
  </paper>
</bibliography>
```

- Tags allow for the structured representation of contents using XML

```
<?xml version="1.0"?>
<bibliography>
  <paper title="Evolutio...
    type ="article">
    <authors>
      <author firstName=...
      <author firstName=...
      <author firstName=...
    </authors>
    <inJournal name =...
      volume="27"/>
    <date year="2012" />
    <abstract>
      Evolutionary Comput...
      ....
    </abstract>
  </paper>
</bibliography>
```



Element name/type:
start

element contents

attribute name

attribute value

```
<?xml version="1.0"?>
<bibliography>
  <paper title="Evolutionary Optimization: Pitfalls and Booby Traps"
    type="article">
    <authors>
      <author firstName="Thomas" familyName="Weise" />
      <author firstName="Raymond" familyName="Chiong" />
      <author firstName="Tang" familyName="Ke" />
    </authors>
    <inJournal name="Journal of Computer Science and Technology"
      volume="27"/>
    <date year="2012" />
    <abstract>
      Evolutionary Computation (EC), ...
      ....
    </abstract>
  </paper>
</bibliography>
```

Element name/type: end

text contents

- Two complementary basic concepts: Element and Attribute

- Two complementary basic concepts: Element and Attribute
- Element

- Two complementary basic concepts: Element and Attribute
- Element
 - node in a tree

- Two complementary basic concepts: Element and Attribute
- Element
 - node in a tree
 - can be empty or contain sub-tree (i.e., child-elements)

- Two complementary basic concepts: Element and Attribute
- Element
 - node in a tree
 - can be empty or contain sub-tree (i.e., child-elements)
 - child nodes may occur multiple times

- Two complementary basic concepts: Element and Attribute
- Element
 - node in a tree
 - can be empty or contain sub-tree (i.e., child-elements)
 - child nodes may occur multiple times
 - represents a structural part of a document

- Two complementary basic concepts: Element and Attribute
- Element
 - node in a tree
 - can be empty or contain sub-tree (i.e., child-elements)
 - child nodes may occur multiple times
 - represents a structural part of a document
- Attribute

- Two complementary basic concepts: Element and Attribute
- Element
 - node in a tree
 - can be empty or contain sub-tree (i.e., child-elements)
 - child nodes may occur multiple times
 - represents a structural part of a document
- Attribute
 - belong to an element

- Two complementary basic concepts: Element and Attribute
- Element
 - node in a tree
 - can be empty or contain sub-tree (i.e., child-elements)
 - child nodes may occur multiple times
 - represents a structural part of a document
- Attribute
 - belong to an element
 - only once per element

- Two complementary basic concepts: Element and Attribute
- Element
 - node in a tree
 - can be empty or contain sub-tree (i.e., child-elements)
 - child nodes may occur multiple times
 - represents a structural part of a document
- Attribute
 - belong to an element
 - only once per element
 - contents not structures, only simple strings

- Two complementary basic concepts: Element and Attribute
- Element
 - node in a tree
 - can be empty or contain sub-tree (i.e., child-elements)
 - child nodes may occur multiple times
 - represents a structural part of a document
- Attribute
 - belong to an element
 - only once per element
 - contents not structures, only simple strings
 - represents a relation between an element and additional information

- Element or attribute names may have different meaning

- Element or attribute names may have different meaning:
 - `<element>` in chemistry may refer to a chemical element
 - `<element>` can refer to a geometric object

- Element or attribute names may have different meaning:
 - `<element>` in chemistry may refer to a chemical element
 - `<element>` can refer to a geometric object
- XML namespaces provide clarity

- Element or attribute names may have different meaning:
 - `<element>` in chemistry may refer to a chemical element
 - `<element>` can refer to a geometric object
- XML namespaces provide clarity
- A namespace defines a set of element and attribute names for a given context

- Element or attribute names may have different meaning:
 - `<element>` in chemistry may refer to a chemical element
 - `<element>` can refer to a geometric object
- XML namespaces provide clarity
- A namespace defines a set of element and attribute names for a given context
- A namespace is identified by an URI

- Element or attribute names may have different meaning:
 - `<element>` in chemistry may refer to a chemical element
 - `<element>` can refer to a geometric object
- XML namespaces provide clarity
- A namespace defines a set of element and attribute names for a given context
- A namespace is identified by an URI
- Format: `<namespaceUri:elementName ...>` and
`... namespaceUri:attributeName=...`

- Mechanism for globally unique names

- Mechanism for globally unique names

```
<h:html xmlns:xdc="http://www.xml.com/books"
        xmlns:h="http://www.w3.org/1999/xhtml">
  <h:head>
    <h:title>Book Review</h:title>
  </h:head>
  ...
  <xdc:bookreview>
    <xdc:title>XML: A Primer</xdc:title>
    ...
  </xdc:bookreview>
  ...
</h:html>
```

confusion prevented by using namespaces

- Mechanism for globally unique names

```
<h:html xmlns:xdc="http://www.xml.com/books"
        xmlns:h="http://www.w3.org/1999/xhtml">
  <h:head>
    <h:title>Book Review</h:title>
  </h:head>
  ...
  <xdc:bookreview>
    <xdc:title>XML: A Primer</xdc:title>
    ...
  </xdc:bookreview>
  ...
</h:html>
```

confusion prevented by using namespaces

- Namespaces only identify “vocabulary”

- Mechanism for globally unique names

```
<h:html xmlns:xdc="http://www.xml.com/books"
        xmlns:h="http://www.w3.org/1999/xhtml">
  <h:head>
    <h:title>Book Review</h:title>
  </h:head>
  ...
  <xdc:bookreview>
    <xdc:title>XML: A Primer</xdc:title>
    ...
  </xdc:bookreview>
  ...
</h:html>
```

confusion prevented by using namespaces

- Namespaces only identify “vocabulary”
- Additional mechanisms necessary to define tag structure and semantics

- An XML document is called “well-formed”, if it

- An XML document is called “well-formed”, if it
 - starts with an XML declaration (`<?xml version="1.0" [...] ?>` , currently either `1.0` ^[4] or `1.1` ^[5])

- An XML document is called “well-formed”, if it
 - starts with an XML declaration (`<?xml version="1.0" [...] ?>` , currently either `1.0` ^[4] or `1.1` ^[5])
 - contains at least one element

- An XML document is called “well-formed”, if it
 - starts with an XML declaration (`<?xml version="1.0" [...] ?>` , currently either `1.0` ^[4] or `1.1` ^[5])
 - contains at least one element
 - has exactly one (root) element that contains all other elements as sub-trees

- An XML document is called “well-formed”, if it
 - starts with an XML declaration (`<?xml version="1.0" [...] ?>` , currently either `1.0` ^[4] or `1.1` ^[5])
 - contains at least one element
 - has exactly one (root) element that contains all other elements as sub-trees
 - all elements are nested properly, i.e., `<a> ` and not `<a> `

- An XML document is called “well-formed”, if it
 - starts with an XML declaration (`<?xml version="1.0" [...] ?>` , currently either `1.0` ^[4] or `1.1` ^[5])
 - contains at least one element
 - has exactly one (root) element that contains all other elements as sub-trees
 - all elements are nested properly, i.e., `<a> ` and not `<a> `
 - all attributes are contained within quotation marks

- An XML document is called “well-formed”, if it
 - starts with an XML declaration (`<?xml version="1.0" [...] ?>` , currently either `1.0` ^[4] or `1.1` ^[5])
 - contains at least one element
 - has exactly one (root) element that contains all other elements as sub-trees
 - all elements are nested properly, i.e., `<a> ` and not `<a> `
 - all attributes are contained within quotation marks
 - XML language elements and characters not belonging to the declared character set are specified as character entities (e.g., `<` as `<` ;)

- An XML document is called “well-formed”, if it
 - starts with an XML declaration (`<?xml version="1.0" [...] ?>` , currently either `1.0` ^[4] or `1.1` ^[5])
 - contains at least one element
 - has exactly one (root) element that contains all other elements as sub-trees
 - all elements are nested properly, i.e., `<a> ` and not `<a> `
 - all attributes are contained within quotation marks
 - XML language elements and characters not belonging to the declared character set are specified as character entities (e.g., `<` as `<` ;)
 - More info: see grammars in ^[4, 5]

- Important!

- Important!
 - No restrictions on element names, even for the root element

- Important!
 - No restrictions on element names, even for the root element
 - All elements must be closed properly
(empty elements: `<example attr="bla"/>`)

- Important!
 - No restrictions on element names, even for the root element
 - All elements must be closed properly
(empty elements: `<example attr="bla"/>`)
- A well-formed document does not necessary need to “make sense”

- Important!
 - No restrictions on element names, even for the root element
 - All elements must be closed properly
(empty elements: `<example attr="bla"/>`)
- A well-formed document does not necessary need to “make sense”

```
<?xml version="1.1">
<time color="fish">
  <university price="blue" />
  <employees>
    <fish name="car" />
    <shoe numer="salty">
  </employees>
</time>
```

Which elements are allowed?
Which attributes are allowed?
Which values are allowed?
How can elements be nested?
...

- Important!
 - No restrictions on element names, even for the root element
 - All elements must be closed properly
(empty elements: `<example attr="bla"/>`)
- A well-formed document does not necessary need to “make sense”
- For this, we need the document type or schema

```
<?xml version="1.1">
<time color="fish">
  <university price="blue" />
  <employees>
    <fish name="car" />
    <shoe numer="salty">
  </employees>
</time>
```

Which elements are allowed?
Which attributes are allowed?
Which values are allowed?
How can elements be nested?
...

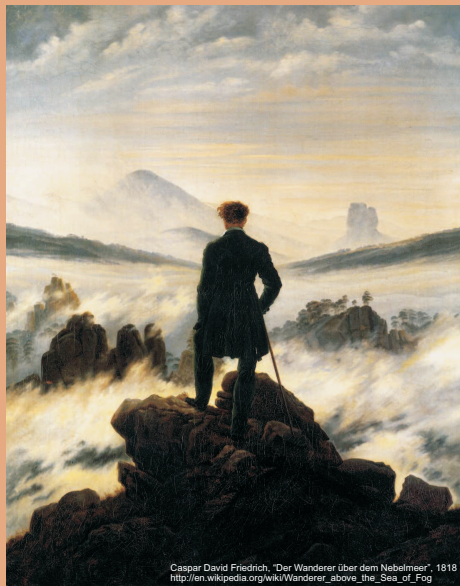
- XML is the predominant data format in internet, basis for XHTML ^[7] and many protocols
- XML is self-describing and easy to read

谢谢

Thank you

Thomas Weise [汤卫思]
tweise@hfu.edu.cn
<http://www.it-weise.de>

Hefei University, South Campus 2
Institute of Applied Optimization
Shushan District, Hefei, Anhui,
China



Caspar David Friedrich, "Der Wanderer über dem Nebelmeer", 1818
http://en.wikipedia.org/wiki/Wanderer_above_the_Sea_of_Fog



1. [MS-DOC]: *Word (.doc) Binary File Format*. Redmond, WA, USA: Microsoft Corporation, v20120122 edition, January 22, 2012. URL <http://msdn.microsoft.com/en-us/library/cc313153.aspx>.
2. Leslie Lamport. *LaTeX: A Document Preparation System. User's Guide and Reference Manual*. Reading, MA, USA: Addison-Wesley Publishing Co. Inc., 1994. ISBN 0201529831 and 9780201529838. URL <http://books.google.de/books?id=19pzdWEECAAJ>.
3. Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Tools and Techniques for Computer Typesetting. Reading, MA, USA: Addison-Wesley Publishing Co. Inc., 1994. ISBN 0201541998 and 9780201541991. URL <http://books.google.de/books?id=54A3MuBzIrEC>.
4. Tim Bray, Jean Paoli, C. Michael Sperberg-McQueen, Eve Maler, and François Yergeau. *Extensible Markup Language (XML) 1.0 (Fourth Edition)*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), September 27, 2007. URL <http://www.w3.org/TR/2006/REC-xml-20060816>. W3C Recommendation 16 August 2006, edited in place 29 September 2006.
5. Tim Bray, Jean Paoli, C. Michael Sperberg-McQueen, Eve Maler, François Yergeau, and John Cowan. *Extensible Markup Language (XML) 1.1 (Second Edition)*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), August 16–September 29, 2006. URL <http://www.w3.org/TR/2006/REC-xml11-20060816>.
6. Dave Raggett, Arnaud Le Hors, and Ciel J. H. Jacobs. *HTML 4.01 Specification*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), December 24, 1999. URL <http://www.w3.org/TR/1999/REC-html401-19991224>.
7. Murray Altheim and Shane McCarron. *XHTML™ 1.1 – Module-based XHTML – Second Edition*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), November 23, 2010. URL <http://www.w3.org/TR/2010/REC-xhtml11-20101123>.
8. *ISO 8879:1986 Information processing – Text and office systems – Standard Generalized Markup Language (SGML)*. Geneva, Switzerland: International Organization for Standardization (ISO), October 1, 1986.
9. Henry S. Thompson, David Beech, Murray Maloney, and Noah Mendelsohn, editors. *XML Schema Part 1: Structures Second Edition*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), October 28, 2004. URL <http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/>.
10. Paul V. Biron and Ashok Malhotra, editors. *XML Schema Part 2: Datatypes Second Edition*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), October 28, 2004.

11. Cliff Binstock. *The XML Schema – Complete Reference*. Reading, MA, USA: Addison-Wesley Professional, 2003. ISBN 0672323745 and 9780672323744. URL <http://books.google.de/books?id=pCLJly0eZwEC>.
12. Marco Skulschus and Marcus Wiederstein. *XML Schema*. Berlin, Germany: Comelio GmbH, 2008. ISBN 3898424723, 393970122X, 9783898424721, and 9783939701224. URL <http://books.google.de/books?id=Vi6dTul6g7kC>.
13. Eric Van der Vlist. *XML Schema – The W3C's Object-Oriented Descriptions for XML*. O'Reilly Series. Sebastopol, CA, USA: O'Reilly Media, Inc., 2002. ISBN 0596002521 and 9780596002527. URL <http://books.google.de/books?id=2h8T6xqKeVUC>.
14. Jack Lindsey. Subtyping in w3c xml schema, part 1 – three degrees of inheritance. *The Data Administration Newsletter – TDAN.com*, 12(4), April 3, 2008. URL <http://www.tdan.com/view-articles/7185>.
15. James Clark. *XSL Transformations (XSLT)*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), version 1 edition, November 16, 1999. URL <http://www.w3.org/TR/1999/REC-xslt-19991116>.
16. Anders Berglund, editor. *Extensible Stylesheet Language (XSL) Version 1.1*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), December 5, 2006. URL <http://www.w3.org/TR/2006/REC-xsl11-20061205/>.
17. G. Radhamani and Udaya Bhaskara Rao N. *Web Services Security and E-Business*. Gale Virtual Reference Library. New York, NY, USA: Idea Group Publishing (Idea Group Inc., IGI Global), 2007. ISBN 1599041685 and 978-1599041681. URL <http://books.google.de/books?id=mkxgEsKb16oC>.
18. Nilo Mitra and Yves Lafon. *SOAP Version 1.2 Part 0: Primer (Second Edition)*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), April 27, 2007. URL <http://www.w3.org/TR/2007/REC-soap12-part0-20070427/>.
19. Martin Gudgin, Marc Hadley, Noah Mendelsohn, Jean-Jacques Moreau, Henrik Frystyk Nielsen, Anish Karmarkar, and Yves Lafon. *SOAP Version 1.2 Part 2: Adjuncts (Second Edition)*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), April 27, 2007. URL <http://www.w3.org/TR/2007/REC-soap12-part2-20070427/>.
20. David Booth and Canyang Kevin Liu. *Web Services Description Language (WSDL) Version 2.0 Part 0: Primer*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), June 26, 2007. URL <http://www.w3.org/TR/2007/REC-wsd120-primer-20070626>.

21. Heiko Ludwig, Alexander Keller, Asit Dan, Richard P. King, and Richard Franck. *Web Service Level Agreement (WSLA) Language Specification*. Armonk (North Castle), NY, USA: International Business Machines Corporation (IBM), January 28, 2003. URL <http://www.research.ibm.com/wsla/WSLASpecV1-20030128.pdf>. Version 1, Revision wsla-2003/01/28.
22. Diane Jordan, John Evdemon, Alexandre Alves, Assaf Arkin, Sid Askary, Charlton Barreto, Ben Bloch, Francisco Curbera, Mark Ford, Yaron Goland, Alejandro Guízar, Neelakantan Kartha, Canyang Kevin Liu, Rania Khalaf, Dieter König, Mike Marin, Vinkesh Mehta, Satish Thatte, Danny van der Rijn, Prasad Yendluri, and Alex Yiu. *Web Services Business Process Execution Language Version 2.0*. OASIS Standard. Burlington, MA, USA: Organization for the Advancement of Structured Information Standards (OASIS), April 11, 2007. URL <http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf>. Technical Committee: OASIS Web Services Business Process Execution Language (WSBPEL) TC.
23. Tony Andrews, Francisco Curbera, Hitesh Dholakia, Yaron Goland, Johannes Klein, Frank Leymann, Kevin Liu, Dieter Roller, Doug Smith, Satish Thatte, Ivana Trickovic, and Sanjiva Weerawarana. *BPEL4WS, Business Process Execution Language for Web Services Version 1.1*. San José, CA, USA: BEA Systems, Inc., Armonk (North Castle), NY, USA: International Business Machines Corporation (IBM), Redmond, WA, USA: Microsoft Corporation, Walldorf (Baden), Germany: SAP AG, and San Mateo, CA, USA: Siebel Systems, May 5, 2003. URL <http://www.ibm.com/developerworks/library/specification/ws-bpel/>.
24. *SOA Best Practices: The BPEL Cookbook*. URL http://www.oracle.com/technology/pub/articles/bpel_cookbook/.
25. Nickolas Kavantzaz, David Burdett, Gregory Ritzinger, Tony Fletcher, and Yves Lafon. *Web Services Choreography Description Language Version 1.0 – W3C Candidate Recommendation*. W3C Recommendation. MIT/CSAIL (USA), ERCIM (France), Keio University (Japan): World Wide Web Consortium (W3C), November 9, 2005. URL <http://www.w3.org/TR/2005/CR-ws-cd1-10-20051109/>.