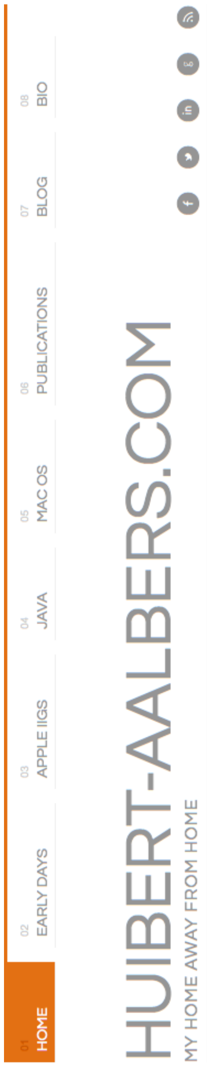
Introduction to XML - Validation & Parsing

Huibert Aalbers
Senior Certified Software IT Architect

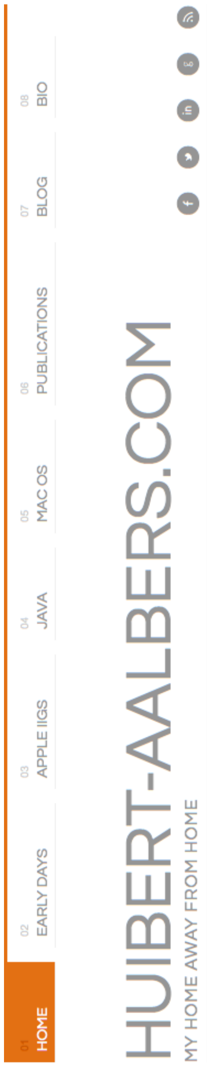


IT Insight podcast

- This podcast belongs to the IT Insight series
  You can subscribe to the podcast through iTunes.

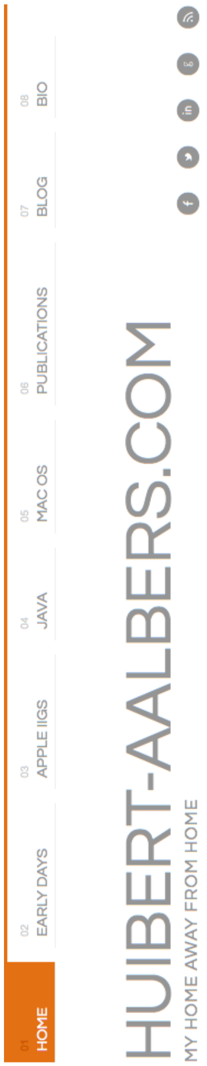
- Additional material such as presentations in PDF format or white papers mentioned in the podcast can be downloaded from the IT insight section of my site at http://www.huibert-aalbers.com

- You can send questions or suggestions regarding this podcast to my personal email, huibert_aalbers@mac.com



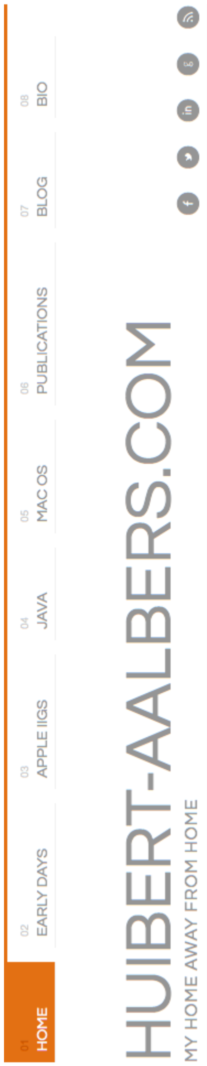
Introduction to XML

- XML stands for eXtensible Markup Language
- XML resulted from a proposal to simplify SGML
- XML is a data description language
- The data structure of an XML file can be validated by linking it to a DTD or XML-Schema file
- XML files in conjunction with their DTD or XML- Schema do not require additional description
- XML is based on recommendations from the W3C



Introduction to XML

- XML is free and extensible
- Anyone can create a new XML document type to represent any kind of data using their own tags and rules.
- Many standard XML documents have already been defined.
- MathML (a standard way to represent mathematical expressions)
- SVG (Scalable Vector graphics) TaxML,etc.
- Before creating a new XML document type you should first check if a standard has not already been defined



Introduction to XML

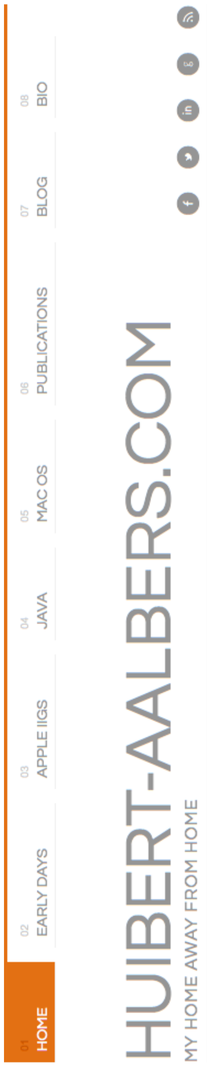
- A valid XML document must be well formed
  - It has to begin with a standard header
    - <?xml version=”1.0” encoding=”ISO-8859-1”?>
  - All tags must be closed
    - <person><name>Huibert</name></person> is valid
    - <person><name>Huibert</person> is not valid
    - <age value=”15” /> is valid
- Tags must be properly nested
  - <person><name>Huibert</person></name> is not valid

Introduction to XML

- A valid XML document must be well formed
    - A document can only a single root element
    - Attributes must be surrounded by quotes (single or double)
        - <age value=15 /> is not valid
    - Tags are case sensitive
        - <Person><name>Huibert</name></person> is not valid
    - White space is preserved
        - Unlike HTML where multiple space characters are treated as a single one
    - You can insert comments in a XML file
        - <!- This is a comment -->

Introduction to XML

- XML is extensible
    - If the structure of the XML document is changed to hold more information, well written applications that use the document should not be affected.
- XML elements have relationships
    - Elements are related as parents and children



Relationship between XML and HTML

- HTML is not a type of XML
    - It is possible to create valid HTML documents which are not valid XML documents
    - This is a major problem since there is no standard way to validate HTML documents and therefore even incorrect HTML documents are rendered
    - Each browser tries to represent invalid HTML documents the best they can and this leads to inconsistent results
- xHTML is a new way to describe HTML documents based on XML
    - All current browsers support xHTML

XML document validation

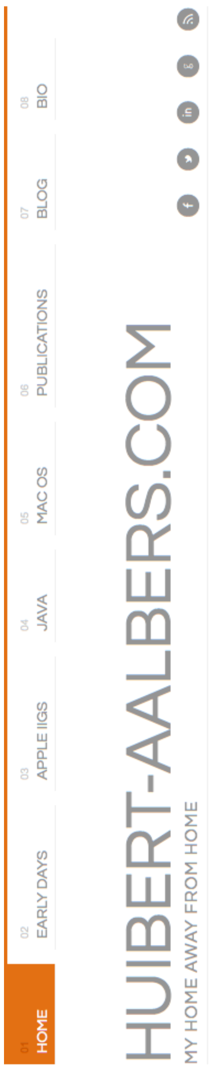
- XML documents way be well formed but not valid if their structure does not match the one defined in their DTD or XML-Schema
- XML document validation is performed by an XML parser
  - Java programmers normally use Xerces, based on XML4J, the first Java parser, created by IBM
  - Xerces contains two types of parsers (SAX and DOM)
- The use of DTDs (Document Type Definition) is being strongly discouraged
  - Because it does not provide strong data typing DTDs are not based on XML
  - DTDs are not based on XML

SAX vs DOM

- SAX stands for Simple API for XML processing
  - The whole document is read and different events are generated during the process
  - SAX is mainly used to quickly process a document that does not need to be modified
- DOM stands for Document Object Model
  - The whole XML document is loaded into memory
  - The API offers methods to add, delete or modify nodes and leafs
  - Unlike SAX, DOM allows programmers to navigate the document in any direction
  - The increased power of DOM comes at a larger cost in memory

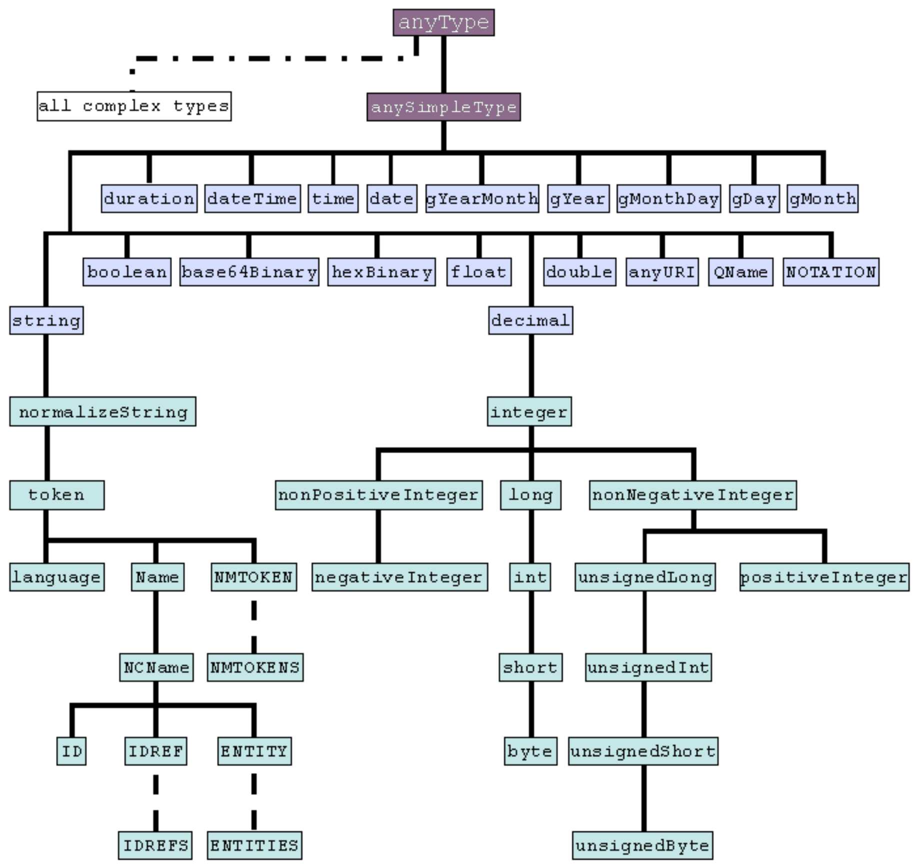
Introduction to XML-Schema

- It is an alternative to DTD, based on XML
- It is also known as XML-Schema Definition (xsd)
- It is easier to use XML-Schema in conjunction with relational databases
  - In general it is easy to convert a database schema, including constraints into a XSD file
- XML-Schema allows the definition of new types
  - Based on existing types
    - Through restriction, extension and inheritance
  - New types

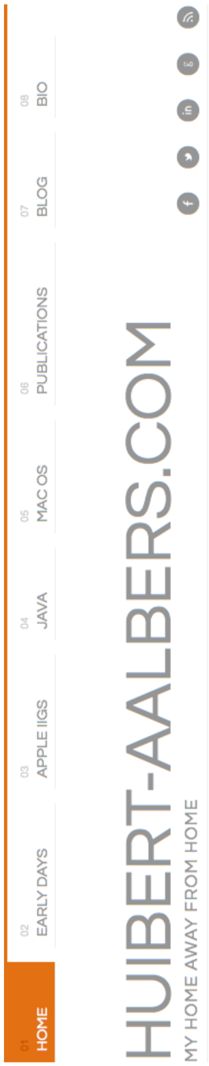


HUIBERT-AALBERS.COM
MY HOME AWAY FROM HOME
01 HOME
02 EARLY DAYS
03 APPLE IIGS
04 JAVA
05 MAC OS
06 PUBLICATIONS
07 BLOG
08 BIO

XML Schema



anyType
all complex types
anySimpleType
duration
dateTime
time
date
gYearMonth
gYear
gMonthDay
gDay
gMonth
boolean
base64Binary
hexBinary
float
double
anyURI
QName
NOTATION
string
decimal
normalizeString
integer
token
nonPositiveInteger
long
nonNegativeInteger
language
Name
NMTOKEN
negativeInteger
int
unsignedLong
positiveInteger
NCName
NMTOKENS
short
unsignedInt
ID
IDREF
ENTITY
byte
unsignedShort
IDREFS
ENTITIES
unsignedByte



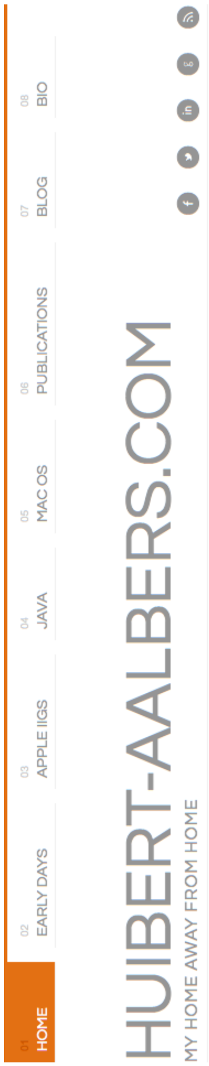
Introduction to XML-Schema

Imagine the following XML file representing a message:

```
<?xml version="1.0"?>
<message>
  <to>John</to>
  <from>Elizabeth</from>
  <title>Do not forget</title>
  <body>Do not forget to attend the WebSphere user group meeting</body>
</message>
```

We could validate it with the following DTD:

```
<!ELEMENT message (to, from, title, body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT titulo (#PCDATA)>
<!ELEMENT body (#PCDATA)>
```

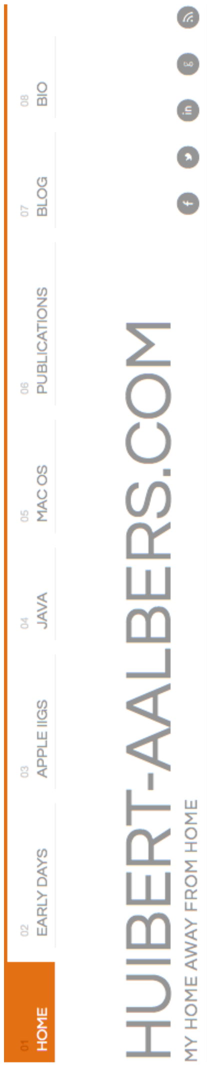


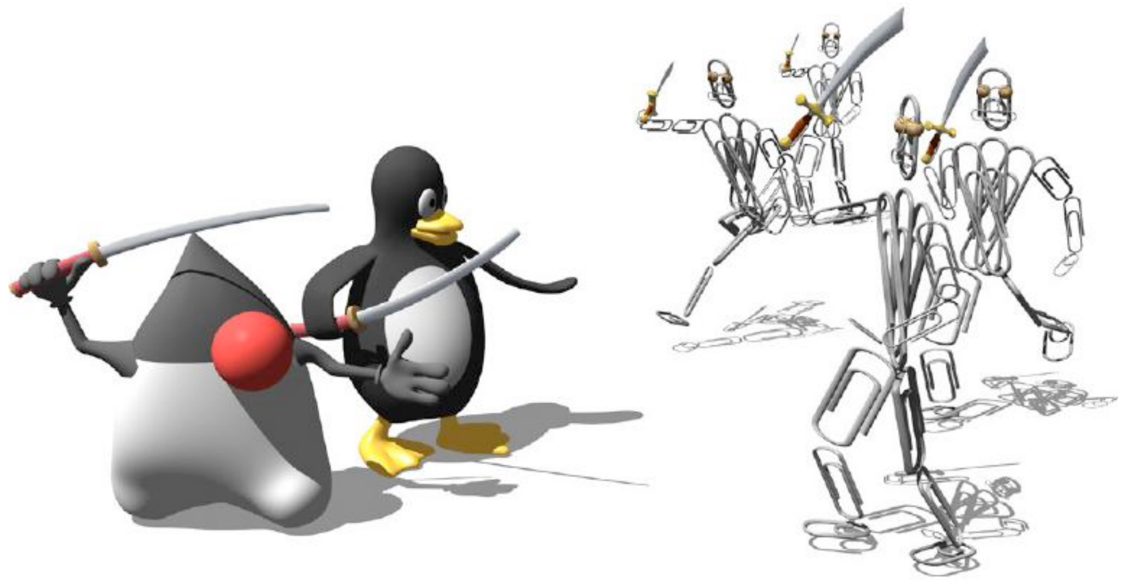
Introduction to XML-Schema

With XML-Schema we would use the following file:

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.ibm.com"
xmlns="http://www.ibm.com" elementFormDefault="qualified">

<xs:element name="message">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="to" type="xs:string"/>
        <xs:element name="from" type="xs:string"/>
        <xs:element name="title" type="xs:string"/>
        <xs:element name="body" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>
```





For more information, please contact me at
huibert_aalbers@mac.com