

Intro to XML

Carl Swaney
MITRE

With special thanks from:



MITRE

A STAR ALLIANCE MEMBER



Agenda

1. Mark-up Languages
2. What is XML?
3. When can XML be used?
4. XML Syntax
5. XML Elements
6. XML Attributes
7. XML Elements vs Attributes
8. XML DTD/Schema
9. XML Validation
10. Other Subjects
11. Further Reading



What is XML?

- What is XML?

- XML stands for:

EXtensible **M**arkup **L**anguage

- Uses a Document Type Definition (DTD) or an XML Schema
 - XML with a DTD or XML Schema is designed to be self-descriptive

© AIRBUS S.A. All rights reserved. Confidential and proprietary document.

Introduction to XML

June 2005

Page 4



Mark-up Languages

- 3 core Mark-up Languages

- Standardised General Mark-up Language (SGML)
 - Extremely powerful and complex
 - With many unused features
- Hypertext Mark-up Language (HTML)
 - Designed for displaying data - to focus on how data looks
 - The language of the World Wide Web (WWW)
- Extensible Mark-up Language (XML)
 - Designed for describing data - to focus on the data content

- Many derivatives developed for specific industries

- Medical, financial, automotive, telecommunication, aerospace, etc.

© AIRBUS S.A. All rights reserved. Confidential and proprietary document.

Introduction to XML

June 2005

Page 3



When can XML be used?

- XML is:
 - Cross-platform
 - Software and hardware independent
 - For information exchange
- Data can be exchanged between incompatible systems
 - In the real world, computer systems and databases contain data in incompatible formats
 - One of the most time-consuming challenges for developers has been to exchange data between such systems over the Internet
 - Converting the data to XML can greatly reduce this complexity and create data that can be read by many different types of applications



When can XML be used?

- XML data is stored in plain text format, so it is:
 - Easier to create data for different applications to understand
 - Easier to expand or upgrade a system to new operating systems, servers, applications, and new browsers
- XML can also be used to store data in files or in databases
 - Applications can be written to store and retrieve information from the store
 - Generic applications can be used to display the data
- Other clients and applications can access your XML files as data sources, like they are accessing databases
- Data can be made available to all kinds of "reading machines" (applications)



XML Syntax

- The syntax rules of XML are relatively simple but very strict
- The rules are “easy” to learn and “easy” to use
- Therefore, creating software that can read and manipulate XML is “very easy” to do

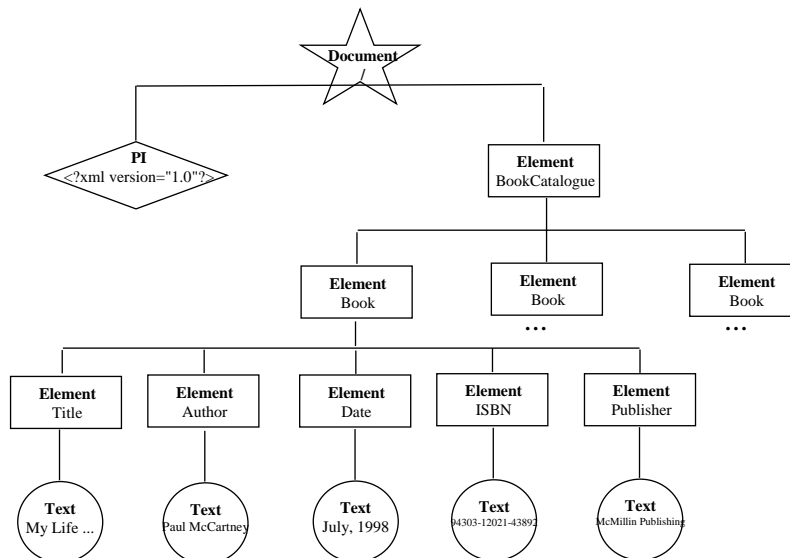
© 2002 Airbus SAS. All rights reserved. Confidential and proprietary document.

Introduction to XML

June 2005 Page 9



XML Document Tree



Complete XML Document for the Book Catalogue

```
<?xml version="1.0"?>
<BookCatalogue>
  <Book>
    <Title>My Life and Times</Title>
    <Author>Paul McCartney</Author>
    <Date>July, 1998</Date>
    <ISBN>94303-12021-43892</ISBN>
    <Publisher>McMillin Publishing</Publisher>
  </Book>
  <Book>
    <Title>Illusions The Adventures of a Reluctant Messiah</Title>
    <Author>Richard Bach</Author>
    <Date>1977</Date>
    <ISBN>0-440-34319-4</ISBN>
    <Publisher>Dell Publishing Co.</Publisher>
  </Book>
  <Book>
    <Title>The First and Last Freedom</Title>
    <Author>J. Krishnamurti</Author>
    <Date>1954</Date>
    <ISBN>0-06-064831-7</ISBN>
    <Publisher>Harper & Row</Publisher>
  </Book>
</BookCatalogue>
```

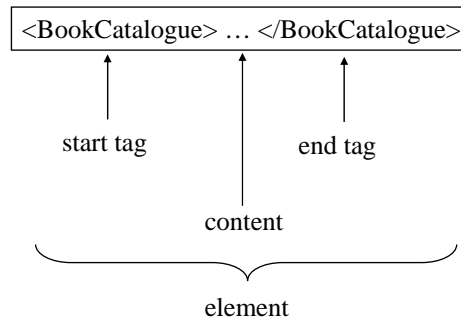
BookCatalogue.xml

The root element. Every XML document must have one element that wraps everything else. That element is called the root element.

```
<?xml version="1.0"?>
<BookCatalogue>
  <Book>
    <Title>My Life and Times</Title>
    <Author>Paul McCartney</Author>
    <Date>July, 1998</Date>
    <ISBN>94303-12021-43892</ISBN>
    <Publisher>McMillin Publishing</Publisher>
  </Book>
  <Book>
    <Title>Illusions The Adventures of a Reluctant Messiah</Title>
    <Author>Richard Bach</Author>
    <Date>1977</Date>
    <ISBN>0-440-34319-4</ISBN>
    <Publisher>Dell Publishing Co.</Publisher>
  </Book>
  <Book>
    <Title>The First and Last Freedom</Title>
    <Author>J. Krishnamurti</Author>
    <Date>1954</Date>
    <ISBN>0-06-064831-7</ISBN>
    <Publisher>Harper & Row</Publisher>
  </Book>
</BookCatalogue>
```

BookCatalogue.xml

Terminology



XML Elements

- XML Elements are Extensible - documents can be extended to carry more information

▸ Look at the following XML example:

```
<note>
  <to>Jean-yves</to>
  <from>Vincent</from>
  <body>Don't forget the rugby match this weekend!</body>
</note>
```

▸ An application extracts the <to>, <from>, and <body> elements from the XML document to produce this output:

```
MESSAGE
To: Jean-yves
From: Vincent
Don't forget the rugby match this weekend!
```

XML Elements

- Adding elements to an XML document is possible:

```
<note>  
  <date>2002-08-01</date>  
  <to>Jean-yves</to>  
  <from>Vincent</from>  
  <heading>Reminder</heading>  
  <body>Don't forget the rugby match this weekend!</body>  
</note>
```

- The same application should still be able to find the <to>, <from>, and <body> elements in the XML document and produce the same output

- Therefore, XML documents are Extensible

- Another application could extract the additional elements or, alter which elements are displayed
- Or the existing application could be developed further to recognise the latest data available

© AIRBUS S.A. All rights reserved. Confidential and proprietary document.

Introduction to XML

June 2005 Page 20



Terminology - Attributes

```
<BookCatalogue CreateDate="1 Feb 2006"> ... </BookCatalogue>
```

↑
start tag

↑
Attribute

Attributes are data about data. This is also known as METADATA

Motivation for Attributes

Suppose that you see an XML document which contains:

```
<altitude>12000</altitude>
```

What does 12000 mean? 12000 what? 12000 feet? 12000 meters? Etc.

We would like to provide some extra information about the contents of altitude (namely, the units). We would like to provide some data about the data contained in <altitude>, i.e., metadata about the data. That's the purpose of attributes!

Attributes

Attributes are bundled in with the start tag:

```
      units="feet"  
    ↙  
<altitude>12000</altitude>
```

Thus, we have:

```
<altitude units="feet">12000</altitude>
```

"The tag called altitude contains the data 12000. The units of that data is feet. In other words, altitude contains 12000 feet."

Attributes

```
<altitude units="feet">12000</altitude>
```

12000 feet - with respect to what? 12000 feet Above Ground Level (AGL), or 12000 feet Mean Sea Level (MSL)? We need more metadata if we wish to fully describe what 12000 means.

Indeed, we can have any number of attributes (**and order is irrelevant**):

```
<altitude units="feet" reference="AGL"> 12000</altitude>
```

Note that attributes are separated by white space.

XML Elements vs Attributes

• Use of Elements vs. Attributes

- ▶ Data can be stored in child elements or in attributes

- ▶ Take a look at these examples:

```
<person sex="female">  
<firstname>Anna</firstname>  
<lastname>Smith</lastname>  
</person>
```

```
<person>  
<sex>female</sex>  
<firstname>Anna</firstname>  
<lastname>Smith</lastname>  
</person>
```

- ▶ Both examples provide the same information
- ▶ There are no rules about when to use attributes/elements
- ▶ Use child elements if the information feels like data

XML DTD/Schema

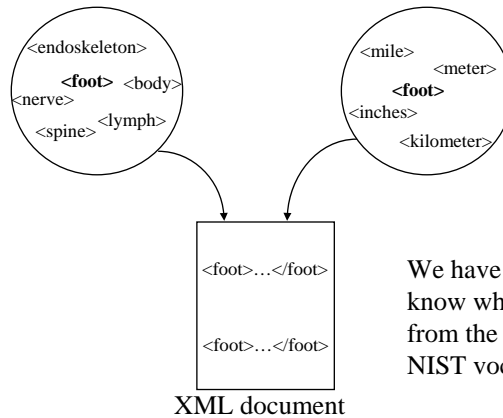
- XML DTD/Schema
 - The purpose of a DTD/Schema is to define the legal building blocks of an XML document
 - It defines the document structure with a list of legal elements
- XML Schema is the XML based alternative to DTD

XML Validation

- "Well Formed" XML
 - is a document that conforms to the XML syntax rules
- "Valid" XML
 - is a "Well Formed" XML document, which also conforms to the rules of a Document Type Definition (DTD), e.g.

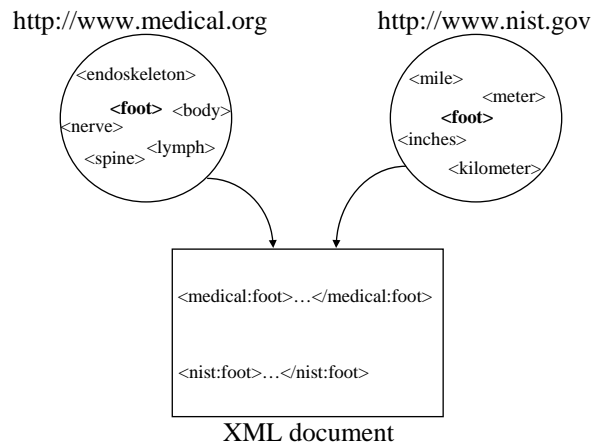
```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE note SYSTEM "InternalNote.dtd">
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```

Medical and NIST XML Vocabularies



We have an ambiguity - we don't know whether a <foot> tag came from the Medical vocabulary or the NIST vocabulary.

Resolving the Ambiguity - give a "label" to each XML Vocabulary



Purpose of Namespaces

- Namespaces enables disambiguation by extending the XML syntax to allow element names to be *qualified* with a name.

```
Alias pub with http://www.publishing.org
Alias med with http://www.medical.org
<pub:Chapter>
  <pub:Body>
    <med:Body> ... </med:Body>
  </pub:Body>
</pub:Chapter>
```

What does an XML Schema accomplish?

```
Declare a location element. Require that its content
be latitude, longitude, and uncertainty.
Declare a latitude element. Require that its value be
between -90 and +90.
Declare a longitude element. Require that its value be
between -180 and +180.
Declare a uncertainty element with a units attribute.
Require that the element's value be between 0 and 10.
Require that the attribute's value be either feet or meters.
```

XML Schema

Answer:

It creates an XML vocabulary:

<location>, <latitude>, <longitude>, <uncertainty>

It specifies the contents of each element, and the restrictions on the content.

It does one more thing ...

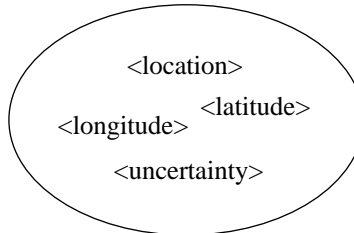
Namespace = <http://www.example.org/target>
Declare a location element. Require that its content be latitude, longitude, and uncertainty.
Declare a latitude element. Require that its value be between -90 and +90.
Declare a longitude element. Require that its value be between -180 and +180.
Declare a uncertainty element with a units attribute. Require that the element's value be between 0 and 10. Require that the attribute's value be either feet or meters.

XML Schema

An XML Schema specifies that the XML vocabulary that is being created shall be in a "namespace"

The <http://www.example.org/target> Namespace

<http://www.example.org/target>



DoD XML Registry - XML Registry - Netscape

File Edit View Go Communicator Help

http://diides.ncr.disa.mil/xmlreg/user/index.cfm

Back Forward Reload Home Search Guide Print Security Stop

Bookmarks Netsite: http://diides.ncr.disa.mil/xmlreg/user/index.cfm

Instant Message WebMail Calendar Radio People Yellow Pages Download Customize... RealPlayer

DoD METADATA REGISTRY AND CLEARINGHOUSE

DoD XML Registry **your login access**

[Login](#)

XML Registry

Search query for schemas

XML registries are a vital component in the implementation of shared data exchanges. Developers looking to express information using XML need support in establishing common lexicons and grammars. This Registry enables the consistent use of XML, both vertically within projects and horizontally across organizations.

The DoD XML Registry constitutes guidance in the generation and use of XML among DoD communities of interest and is the authoritative source for registered XML data and metadata components.

Note: Due to DoD Operations Security changes, a user must log in to access the majority of the features within the Metadata Registry. The only publicly available pages will be the [user registration form](#), [What's New](#) and other general information pages.

[Management Initiative Decision \(MID\) 905 Survey Questionnaire](#)

Fill out this survey to satisfy the MID905 Metadata Registry requirements.

XML Registry Home

What's New

Briefings & Documentation

Info Desk

Tools

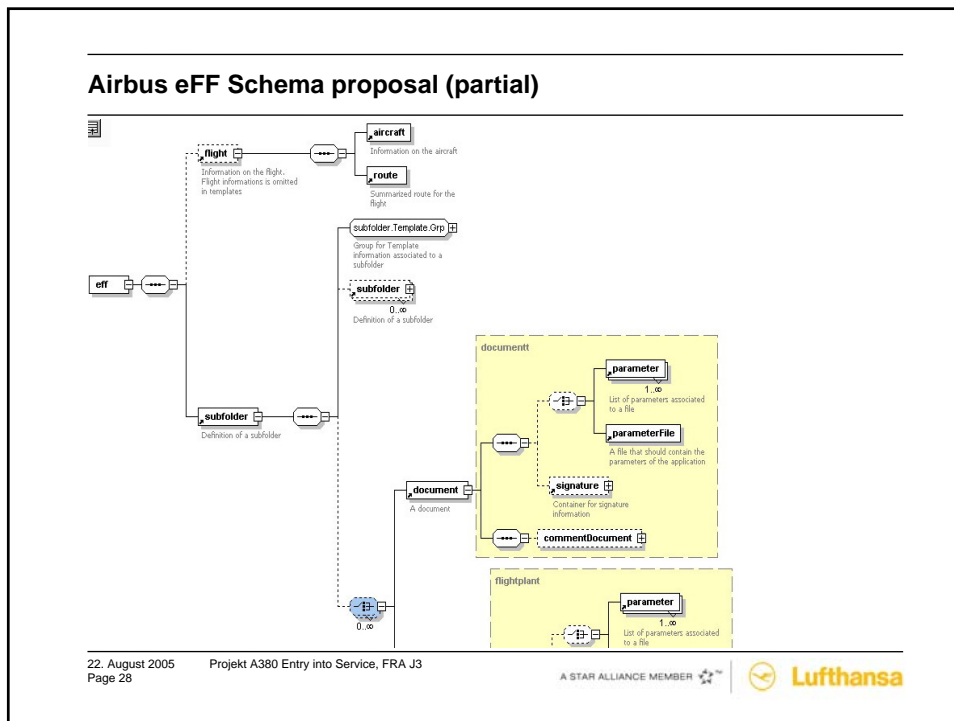
Search

Registration

Links

MetaData

Document: Done



Example of a XML-coded EFF Document (Airbus proposal, partial)

```

<FlightPlan
  version="1"
  computedDate="2004-06-30"
  flightDate="2004-06-30"
  flightID="AIB04"
  regist="AIB-PPTEST"
  xmlns="http://aeec.arinc.net/633/flightPlan"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://aeec.arinc.net/633/flightPlan .\flightplan.xsd"
  acmodel="A380"
  xmlns:ext="www.mycompany.com/myschemaextension">
  <FuelHeader>
    <TRIP>
      <EFUEL>151646</EFUEL>
      <AFUEL>... </AFUEL>
      <ETIME>17:46:00</ETIME>
      <airport iataCode="WSSS" icaoCode="WSSS"/>
      <NAM>8519</NAM>
      <NM>8590</NM>
      <FLT>330</FLT>
    </TRIP>
    <RESV>
      <EFUEL>007582</EFUEL>
      <AFUEL>... </AFUEL>
      <ETIME>00:53:00</ETIME>
      <COEF>0.05</COEF>
    </RESV>
    <alternates>
      <alternate>
        <EFUEL>005715</EFUEL>
        <AFUEL>... </AFUEL>
        <ETIME>00:36:00</ETIME>
        <airport iataCode="WMKK" icaoCode="WMKK"/>
        <NAM>0190</NAM>
        <NM>0190</NM>
        <FLT>200</FLT>
      </alternate>
    </alternates>
    <HOLD>
      <EFUEL>003415</EFUEL>
      <AFUEL>... </AFUEL>
      <ETIME>00:30:00</ETIME>
    </HOLD>
  </FuelHeader>

```



Route Clearance Notation

RouteClearance ::= SEQUENCE

airportDeparture	[0]	Airport	OPTIONAL,
airportDestination	[1]	Airport	OPTIONAL,
runwayDeparture	[2]	Runway	OPTIONAL,
procedureDeparture	[3]	ProcedureName	OPTIONAL,
runwayArrival	[4]	Runway	OPTIONAL,
procedureApproach	[5]	ProcedureName	OPTIONAL,
procedureArrival	[6]	ProcedureName	OPTIONAL,
routeInformations	[7]	SEQUENCE SIZE (1..128) OF RouteInformation	OPTIONAL,
routeInformationAdditional	[8]	RouteInformationAdditional	OPTIONAL

ASN.1

```

<xsd:complexType name="RouteClearance">
  <xsd:sequence>
    <xsd:element name="airportDeparture" type="Airport" minOccurs="0"/>
    <xsd:element name="airportDestination" type="Airport" minOccurs="0"/>
    <xsd:element name="runwayDeparture" type="Runway" minOccurs="0"/>
    <xsd:element name="procedureDeparture" type="ProcedureName" minOccurs="0"/>
    <xsd:element name="runwayArrival" type="Runway" minOccurs="0"/>
    <xsd:element name="procedureApproach" type="ProcedureName" minOccurs="0"/>
    <xsd:element name="procedureArrival" type="ProcedureName" minOccurs="0"/>
    <xsd:element name="routeInformations" type="RouteInformation" minOccurs="0" maxOccurs="128"/>
    <xsd:element name="routeInformationAdditional" type="RouteInformationAdditional" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name="Airport">
  <xsd:restriction base="IA5String">
    <xsd:minLength value="4"/>
    <xsd:maxLength value="4"/>
  </xsd:restriction>
</xsd:simpleType>

```

XML

Further reading

Some useful links to XML websites:

<http://www.w3.org/XML/>

<http://www.xml.org/>

<http://www.w3schools.com/>

© AIRBUS S.A.S. All rights reserved. Confidential and proprietary document.

Introduction to XML

June 2005 Page 35



© AIRBUS S.A.S. All rights reserved. Confidential and proprietary document.

This document and all information contained herein is the sole property of AIRBUS S.A.S.. No intellectual property rights are granted by the delivery of this document or the disclosure of its content. This document shall not be reproduced or disclosed to a third party without the express written consent of AIRBUS S.A.S.. This document and its content shall not be used for any purpose other than that for which it is supplied.

The statements made herein do not constitute an offer. They are based on the mentioned assumptions and are expressed in good faith. Where the supporting grounds for these statements are not shown, AIRBUS S.A.S. will be pleased to explain the basis thereof.

AIRBUS, its logo, A300, A310, A318, A319, A320, A321, A330, A340, A350, A380, A400M are registered trademarks.



AIRBUS

AN EADS JOINT COMPANY
WITH BAE SYSTEMS

© AIRBUS S.A.S. All rights reserved. Confidential and proprietary document.

Introduction to XML

June 2005 Page 36