SEMIDE

EMWIS

SEMIDE EMWIS

http://www.semide.org

http://www.emwis.org

An introduction to XML for the SEMIDE Project

Francesco Fedele

Web Area Manager and Chief Internet Architect, *CM InfoLand*

CEO, *Effedue Consulting*

EURO-MEDITERRANEAN PARTNERSHIP

SEMIDE

EMWIS

SEMIDE

http://www.semide.org

EMWIS

http://www.emwis.org

The XML standard

EURO-MEDITERRANEAN PARTNERSHIP

SEMIDE
EMWIS

The markup concept

- The electronic production of texts has been solved by the use of two different techniques:
  - word processors
  - markup languages
- Word processors are WYSIWYG (What You See Is What You Get) systems that are limited insofar as portability and re-use are concerned
- On the other hand, markup languages describe the structural and representative mechanisms of the text by means of real languages that often use standardised conventions, thus usable on a greater range of systems
- A markup language called SGML has existed for years
- A new markup language was created for the web: HTML – the web was born and documents were “published”

SEMIDE
EMWIS

An example of HTML

```
<!--Example of HTML  -->
<h1>Invoice 01/00</h1>
<p>Effedue Consulting
<p>Date: 31 October 2000
<br><br>
<p>IBM SpA
<br>
<p>Amount: GBP 2,000
<p>VAT: 20%
<p>
<b>Total: GBP 2,400
```

SEMIDE
EMWIS

Separating the contents from the presentation

- Characteristics of HTML
  - guarantees the correct use of the information due to the distinction made between structure and presentation
  - easy to use
  - adapted to basic publishing
  - the instructions used do not require the browser to carry out a validation test on the hierarchical structure of the information
- Limits of HTML
  - created as a markup language for making documents available on line, it has become a technology assembler capable of meeting the requirements of the “net”
  - it is not extensible
  - it is display-centred
  - it is not, in general, re-usable
  - it has little or no semantic structure
  - it is rigid, supplying a single “view” of the data

SEMIDE
EMWIS

The evolution of the Web

- From a document publishing tool to:
  - directory tool
    - goods, services and information
  - application platform
    - sites with integrated applications that help with the management and analysis of the data
- This development poses problems of integration between heterogeneous data and data in various formats:
  - management of integration based on the logical display of the data and not through its physical implementation
  - describing, in a standardised manner, the logical display of the information available
  - Creating interactions between the site and other sites (server/server), not only with the client (server/client)

SEMIDE
EMWIS

The need for a substitute for HTML

- New Internet requirements have generated the need to evolve away from HTML
  - e-commerce applications require an open and robust format for the exchange of data
  - SGML (Standard Generalized Markup Language) does exist, but it is too complex and was not designed for use on the web
  - HTML (HyperText Markup Language) is not data-oriented, it is only adapted to Web page presentations
- A tool capable of facilitating data exchange is required
  - a markup language with 80% of SGML's functionalities and 20% of its complexity
  - adapted to document processing and web publishing but also to the development of applications
  - similar to HTML, but more flexible and data-oriented

The creation of XML

- A standard proposed by the World Wide Web Consortium (W3C)
- eXtensible Markup Language
  - Language – it is a language
  - Markup – instructions controlling the content and information aspect
  - eXtensible – users may define a set of tags according to their requirements
- A TAG based text format
  - Separates contents and presentation
  - It is responsible for data definition only

SEMIDE
EMWIS

The creation of XML

- It is a direct descendant of SGML
  - Simplified for the Web
  - HTML is also a SGML subset
- Rigorous, but easy to interpret
- It can represent:
  - traditional record-structured relational data
  - hierarchical data
  - unstructured data
- XML separates contents and presentation

SEMIDE
EMWIS

XML is new, but not that new

- The development of XML started in 1996
- This 1998 W3C standard may have generated the suspicion that the technology at hand was rather immature. In reality the technology used is not really new
- SGML preceded XML, created in the early 80s,
  1986 ISO standard, mostly used for large documentation projects.
- XML has benefited from the experience acquired with HTML, the development of which started in 1990
- The XML developers took the best elements of SGML, using the experience gained with HTML as a guideline, and produced something just as powerful as SGML, but more uniform and easier to use
- Differentiating each evolution from the revolutions is a difficult task…
- While SGML is mainly used for technical documentation and far less for other types of data, the complete opposite is true for XML

Rules for XML tags

- The tags must be nested and each opening necessarily requires a closing tag
  - <a>
    -
    -
  -
- Tags that do not contain text but only their name and attributes may also contain their own ending.
  The following lines are equivalent:
  - <Lilia age="4"></Lilia>
  - <Lilia age="4"/>

SEMIDE
EMWIS

Opening and closing tags

- Tags may be nested one inside another in order to describe the relations existing between data blocks
  - E.g. <cartypes>
    <cabriolet>
    <normal>
    <automatic>
    <auto> Model A</auto>
    </automatic>
    <manual>Model B</manual>
    </normal>
    </cabriolet>
    </cartypes>
  - Car types
    - Cabriolet
      - Normal
        - Automatic - Model A
        - Manual - Model B

SEMIDE
EMWIS

Attributes, Entity, Comments

- Attribute assignment:
  - format always
    [attribute name] “=“ [attribute value]
- References to the entity:
  - used to insert characters that would otherwise confuse the XML parser
  - e.g. in XML “<“ is represented by “lt”
  - references to the entity are predefined in order to face such problems
- Comments
  - Same format as for HTML
  - <!-- this is a comment -->

Rules for XML documents

- The XML document must be contained within the opening and closing tags of a single element called the ***root element***
  - an element is made up of an opening tag, a closing tag and everything that is contained between the two (including other elements)
- Although unnecessary for many parsers, XML documents always start with a statement:
  - <?xml version="1.0"
  - encoding="ISO-8859-1"?>
    - The *version* attribute is compulsory; if left out *encoding*="UTF-8" is the default setting

SEMIDE
EMWIS

Well formed documents

- Documents that respect all rules for XML documents
  - nested tags
  - a closing tag for each opening tag
  - etc.

  are said to be “well formed”

SEMIDE
EMWIS

From HTML to XML

```
<!--Example of HTML  -->
<h1>Invoice 01/00</h1>
<p>Effedue Consulting
<p>Date: 31 October 2000
<br><br>
<p>IBM SpA
<br>
<p>Amount: GBP 2,000
<p>VAT: 20%
<p>
<b>Total: GBP 2,400
```

```
<!-- Example of XML  -->
<Invoice>
<Number>01/00</Number>
<Emitted by>Effedue
    Consulting</Emitted by>
<Date year = '2000' month =
    '10' day = '31' />
<To>IBM</To>
<Amount value = 'GBP'>2000
    </Amount>
<%VAT>20</%VAT>
<Total value =
    'GBP'>2400</Total>
</Invoice>
```

XML slightly resembles HTML but is not HTML

- Like HTML, XML uses tags (words comprised between the less than and greater than signs ('<' and '>') and attributes (in name format="value"), however, while HTML specifies the signification of each tag and attribute (and often what the text contained between them will look like in a browser), XML only uses tags to delimit parts of information, leaving the interpretation of that information entirely to the application reading it
- In other words, if a "<p>" tag appears in an XML file, it cannot be assumed that it represents a paragraph. Depending on the context it may be a price, a parameter, a person, a p….
  (by the way, who said that it should be a word starting with "p"?)

SEMIDE EMWIS

SEMIDE

http://www.semide.org

EMWIS

http://www.emwis.org

Standards of the XML family

EURO-MEDITERRANEAN PARTNERSHIP

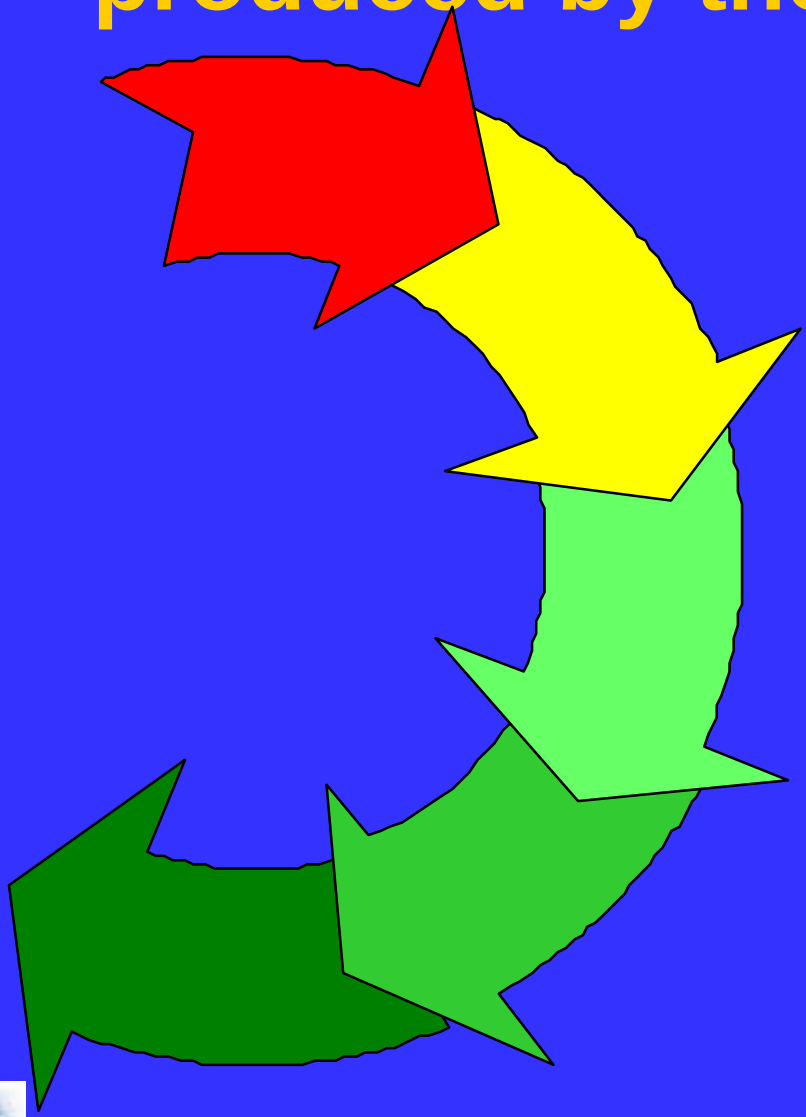
SEMIDE
EMWIS

The W3C and the XML Coordination Group

- The World Wide Web Consortium (W3C) is the international body that presides over the development of standards in relation to the Web
- This group defines HTML (up to version 4.01)
- XML was created within the W3C (during a chat between Tim Bray and Dave Hollander leaving the WWW3 in a taxi)
- Today, the organisation has a specific group for the development of XML

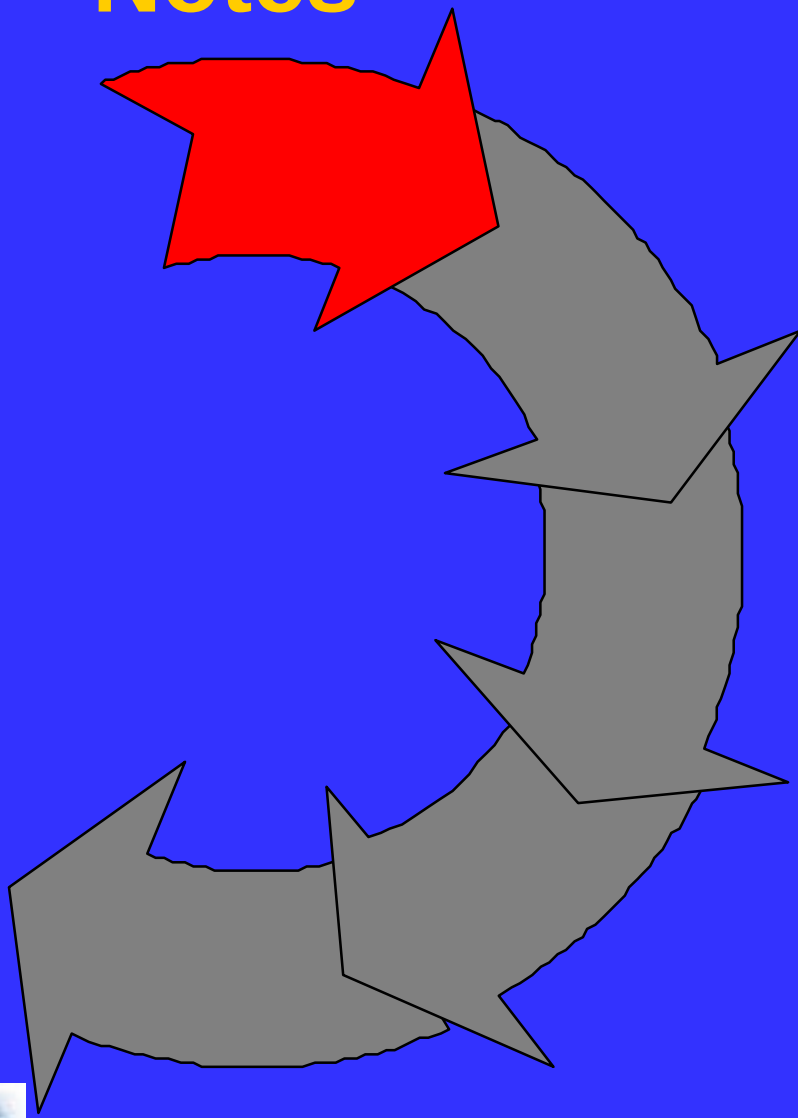
- XML Coordination Group
  - XML Core
    - errata, X-Include, Information Set
  - XML Schema Working Group
    - Parts 0, 1, 2, 3
  - XML Linking WG
    - XML Base, Xpath, Xlink, Xpointer
  - XML Query WG
    - Data Model, Algebra, Language
  - XML Namespaces
- 17 different standards have been created in relation to XML !!!

Types of documents produced by the W3C



- Notes
- Working Drafts
  - Working Drafts in development
  - Working Drafts no longer in development
  - Working Drafts in Last Call
- Candidate Recommendations
- Proposed Recommendations
- Recommendations

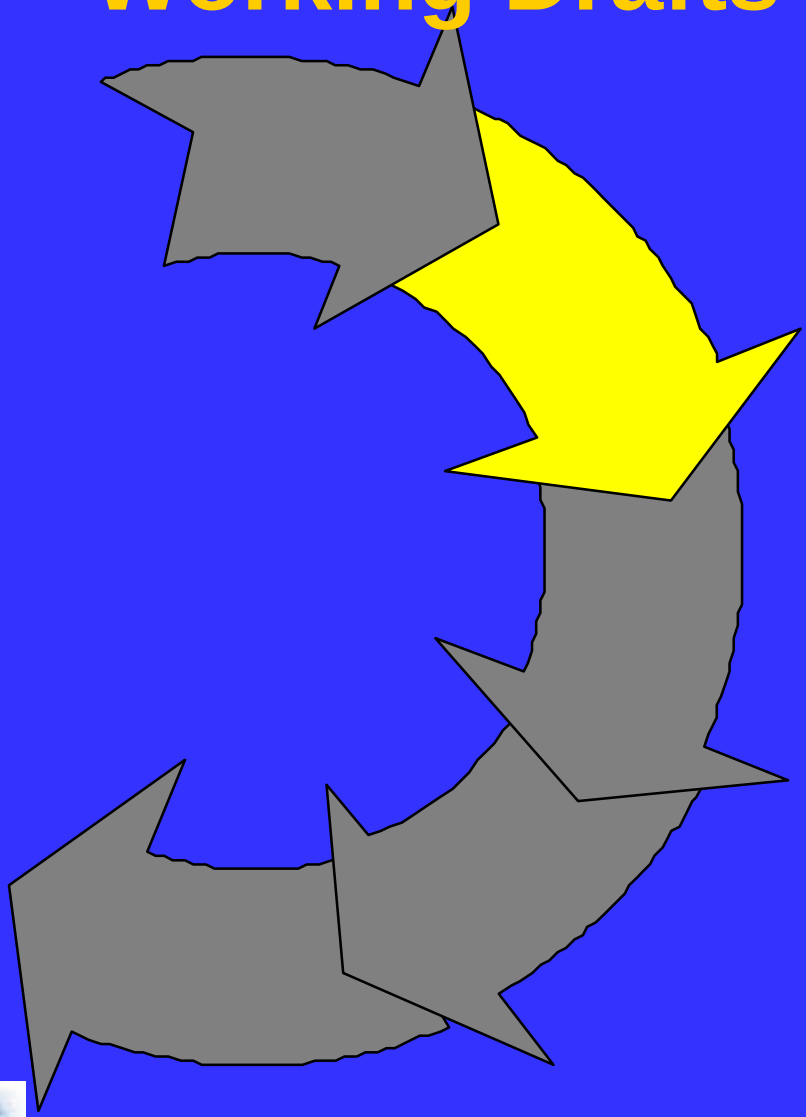
Notes



- A note is the publication of an idea, a comment or a document at a certain date
- A note represents no obligation for the W3C to continue the work related to the note and is published at the Director's discretion
- Some notes are proposals made by known members of the consortium (Acknowledged Member Submissions)

SEMIDE
EMWIS

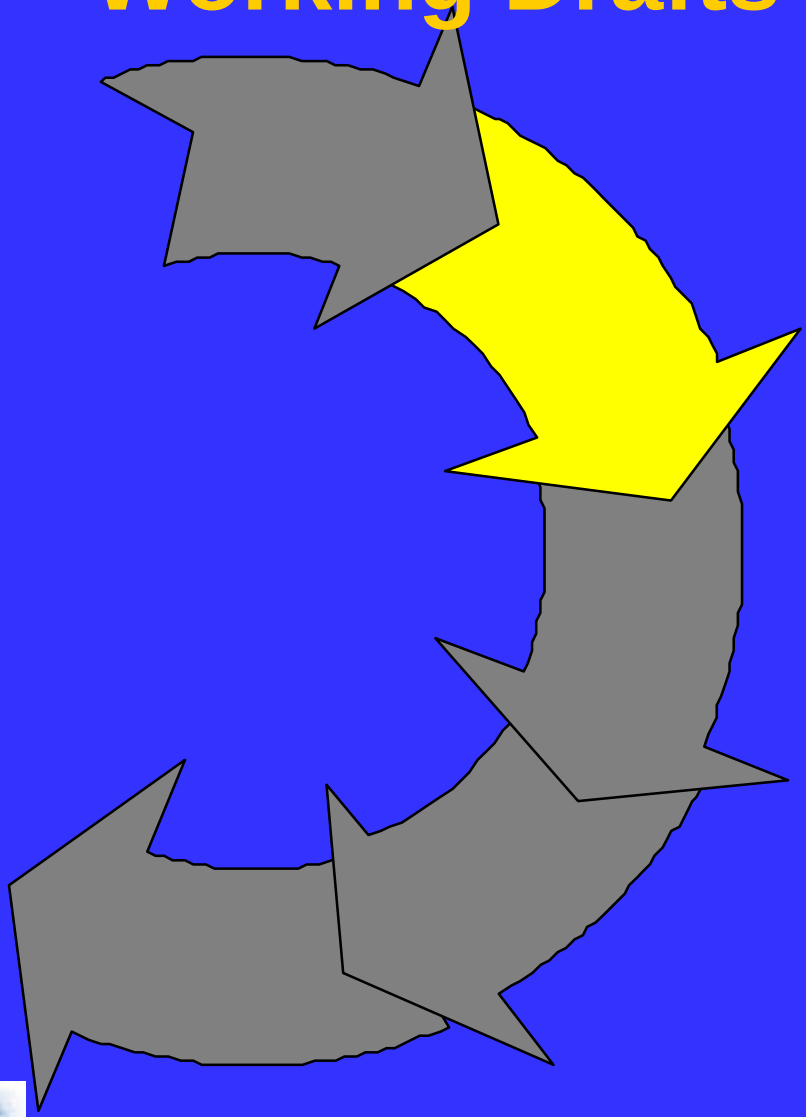
Working Drafts



- Development phase work and an obligation for the W3C to pursue their objective in that domain
- It does not imply agreement of a group or of the W3C, but is published in order to be analysed and revised by members of the W3C and other interested parties
- Drafts may be changed, replaced or become obsolete at any time
- Use of Working Drafts as reference material would be inappropriate as would be considering them as other than work still in the development phase

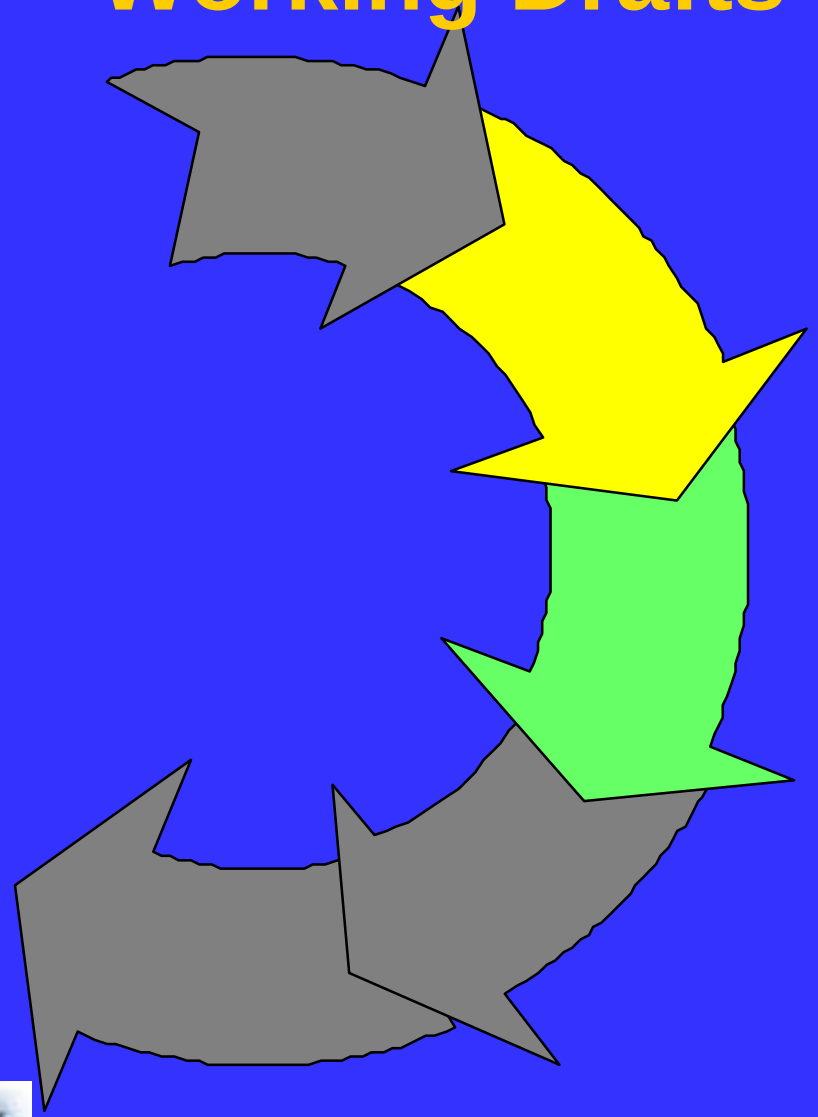
SEMIDE
EMWIS

Working Drafts



- In development
  - Most Working Drafts are currently in development
- No longer in development
  - some are abandoned or are incorporated into other documents
  - the final versions of these documents constitute their own epilogue
  - the status section of the final document describes the reason why the work was abandoned or where it was incorporated

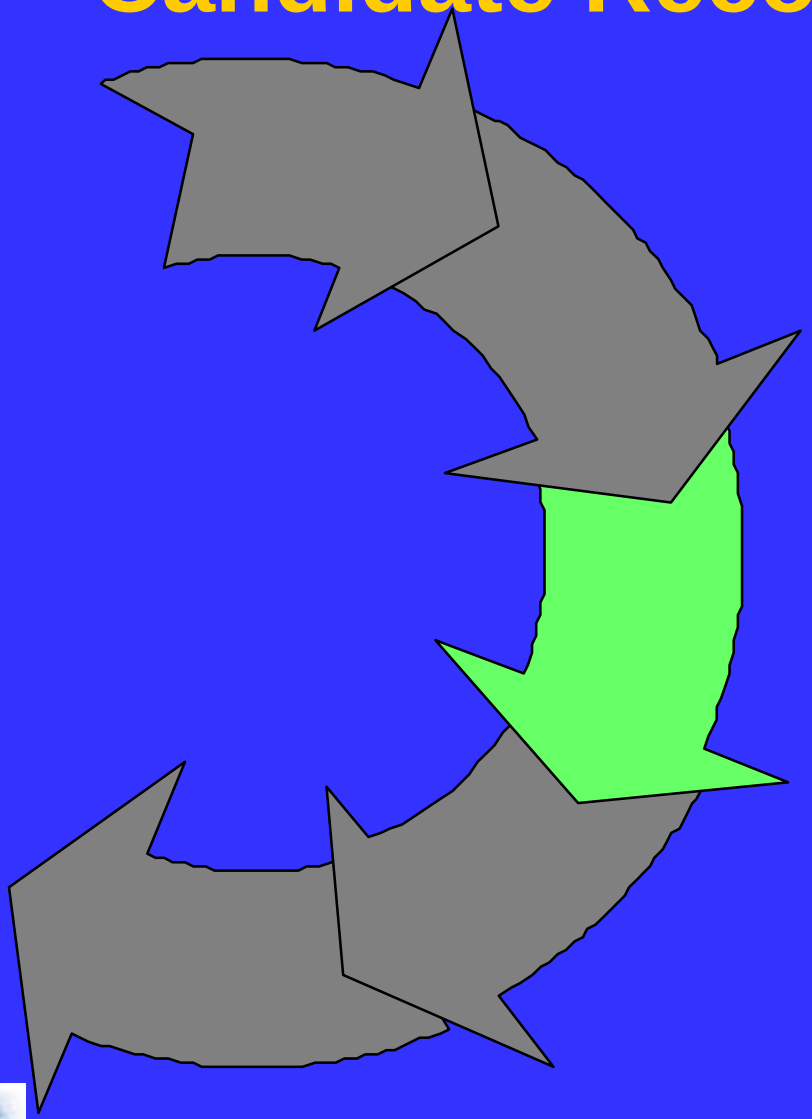
Working Drafts in Last Call



- A Working Draft in Last Call must be re-examined by the work groups depending on such technology or at least showing a qualified interest in it
- The length of the revision period is noted in the document's status section

SEMIDE
EMWIS

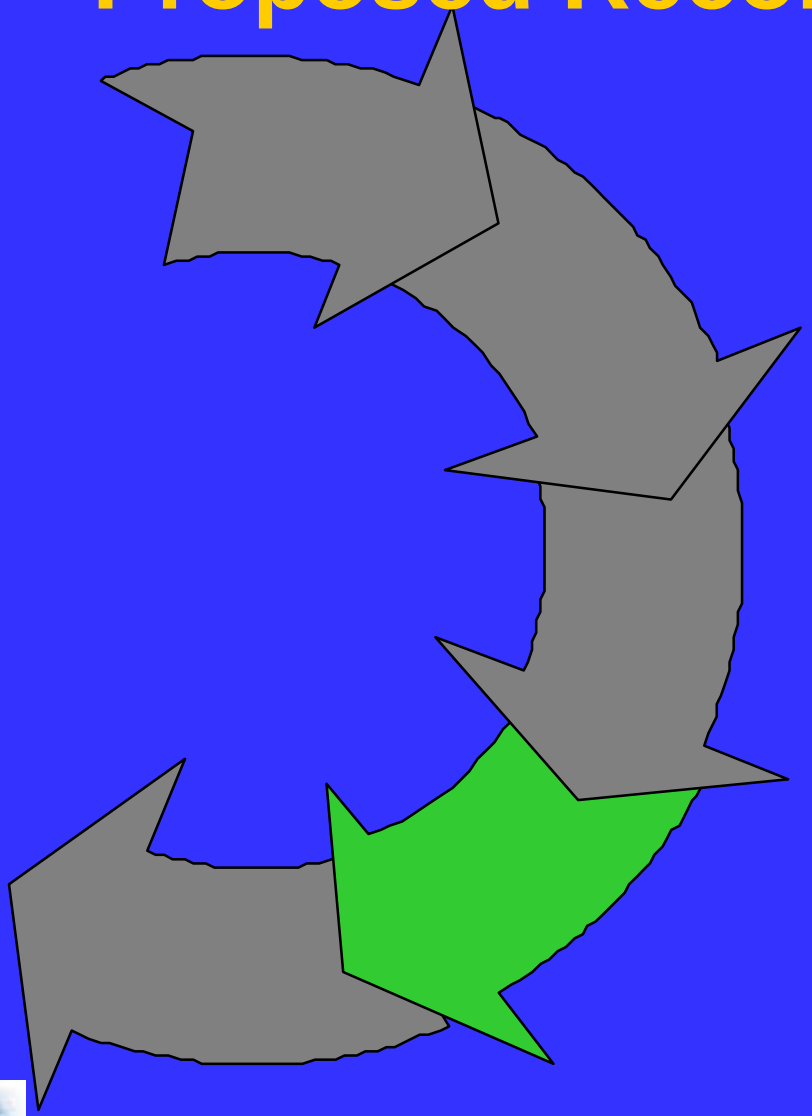
Candidate Recommendations



- A Candidate Recommendation is work that has been significantly analysed by its technical group
- It constitutes an explicit invitation to those outside the work groups or the W3C itself for implementation and technical verification

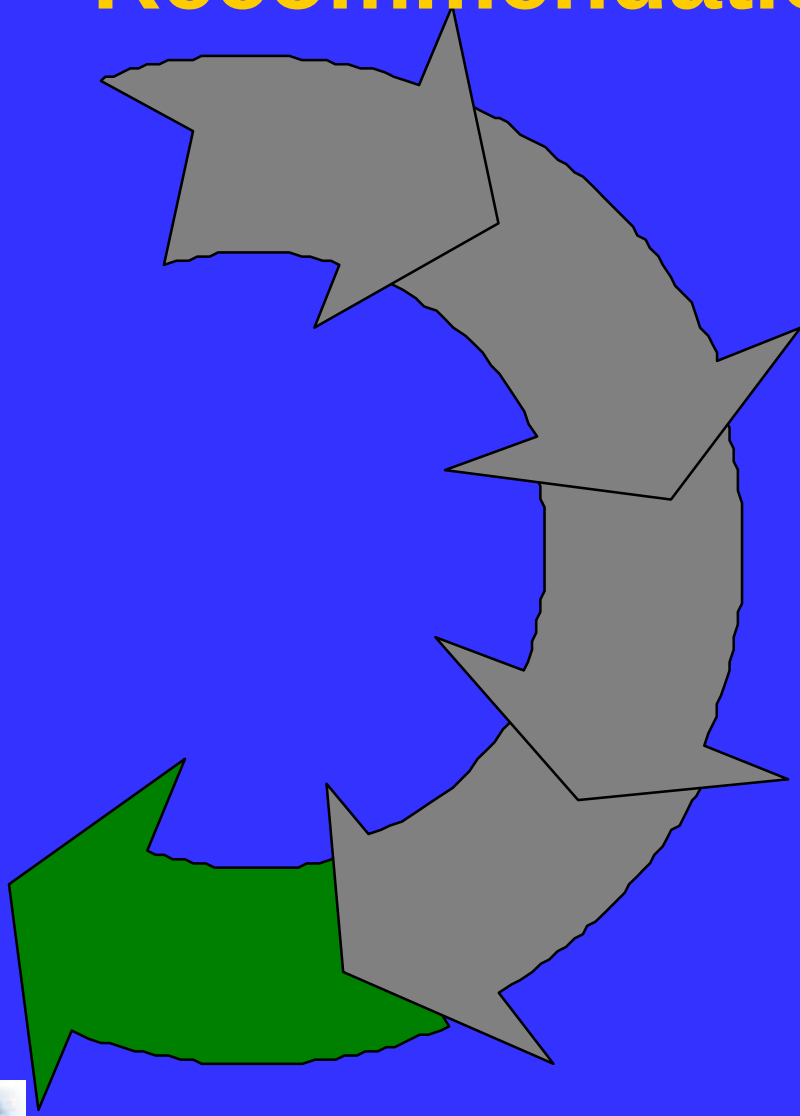
SEMIDE
EMWIS

Proposed Recommendations



- A Proposed Recommendation is work that
  - has obtained a consensus within the work group that produced it
  - is proposed by the Director to the Advisory Committee for Review

Recommendations



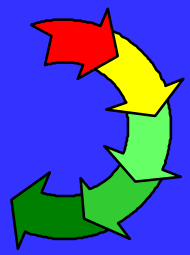
- Is a work that has obtained a consensus within the W3C and to which the Director has given his approval
- The W3C declares that the ideas or technology that are the subject of the recommendation may be circulated and may promote the mission of the W3C
- Sometimes a “first edition” of a recommendation may be followed by a “second edition”, the aim of which is mainly to correct any possible misprints present in the “first edition”

SEMIDE
EMWIS

Standards of the XML family

- Standards exist in relation to:
  - XML
  - data schemas and structure
  - research and transformation tools
  - API Parser
  - document linking
  - security and access to remote objects

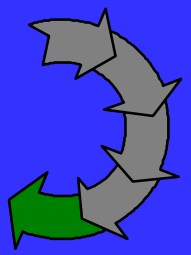
- They are classified according to:
  - Relevance
  - Dissemination
  - Level of standardisation



- The following information is supplied for each one:
  - Date
    - dd.mm.yyyy
  - Version
    - 1.0
  - URL
    - http://

XML

- Relevance ★★★
- Dissemination ★★★
- Standardisation



Date
10.2.1998
Version
1.0
URL
http://www.xml.it/REC-xml-19980210-it.html
http://www.w3.org/TR/REC-xml

- The text is quite short (for a standard)
- On 6th October 2000 the “second edition” XML standard was published as a Recommendation,
- It was mainly developed by:
  - John Bosak (Sun) (Chairman)
  - James Clark (Technical Lead)
  - Tim Bray, Textuality and Netscape (XML Co-ed)
  - Jean Paoli, Microsoft (XML Co-ed)
  - C. M. Sperberg-McQueen, U. of Ill. (XML Co-ed)
  - Dan Connolly, W3C (W3C Liaison)
  - Dave Hollander, HP

SEMIDE
EMWIS

Standards in relation to data schemas and structure

- DTD
- XML Schema
- RDF
- Namespace

SEMIDE
EMWIS

Structuring an XML document

- Since XML is an extensible language, the user must create the rules establishing
  - what data should be included in an XML document
  - what data is considered necessary or optional
  - how the data should be interpreted
- Rules are grouped together in an appropriate document
  - the conformity of the XML documents subjected to such a structure is verified
- Determining whether there is any missing data is easy
  - thus guaranteeing accuracy in a catalogue
  - imposing links to the format of the data is possible

SEMIDE
EMWIS

Two alternatives: DTD or XML Schema

Capable of defining grammatical specifications for applications

- DTD (Document Type Definition)
  - grammar for tag and attribute descriptions
- XML Schema
  - Richer grammar for XML-based descriptions

SEMIDE
EMWIS

DTD (Document Type Definition)

- Relevance ★★★
- Dissemination ★★★
- Standardisation

Date

Version

URL

Standard related to SGML

DTDs were defined at the time of SGML

- They are used to define the syntax of a document and define:
  - tags that may or must be present in the document
  - how the tags are to be nested
  - how often a given tag may be repeated in the document
  - tag attributes and their default values
  - all valid values for given attributes
- They may semantically define a group of data specifying the different possibilities allowed and when these are possible
- They are written in a specific language (not XML)

SEMIDE
EMWIS

An example of DTD

```
<!ELEMENT Address
    (name, street, town, country, postcode>
    <!ELEMENT person
    (title?, forename, surname)>
    <!ELEMENT title (#PCDATA)>
    <!ELEMENT forename (#PCDATA)>
    <!ELEMENT surname (#PCDATA)>
    <!ELEMENT street (#PCDATA)>
    <!ELEMENT town (#PCDATA)>
    <!ELEMENT country (#PCDATA)>
    <!ELEMENT postcode (#PCDATA)>
```

SEMIDE
EMWIS

Processing, CDATA, DTDeclaration

- CDATA
  - used to insert a large number of special characters in an XML parser
  - converts data into character data

    <!CDATA[ ]]>
- Processing instructions
  - used when an application - a specific communication is necessary
  - Enables instructions to be included referring to other applications in the code

    <?rtf \page?>
- Document Type Declaration
  - associates a DTD with an XML document and declares where the DTD is stored in memory

Creating a DTD

- Element type declaration
  - establishes the name of the element type and what it may contain
  - e.g. <!ELEMENT contact (name, address, telephone)> declares that a contact type element must contain a name, an address and a telephone number (in that order)
- Declaration of a list of attributes
  - establishes the names of attributes and associates an element type
    - <!ATTLIST PRODUCT
    - name…
    - colour...
    - this means that a product element type has two attributes noted as name and colour

Creating a DTD

- **Declaration of an entity:**

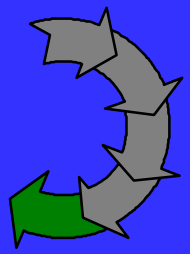
  <!ENTITY chapter 1
  SYSTEM
  “http://www.digitome.com/cap1.xml”

  this means that a chapter 1 entity exists.

  On detection of a reference to that entity within an XML document, the parser inserts the contents of the file http://www.digitome.com/cap1.xml

XML Schema

- Relevance ★★★
- Dissemination ★★
- Standardisation



Date
24.10.2000
Version
1.0
URL
http://www.w3.org/TR/xmlschema-… 0,1,2

- XMLSchema is a mode for the description of the structure, contents and type of the XML data
- It extends the functionalities of the DTDs and solves some of their problems
- It allows greater control of the XML file elements, their contents, the data types and their value
- Like DTD it may be used to validate XML
- Unlike the DTDs, XMLSchema uses an XML syntax

XML Schema and Data Types

XML Schema permits the use of a large number of data types in addition to the string type

- ui1, ui2, ui4
- r4
- r8
- uuid
- uri
- bin.hex
- bin.base64
- string
- int
- float
- boolean
- dateTime
- date
- time
- i1, i2, i4

SEMIDE
EMWIS

XML Schema vs. DTD

- XML Schema

```
<ElementType name="foo" dt:type="int"
   minOccurs="1" maxOccurs="*" />
<ElementType name="bar">
    <element type="foo" />
</ElementType>
```

- DTD

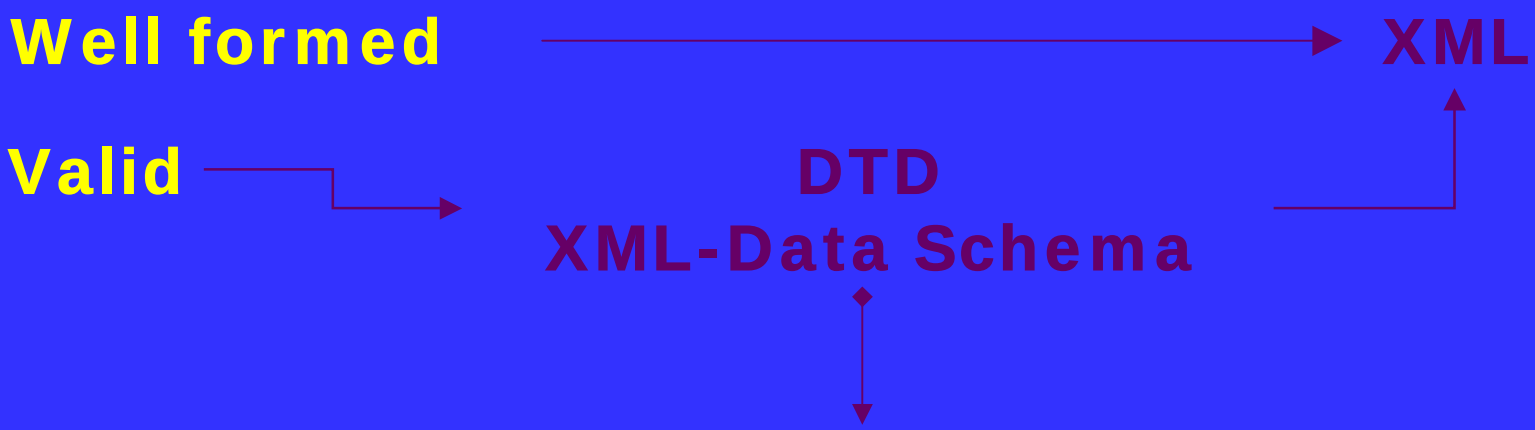
```
<!DOCTYPE bar [
  <!ELEMENT bar (foo)+>
  <!ELEMENT foo EMPTY>
  <!ATTLIST foo dt:type #FIXED "int">
]>
```

SEMIDE
EMWIS

Valid documents

- **Well formed documents using a Document Type Definition and conforming to all DTD rules are also “valid” documents**

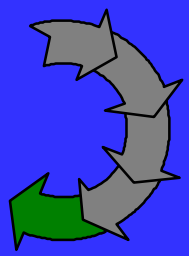
Parsing




```
<?XML version="1.0" standalone="no" encoding="UTF-8"?>
<!DOCTYPE book SYSTEM "automobile.dtd">
```

RDF

- Relevance ★★
- Dissemination ★★
- Standardisation



Date
22.2.1999
Version

URL
http://www.w3.org/TR/REC-rdf-syntax

- Enables the insertion of user-determined specific metadata in an XML file, such as the specific key for that document, the name of the author, the protocol number, copyright information, etc.
- An RDF schema thus extends the HTML metatag concept and allows the user to define metatag structure models to be associated with all documents of a given category, providing them with the same identification structure

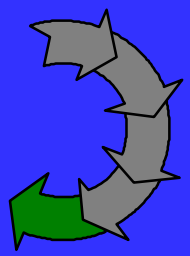
SEMIDE
EMWIS

Namespaces

- Relevance ★★★
- Dissemination ★★★
- Standardisation

Date
14.1.1999
Version

URL
http://www.w3.org/TR/REC-xml-namespaces



A mechanism allowing the user to mix elements and attributes with the same name but with different significations

- **Explicit Namespace**

```
<BOOKS>
    <bk:BOOK xmlns:bk="urn:BookLovers.org:BookInfo
    "xmlns:money="urn:Finance:Money">
    <bk:NAME>A Suitable Boy</bk:NAME>
    <bk:PRICE money:NAME="US Dollar">22.95</bk:PRICE>
    </bk:BOOK>
</BOOKS>
```

- **Default Namespace**

```
    <BOOK xmlns="urn:BookLovers.org:BookInfo">
      <TITLE>A Suitable Boy</TITLE>
      <PRICE currency="US Dollar">22.95</PRICE>
    </BOOK>
```

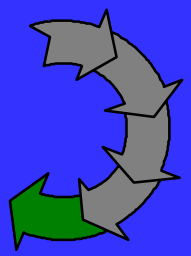
SEMIDE
EMWIS

Standards related to research and transformation tools

- XMLQuery
- Xpath
- Xpointer
- XSL
- XSLT

XPath

- Relevance ★★★
- Dissemination ★★★
- Standardisation

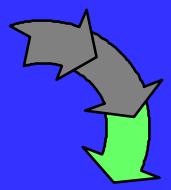


Date
16.11.1999
Version
1.0
URL
http://www.w3.org/TR/REC-xpath

- XML Path Language is a standard for locating information within XML documents
- It is a complex and sophisticated language
  - XPath may be used to locate text data, elements, attributes or any other information type in an XML document
- Xpath can, for example, search for the “third element from the beginning of the document”

Xpointer

- Relevance ★★★
- Dissemination ★★★
- Standardisation

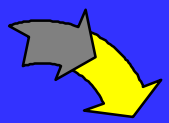


Date
7.6.2000
expiry 7.9.2000
Version
1.0
URL
http://www.w3.org/TR/xptr

- XPointer is a personalised extension of XPath
- XPointer enables the use of XPath expressions as parts of URI (URL included)
  - this allows creating a link to internal parts of XML documents, based on (e.g.) position, element type, ID, etc.
- Xpointer allows research by character strings and specific ranges. This allows selecting a single element, a group of elements or a group of elements that answer a search criteria or a totally arbitrary range

XML Query

- Relevance
- Dissemination
- Standardisation

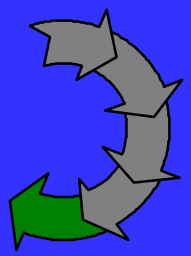


Date
15.8.2000
Version
1.0
URL
http://www.w3.org/TR/xmlquery-req

- The aim of XMLQuery is to insert typical database research abilities in XML
- This standard is still in development

XSL

- **Relevance** ★★★
- **Dissemination** ★★★
- **Standardisation**



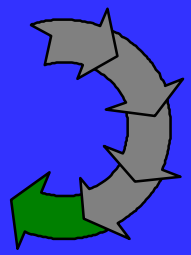
Date
18.10.2000
Version
1.0
URL
http://www.w3.org/TR/WD-xsl-20001018

- **eXtensible Stylesheet Language (XSL) is a W3C tool that is still in development aiming at creating a standard in the domain of XML document formatting**
- **XSL is an extension of the HTML style sheet concept (CSS)**
- **Includes XSL-FO for formatting specifications**
- **It allows the coding of the style rules, e.g. “all titles must be in bold type”. XSL thus guarantees homogeneous formatting of all XML documents of the same type. If the need to modify the formatting arises, all that is necessary is to modify the associated XSL**

XSLT

- **Relevance** ★★★
- **Dissemination** ★★★
- **Standardisation**

Date
16.11.1999
Version
1.0
URL
http://www.w3.org/TR/REC-xslt



- **XSL Transformations was, for a long time, only a part of XSL and has recently become an independent language allowing the transformation of an XML type document into a different type**
  - **XSL-T now constitutes the first part of the standard**
  - **the second part (XSL) defines the semantics of the formatting**
- **Allows the handling of a document**
- **Using XSLT it is possible to generate:**
  - **an HTML document**
  - **another XML document with a different structure**
  - **a VML document**
  - **an audio document**
  - **… almost any other document type**

SEMIDE
EMWIS

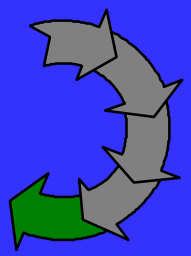
Standards for access to XML structures

Standards related to access to XML structures (API for Parsers):

- DOM
- DOM Level 2
- SAX

DOM

- Relevance ★★★
- Dissemination ★★★
- Standardisation

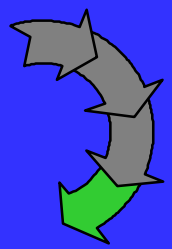


Date
1.10.1998
Version
1.0
URL
http://www.w3.org/TR/REC-DOM-Level-1

- The DOM is at the basis of XML parsers, in that it is a model that permits the reproduction of the data hierarchy in the native object format of the programming language used, providing programmers with a simple method for locating and processing data contained in the document
- Developers can navigate within the tree structure built by the parser, containing all the elements of the document and their possible attributes that, in turn, contain the data: it is even possible to change the contents of the document in a programmed manner
- The publication of a “second edition” of DOM is at an advanced stage of development

DOM Level 2

- Relevance ★★★
- Dissemination ★★
- Standardisation



Date
27.9.2000
expires 25.10.2000
Version
1.0
URL
http://www.w3.org/TR/REC-DOM-Level-2-…
(Core, Events, HTML, Style, Trasversal-Ranc, Views)

- DOM Level 2 extends the basic representation capacities of API DOM to support other program extensions such as the user's "namespace", style sheets, event management, iterations, filters and predefined value verification and management functions (range)
- This provides developers with a standard procedure for calling up and using functions that must first be developed as such or supplied – in a non-standardised manner – by that specific implementation

SAX

- Relevance ★★★
- Dissemination ★★★
- Standardisation
  - not a W3C standard
  - created as an open-source project

Date
May 2000
Version
R.2

- SAX was created for the management of large XML documents
- Unlike DOM, SAX does not load the whole document in memory, but opens it and reads it in sequence and collects the “events” related to the data identified
- An application checks those events collected from the parser and, once a particular event has been identified, can place itself at a specific portion of the document
- SAX carries out a data research, reading the document in sequence until an element answering certain search criteria is identified
- SAX cannot be reversed and is thus solely adapted for the scanning of large files once, from start to end

SEMIDE
EMWIS

Standards related to document linking

- Xbase
- Xlink
- Xinclude

XBase

- Relevance ★★
- Dissemination ★★
- Standardisation

Date
8.9.2000
expires 8.12.2000
Version

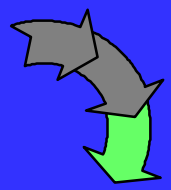
URL
http://www.w3.org/TR/xmlbase

- XBase permits the use of URI (Uniform Resource Identifiers) paths in XML documents
- The use of URI paths simplifies the codification of XML documents that use references to external URLs
- XBase simplifies the coding and facilitates the maintenance of the document in the case of the address of the location containing the external files being changed

SEMIDE
EMWIS

Xlink

- Relevance ★★★
- Dissemination ★★
- Standardisation

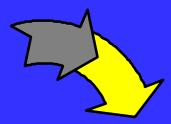


Date
3.7.2000
expires 3.10.2000
Version
1.0
URL
http://www.w3.org/TR/xlink

- XML Linking Language allows far more powerful linking compared with that used in HTML, which is unidirectional
- Multi-directional links are possible, crossing in multiple directions and the links may be memorised in databases independent of the document to which they refer
- The functionalities of Xlink also allow specification of the navigation sequence between the various documents specified in the links
- Use of Xlink is particularly suited to the creation of teaching materials or Public Administration documents for which the use of a specific sequence is required to consult the documents

XInclude

- Relevance ★★★
- Dissemination ★
- Standardisation



Date
17.7.2000
Version
1.0
URL
http://www.w3.org/TR/xinclude

- XInclude copies a whole XML document, or selected portions of an XML document, to the current document
- XInclude makes far easier the use of the “copy and paste” functions, usually used during the creation of documents
- XInclude uses Xpointer and Xpath to determine which parts of a document are to be included

SEMIDE
EMWIS

Standards for security and access

Standards related to security and access to remote objects:

- XML Signature
- SOAP

XML Signature

- Relevance ★★★
- Dissemination ★
- Standardisation

Date
11.11.2000
Version
1.0
URL
http://www.w3.org/TR/xmldsig-core

- XML Signature adds the “electronic signature” to XML files, specifying its rules and syntax
- Enables the addition of signature integrity and authentication controls to XML files for all data types

SOAP

- Relevance ★★★
- Dissemination ★★
- Standardisation
  - not a W3C standard
  - implemented by IBM, MS, LOTUS

Date
10.2.1998 / 2000
Version
1.0, 1.2
URL
http://www.w3.org/TR/REC-DOM-Level-1

- SOAP specifications enable applications to invoke methods or functions that are on remote servers
- A SOAP application generates a request contained in an XML string, providing all data required by the remote method and its address
- The SOAP server transmits the request to the object specified in the address, typically using the HTTP protocol, for the transport of requests and data strings
- Once the request has been carried out and the remote execution completed, the method's response is XML formed and returned to the application

SEMIDE EMWIS

SEMIDE

http://www.semide.org

EMWIS

http://www.emwis.org

Creating XML applications in Java

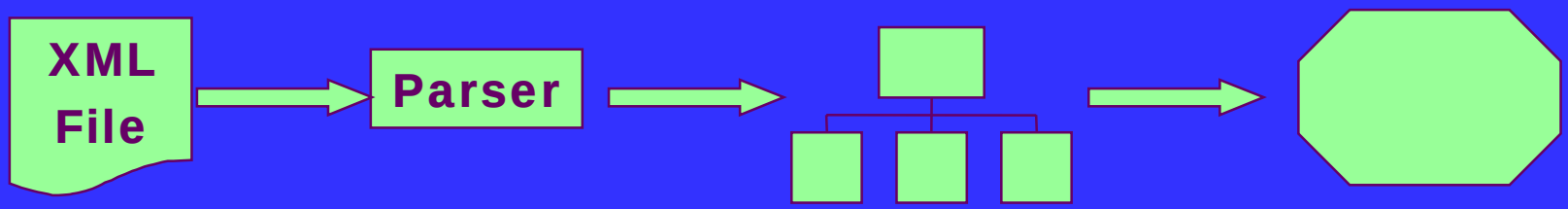
EURO-MEDITERRANEAN PARTNERSHIP

Parser - base concepts

- XML Parsing
  - XML is a metalanguage particularly adapted to data exchange
  - it is possible to consider an XML document as a simple text document, and thus to process it as such
  - XML may represent:
    - relational data
    - hierarchical data
    - unstructured data
  - XML separates the contents from the presentation
  - It is not for use with the WEB only

Parser – base concepts

- XML Parsing
  - the core component of an XML-based application is the Parser
  - A traditional XML-based application is composed of an XML file, a Parser that analyses it and the code that implements its functionality



SEMIDE
EMWIS

Parser – base concepts

- XML Parsing
  - Two methodologies exist for XML parsing
    - Non validating --> produces a ‘well formed’ document
    - Validating --> produces a ‘valid’ document
  - Non-validating parsing is carried out solely by means of syntactic control of the XML document, checking that it conforms to the specifications
  - Validating parsing also carries out a semantic check of the document, checking its conformity with a specific DTD or an XMLSchema
  - In the case of validating parsing, a second file must be written (DTD or XMLSchema), depending on the preferred approach
  - Whether dealing with validating or non-validating parsers, both are capable of supporting one or both of the more advanced APIs for XML document management, DOM and SAX

API DOM base concepts

- Document Object Model - introduction
  - DOM is Object Based
  - DOM is a framework, independent in both platform and language
  - the DOM represents an abstraction of the XML document that enables the developer to handle it with extreme ease

SEMIDE
EMWIS

API DOM base concepts

- Document Object Model - Introduction
  - a Parser supporting the DOM restores at the end of the process the document's whole hierarchical structure, in which the root node is the document itself, and the other elements (Tag, text, etc.) are the child nodes
  - the API provides the functions enabling navigation of the tree, restitution of node properties, addition or deletion of nodes, generation of another tree with a different root node, etc.

SEMIDE
EMWIS

API DOM base concepts

- Document Object Model - Standard
  - The W3C only defines the DOM, (be it a node, a list of attributes, the relations between the objects that form the model, etc.) in other words, it defines the architectural specifications of the DOM
  - The W3C does not however, being an independent organisation, specify the operating procedures of the interfaces
  - Thus, various methods exist for setting up DOM objects, according to the implementation

SEMIDE
EMWIS

API DOM base concepts

- Document Object Model - Specifications
  - The present specifications provides for two levels:
    - DOM Level 1
    - DOM Level 2
  - DOM Level 1
    - defines the generic interfaces for document handling
  - DOM Level 2
    - extension of the latter, defines the interfaces for the management of Namespace, Stylesheet, Events, etc.

SEMIDE
EMWIS

API DOM base concepts

- Il Document Object Model - Specifications
  - DOM Level 1
    - CORE DOM defines a set of objects that are capable of representing alone the instance of a document, either HTML (base 4.0) or XML 1.0
    - HTML DOM defines complete HTML support

API DOM objects

- CORE DOM main objects – Structural model
  - **Document**:
    - the XML document
  - **Node**:
    - base object, as a rule it has a parent (provided that it is not the root node), it may or may not have child nodes
  - **Element**:
    - represents all elements of a document
    - may have an associated list of attributes
  - **DocFragment**:
    - Root node of a Document Fragment

API DOM objects

- DOM XML main objects
  - XMLNode:
    - Provides specific methods for XML to the Node object
  - DocumentType:
    - Provides access to the Element, Entity and Notation type declarations
  - ElementDefinition:
    - Represents the definition of each Element contained in a DTD be it internal or external
  - AttributeDefinition:
    - Enables access to the information on the attribute of a given element

API SAX base concepts

- Simple Api for XML - introduction
  - API Event based
  - Not entirely supported by all Parsers
  - Offers less potentialities than the DOM

API SAX base concepts

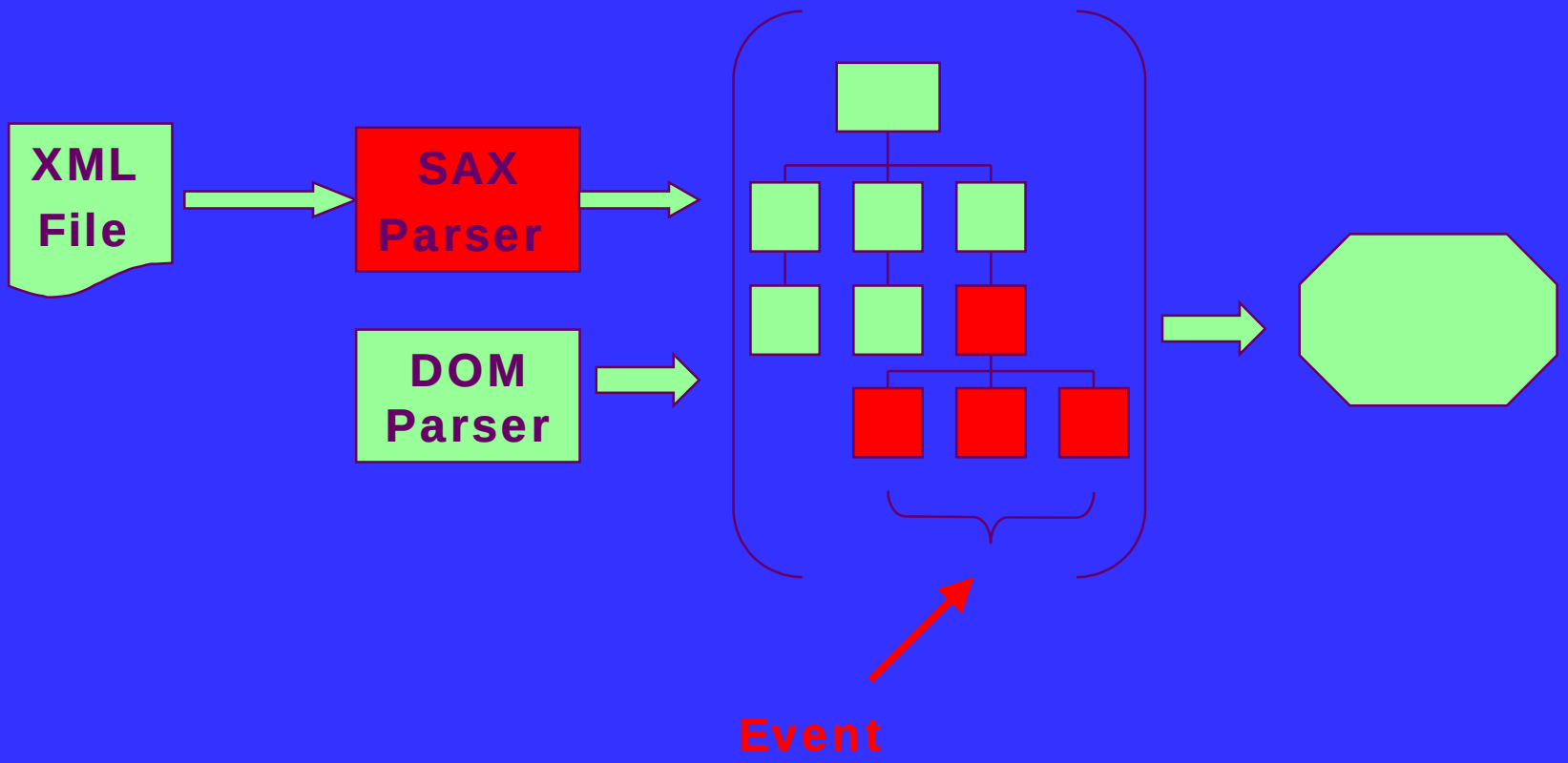
- SAX - Introduction
  - the Event concept as we know it, applied to normal graphic interface programs, is different from that used for SAX parsing
  - on scanning an XML document, the Parser establishes a tree. The nodes of that tree are considered to be Events
  - on checking the Event a Handler function is invoked (object of the DocumentHandler category) that implements the necessary functions for the management of the Events
  - the parsing process does not generate the structure of the whole document but only that related to the Event itself

API SAX base concepts

- SAX - Introduction
  - Event concept, in reference to the elements of a document
    - Start and end of a document
    - tags (opening or closing) of an element
    - structure
    - strings
    - attributes
    - PI
    - etc ...
  - the structure generated by the parsing of an event is conceptually different from the Document Fragment structure of the DOM so it is therefore not possible to obtain information on the document as a whole, but on one part only
  - the parsing process is sequential, the parser is not able to run backwards

API SAX base concepts

- Simple Api for XML
  - SAX Parsing



Conclusions

- Use DOM or SAX?
  - DOM
    - Advantages
      - Standard
      - Total support by the major producers
      - Allows total management of the document
    - Drawbacks
      - Since it loads the entire document in memory, if that document is large undesirable effects could be caused, from waiting time to exhausting physical space
      - Slow
  - SAX
    - Advantages
      - Fast
      - Event-based, selects only the Tag or Tags of interest
    - Drawbacks
      - Does not restore the structure of the document, thus further information must be obtained using codes

SEMIDE

EMWIS

SEMIDE

http://www.semide.org

EMWIS

http://www.emwis.org

Using XML with other languages

EURO-MEDITERRANEAN PARTNERSHIP

SEMIDE
EMWIS

XML and other languages Not only JAVA

- XML is not only for use with the WEB, it may also be used for data description
- XML allows inter-processing between heterogeneous systems
- The API DOM provides an object model for XML documents
- An XML document (XML DOM) is an object

SEMIDE
EMWIS

XML and other languages
Not only JAVA

- The production of an XML document can also be carried out on the Client side
- Browser levels for XML support and possible extension may cause such a production to become rather complex
- Currently, therefore, the best approach is still Server side production

SEMIDE
EMWIS

XML and other languages
Not only JAVA

- The Java and Microsoft platforms are clearly the most widespread insofar as the WEB is concerned
- One of the most widespread languages for the development of WEB applications is ASP
- ASP has thus been updated in order to support the XML DOM model
- No specific directions for use or language exist for the development of XML-based applications, for which apart from Java, other languages may be used such as:
  - ASP, C++, Perl, Python, VB, Javascript, Vbscript, etc.

Transforming XML documents

EURO-MEDITERRANEAN PARTNERSHIP

SEMIDE
EMWIS

XSL-T and XPath

- XSL-T and Xpath are used to transform XML documents
  - XSL-T is Extensible Stylesheet Language for Transformations
  - Xpath is a language for the description of positions within a document
- Both are W3C recommendations (16th November 1999)
  - XSL-T: www.w3.org/TR/xslt
  - Xpath: www.w3.org/TR/xpath
- Other standards are still in development:
  - XSL Formatting Objects: planned for high-quality press (www.w3.org/TR/xslt)
  - XLink: enables insertion of elements for the creation and description of links between resources into the XML document (www.w3.org/TR/xlink)
  - XPointer: uses XPath, enables specification of positions within XML documents (www.w3.org/TR/xptr)

Contents and presentation

- HTML tags are mainly used to define the visual presentation of the document
- XML tags unambiguously identify the data contained in the document, while its visual presentation is left to XSL and XSL-T technology
  - The structure and logical function of the tags requires that far stricter rules for their creation and use be respected than those for HTML

SEMIDE
EMWIS

File Edit Format Help

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

  <ITEM>
    <TITLE>Vase and Stones</TITLE>
    <ARTIST>Linda Mann</ARTIST>
    <DIMENSIONS>20x30 inches</DIMENSIONS>
    <MATERIALS>Oil</MATERIALS>
    <YEAR>1996</YEAR>
    <DESCRIPTION>Still Life</DESCRIPTION>
    <PREVIEW-SMALL src="burl-s.jpg" width="300"
height="194" alt="Vase and Stones"/>
    <BIDS>
      <BID>
        <PRICE>6000</PRICE>
        <TIME>3:02:22 PM</TIME>
        <BIDDER>Chris</BIDDER>
        <TIMESTAMP>1307</TIMESTAMP>
      </BID>
      <BID>
        <PRICE>5700</PRICE>
        <TIME>2:58:42 PM</TIME>
        <BIDDER>John</BIDDER>
        <TIMESTAMP>1315</TIMESTAMP>
      </BID>
      <BID>
        <PRICE>5600</PRICE>
        <TIME>2:54:32 PM</TIME>
        <BIDDER>Andrew</BIDDER>
        <TIMESTAMP>1308</TIMESTAMP>
      </BID>
      <BID>
        <PRICE>5500</PRICE>
        <TIME>2:48:08 PM</TIME>
        <BIDDER>Chris</BIDDER>
        <TIMESTAMP>1307</TIMESTAMP>
      </BID>
      <BID>
        <PRICE>5000</PRICE>
        <TIME>2:47:58 PM</TIME>
        <BIDDER>opening price</BIDDER>
        <TIMESTAMP>1298</TIMESTAMP>
```

ie5.xsl
ie4.xsl
nav3.xsl

nokia.xsl
sony.xsl

edi_x.xsl
sap_y.xsl
flat_z.xsl

SEMIDE

EMWIS

File Edit Format Help

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

  <ITEM>
    <TITLE>Vase and Stones</TITLE>
    <ARTIST>Linda Mann</ARTIST>
    <DIMENSIONS>20x30 inches</DIMENSIONS>
    <MATERIALS>Oil</MATERIALS>
    <YEAR>1996</YEAR>
    <DESCRIPTION>Still Life</DESCRIPTION>
    <PREVIEW-SMALL src="burl-s.jpg" width="300"
height="194" alt="Vase and Stones"/>
    <BIDS>
      <BID>
        <PRICE>6000</PRICE>
        <TIME>3:02:22 PM</TIME>
        <BIDDER>Chris</BIDDER>
        <TIMESTAMP>1307</TIMESTAMP>
      </BID>
      <BID>
        <PRICE>5700</PRICE>
        <TIME>2:58:42 PM</TIME>
        <BIDDER>John</BIDDER>
        <TIMESTAMP>1315</TIMESTAMP>
      </BID>
      <BID>
        <PRICE>5600</PRICE>
        <TIME>2:54:32 PM</TIME>
        <BIDDER>Andrew</BIDDER>
        <TIMESTAMP>1308</TIMESTAMP>
      </BID>
      <BID>
        <PRICE>5500</PRICE>
        <TIME>2:48:08 PM</TIME>
        <BIDDER>Chris</BIDDER>
        <TIMESTAMP>1307</TIMESTAMP>
      </BID>
      <BID>
        <PRICE>5000</PRICE>
        <TIME>2:47:58 PM</TIME>
        <BIDDER>opening price</BIDDER>
        <TIMESTAMP>1298</TIMESTAMP>
```

table.xsl

bar.xsl

art.xsl

Price	Time	Bidder
$6000	3:02:22 PM	Chris
$5700	2:58:42 PM	John
$5600	2:54:32 PM	Andrew
$5500	2:48:08 PM	Chris
$5000	2:47:58 PM	opening price

Vase and Stones by Linda Mann

Chris *(3:02 PM)* $6000

John *(2:58 PM)* $5700

Andrew *(2:54 PM)* $5600

Chris *(2:48 PM)* $5500

$5000 opening price *(2:47 PM)*

Vase and Stones

Linda Mann

Size: 20x30 inches *Oil, 1996*

High bid: $5700 (John)

Opening bid: $5000

SEMIDE
EMWIS

Example of a transformation: XML doc

- We want to transform an XML document into HTML
- XML document:

```
<?xml version="1.0"?>
<greeting>
Hello everybody!
</greeting>
```

SEMIDE
EMWIS

Example: the stylesheet

```
<xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
<xsl:output method="html"/>
<xsl:strip-space elements="*"/>
<xsl:template match="/">
    <xsl:apply-templates select="greeting"/>
</xsl:template>
<xsl:template match="greeting">
    <html>
      <body>
        <h1>
          <xsl:value-of select="."/>
        </h1>
      </body>
    </html>
</xsl:template>
</xsl:stylesheet>
```

SEMIDE
EMWIS

Stylesheet output

- Our stylesheet has produced a complete and valid HTML document, readable with a browser

```
<html>
<body>
<h1>Hello everybody!</h1>
</body>
</html>
```

SEMIDE
EMWIS

Analysis of the example stylesheet

```
<xsl:stylesheet
xmlns:xsl="http://www.
    w3.org/1999/XSL/Tran
    sform"
version="1.0">
<xsl:output
    method="html"/>
<xsl:strip-space
    elements="*"/>
```

- The elements in the top part of the example remain the same in most stylesheets
- Other XSL-T elements can be used to control the output type, specify an output DTD, etc.
- The XSL-T elements are defined in the XSL-T specifications (http://www.w3c.org/TR/xslt)

SEMIDE
EMWIS

Analysis of the example stylesheet

```
<xsl:template match="/">
    <xsl:apply-templates
  select="greeting"/>
</xsl:template>

<xsl:template
  match="greeting">
    <html>
      <body>
        <h1>
          <xsl:value-of
  select="."/>
        </h1>
      </body>
    </html>
</xsl:template>
```

- The first template in the stylesheet is the root template
- The stylesheet processor requires this template first

Example:
a different approach

```
<xsl:template match="/">
   <html>
      <body>
         <xsl:apply-templates select="greeting"/>
      </body>
   </html>
</xsl:template>
```

- The root template may also be written as shown

Summary

- An XSL-T stylesheet contains a certain number of <template> that describe the transformations of a document
- The XSL-T elements define the transformations, while the XPath expressions define which data has to be transformed
- The task is carried out by an XSL-T processor that transforms the XML document according to the rules provided by the stylesheet

XML Application architecture

- There are different means of recalling the stylesheet processor in an application:
  - The stylesheet processor may be recalled on the server, so that it transforms the documents and sends them to the clients
  - The stylesheet processor may be recalled from the client executing the transformation
  - A “precompiled” schema may also be used that transforms and places the more common documents in the cache before the clients request them

Xpath data model

- XSL-T and Xpath treat an XML document as a node tree
- The stylesheet deals with the processed nodes, not the XML source

Moving within the tree

- Each Xpath expression has a point of reference called current node or context node
- When writing Xpath expressions, knowing the whereabouts of the current node is essential
- There are 13 ways of moving around within the tree from the current node

Moving within the document

- Each of these 13 axes specifies a different node with a specific relation to the current node
- The most common relations are:
  - child – the first level of nodes below the current node
  - descendant – all nodes of all levels below and branching from the current node
  - parent – the node from which the current node branches
  - ancestor – all nodes of all levels from which the current node branches

SEMIDE
EMWIS

Examples of XSL-T and XPath

<xsl:template match="/">
- template for the root element

< xsl:template match="b">
- template for all <b> elements

<xsl:template match="h1/b">
- template for all <b> elements contained in <h1> elements

<xsl:template match="b[1]">
- template for the first <b> element

<xsl:template match="b[position()=last]">
- template for the last <b> element

<xsl:template match="b[7]|b[9]">
- template for the seventh or the ninth <b> element

SEMIDE
EMWIS

Examples of XSL-T and XPath

<xsl:apply-templates select="b" />

- applies the XSL-T templates to all <b> tags contained in the current element. The XSL-T processor determines which template should be applied to each

<xsl:value-of select="."/>

- returns the value of the current element, sub-elements included

<xsl:text>insertion</xsl:text>

- inserts the text insertion in the output

<xsl:element name="a">

- creates a <a> element in the output. The name value may be a variable

<xsl:attribute name="href">

- creates a href attribute in the current element; the value of the <xsl:attribute> element is the created attribute text

Test and condition elements

<xsl:if>

- specifies the XSL-T elements that must be processed if the condition is true
  - E.g.:

```
<xsl:if test="count(telephone) > 0">
    <xsl:apply-templates select="telephone"/>
</xsl:if>
```

  - *count* is an Xpath function that returns the number of elements of a certain type
  - note: it is not planned to add an *else* function

Test and condition elements

<xsl:for-each>

- Executes a loop through all nodes corresponding to the select command
  - E.g.

```
<xsl:for-each select="telephone">
    <xsl:apply-templates select="prefix"/>
</xsl:for-each>
```

- Beware: *select* changes the current node!

Test and condition elements

<xsl:choose>

- similar to the select command of lots of programming languages
- is used to implement an if-then-else command
  - structure (one or more *when*, only one *otherwise*):

```
<xsl:choose>
    <xsl:when test="…">
    …
    </xsl:when>
    <xsl:when test="…">
    …
    </xsl:when>
    <xsl:otherwise>
    …
    </xsl:otherwise>
</xsl:choose>
```

SEMIDE
EMWIS

Functions enabling document integration

- The XSL-T specifications contain a little used very powerful function called document()
- document() allows the writing of a single XML document that imports other documents, then the writing of a stylesheet that transforms all of them

SEMIDE

EMWIS

SEMIDE

http://www.semide.org

EMWIS

http://www.emwis.org

Producer support for XML

EURO-MEDITERRANEAN PARTNERSHIP

Producer support for XML

Producer initiatives in agreement with the evolution of B2B web-based services from a

- presentation-oriented approach (Web pages)
- to an automation-oriented approach (Web services)

SEMIDE
EMWIS

XML proposals made by some of the main producers

• IBM	• (WebSphere)
• Informix	• (i.Sell)
• Microsoft	• (BizTalk)
• Oracle	• (8i)
• Software AG	• (Tamino)
• Sun	• (Forte)

IBM - WebSphere

- The WebSphere platform is the development of a family of Application servers that has become the end-to-end software offering for e-business
- It is a middleware for the Web composed of three different levels:
  - WebSphere Foundation
  - WebSphere Foundation Extension
  - Application Accelerator

SEMIDE
EMWIS

What is WebSphere?

- A software infrastructure supporting all e-business development phases that also include XML support
- WebSphere Foundation is the environment in which web applications are executed and provides support for their integration with the existing applications:
  - Application Servers + MQSeries
- WebSphere Foundation Extension provides integrated tools and services, necessary for the growth and adaptation of company processes to new e-business products:
  - Presentation services (Portal server, voice server, ...)
  - Development services (VisualAge for Java, ..)
  - Deployment services (Edge Server,Host Integration, ...)
- Application Accelerator is composed of modular and extensible applicative services:
  - CommerceSuite, B2B Integrator, LotusDomino, MQSeries WorkFlow

IBM - WebSphere

WebSphere Software Platform

CUSTOMER AND PARTNER APPLICATIONS

B2E B2C B2B Process Automation e-Markets

APPLICATION ACCELERATORS

WebSphere Commerce Suite	Lotus Domino	MQSeries Workflow	WebSphere B2B Integrator

FOUNDATION EXTENSIONS

Development
- WebSphere Homepage Builder
- VisualAge for Java
- VisualAge Application Rules
- WebSphere Studio
- WebSphere Business Components
- VisualAge Generator

Presentation
- WebSphere Transcoding Publisher
- WebSphere Voice Server
- WebSphere Portal Server
- WebSphere Everyplace Suite
- WebSphere Personalization
- Web Content Management

Deployment
- WebSphere Site Analyzer
- WebSphere Edge Server
- MQSeries Integrator
- Tivoli Policy Director
- WebSphere Host Integration

FOUNDATION

Web Application Serving and Integration

- WebSphere Application Servers
- MQSeries

SEMIDE
EMWIS

Informix - i.Sell

- The Informix offering that combines Java and object-relational technology for the e-business market
- Enables integration with other B2B environments using its XML language support
- It consists mainly of three modules:
  - i.Sell Personalizer
  - i.Sell Merchandiser
  - i.Sell Application Server
- Current version 2.1

SEMIDE
EMWIS

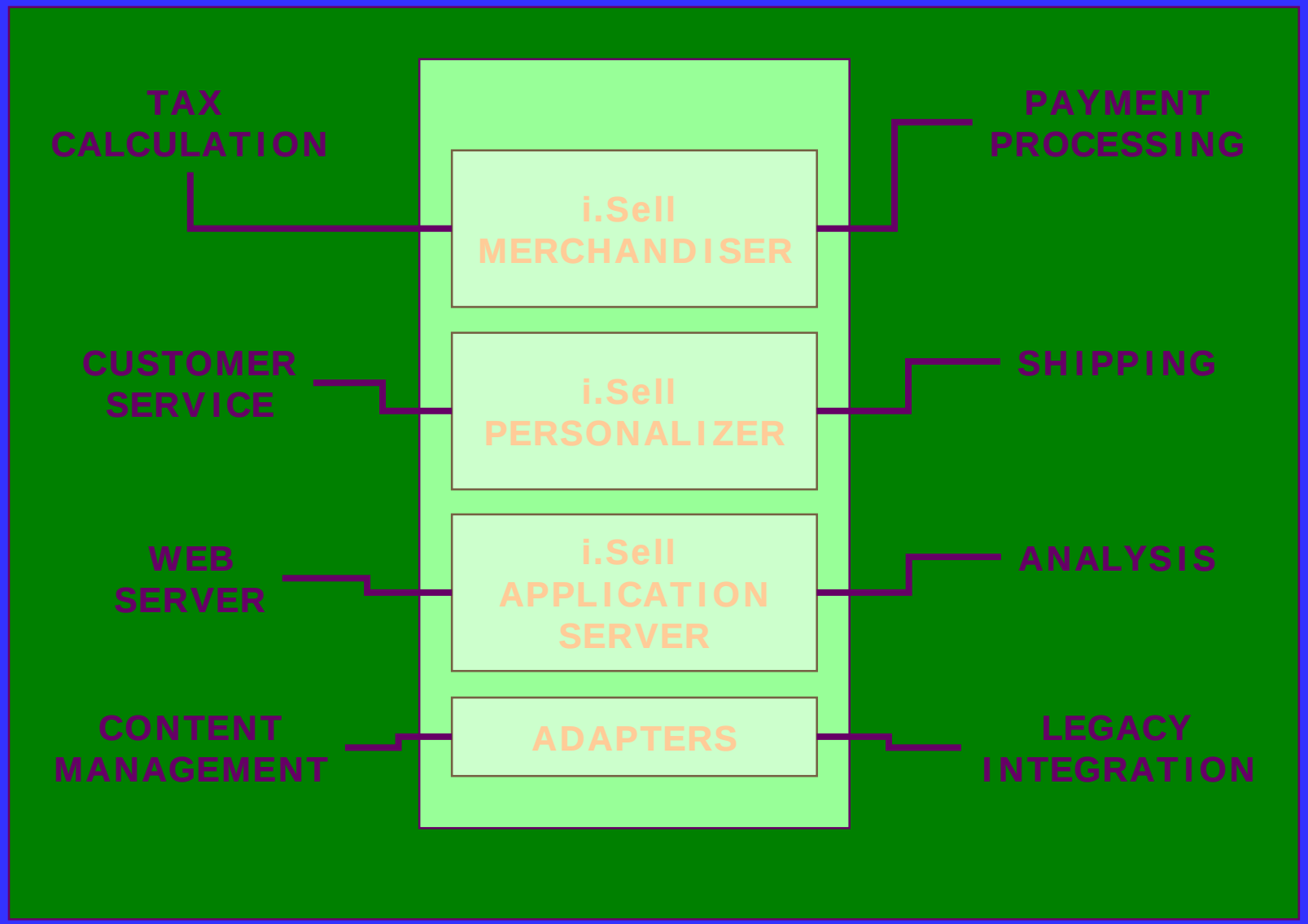
What is i.Sell?

i.Sell adopts an e-business model called PTA

- P = Publish
  - The i.Sell Personalizer tool that allows marketing and business managers to define the personalisation rules controlling the promotions and contents addressed to groups or individuals
- T = Transact
  - The i.Sell Merchandiser tool that deals with executing on-line transactions (e.g. ordering a product from a catalogue)
- A = Analyze
  - This part of the model is represented by intelligent commercial “sensors” capable of tracing the user’s behaviour and automatically carrying out actions, thus proposing contents that are more attractive and suited to the client’s needs

SEMIDE
EMWIS

Informix - i.Sell



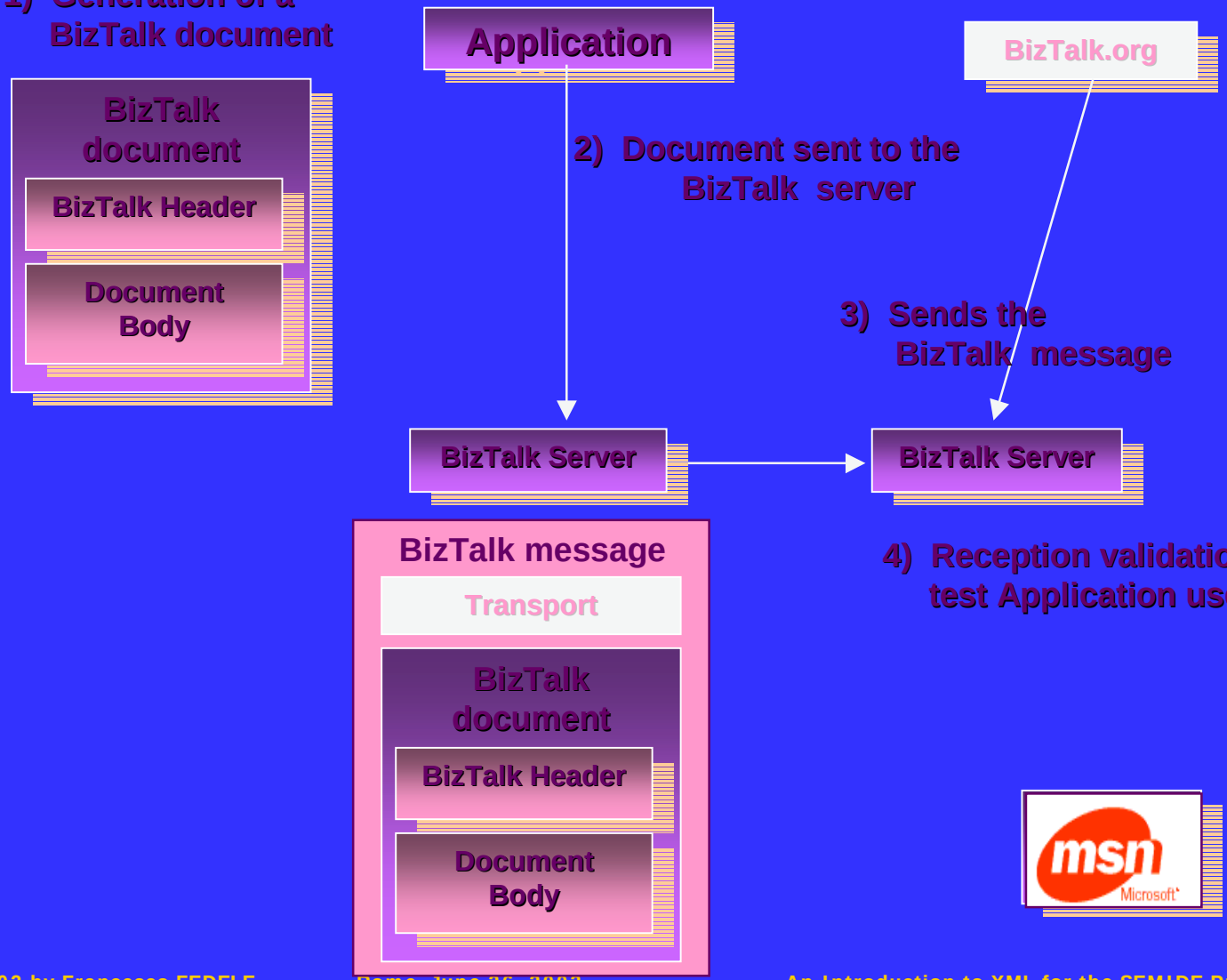
BizTalk – XML framework

- BizTalk is an industrial initiative proposed by Microsoft with the support of a large range of organisations such as SAP and CommerceOne
- BizTalk is NOT a standard
- It is a community of users that, while respecting the standards, pursue the objective of facilitating the rapid adoption of XML for e-commerce development and application integration
- It establishes a series of “tag standards” or DTDs capable of formatting the necessary documents for e-business activities (orders, invoices etc.) so that they become recognisable using machines, applications and various producer application servers and web sites

What is BizTalk ?

- An XML server
- It is also the XML framework that allows the writing of transactional applications
- It establishes a group of XML schema (XDR) products by Microsoft, partners and standardisation groups
- Enables well-defined services for data exchange and integrates them with the existing applications

Dialogue between applications using BizTalk



SEMIDE
EMWIS

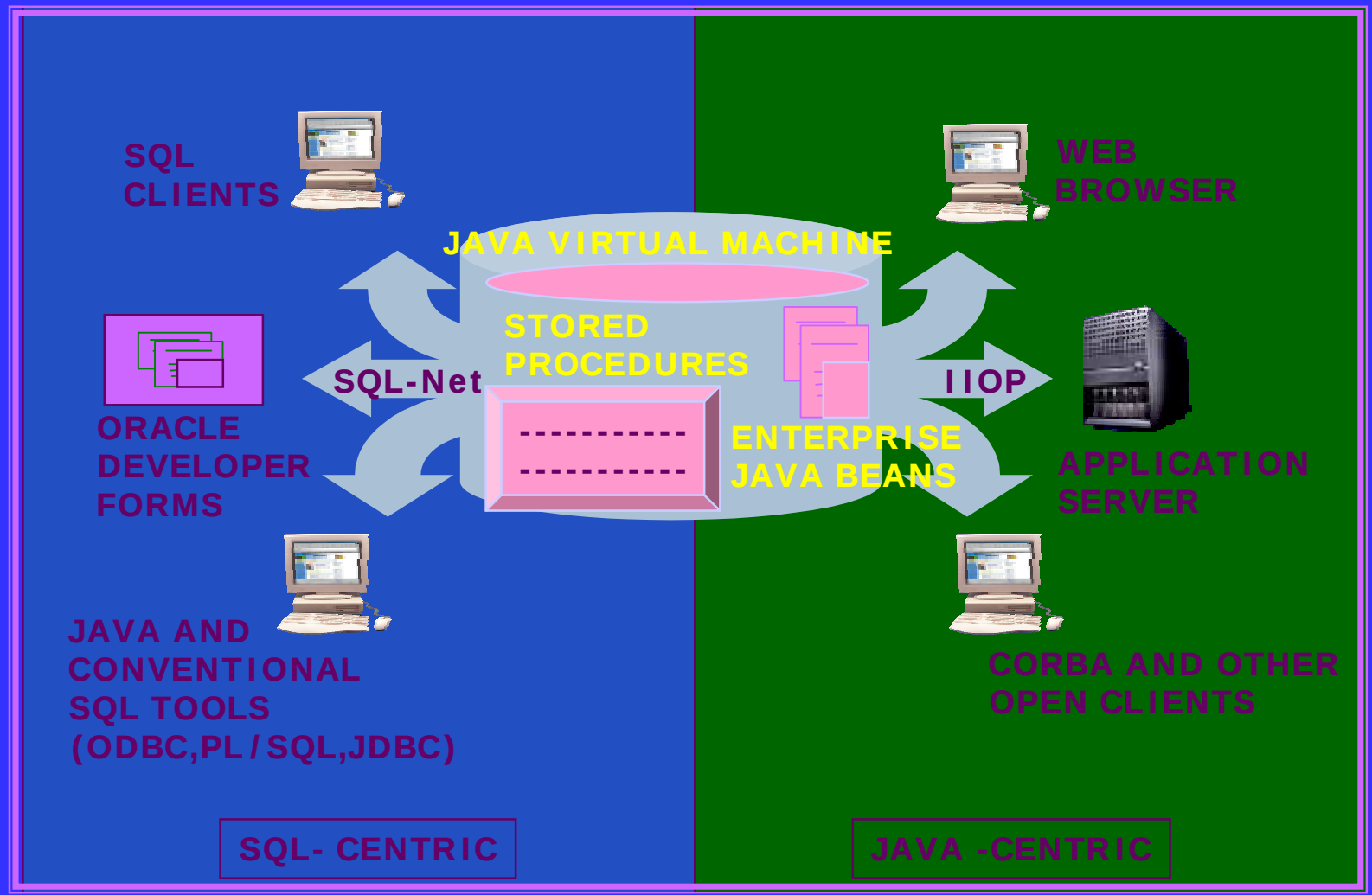
Oracle - 9i

- Oracle Internet Platform
  - RDBMS 9i
  - Internet Application Server (IAS)
    - data-centric architecture
    - Portal Services
    - Data Caching
    - Apache Web Server
    - business intelligence services
    - lite version also available
  - Internet Developer Suite
    - Jdeveloper
    - Oracle Designer
    - Forms Developer
    - Portal
    - Discoverer
    - Reports

What is Oracle 9i?

- A RDBMS + Internet Application Server
  - internet enabled
  - java based, a new JVM has been created
  - Java Stored Procedures support enabled
- Supports Java 2 Enterprise Edition APIs (with Java Messaging Services, Java Transaction Services and JDBC), Java Servlets , JSP
- Offers IIOP protocol support
- Offers XML support through a parser

Oracle 9i - Architecture



Software AG - Tamino

Transaction Architecture for the Management of InterNet Object

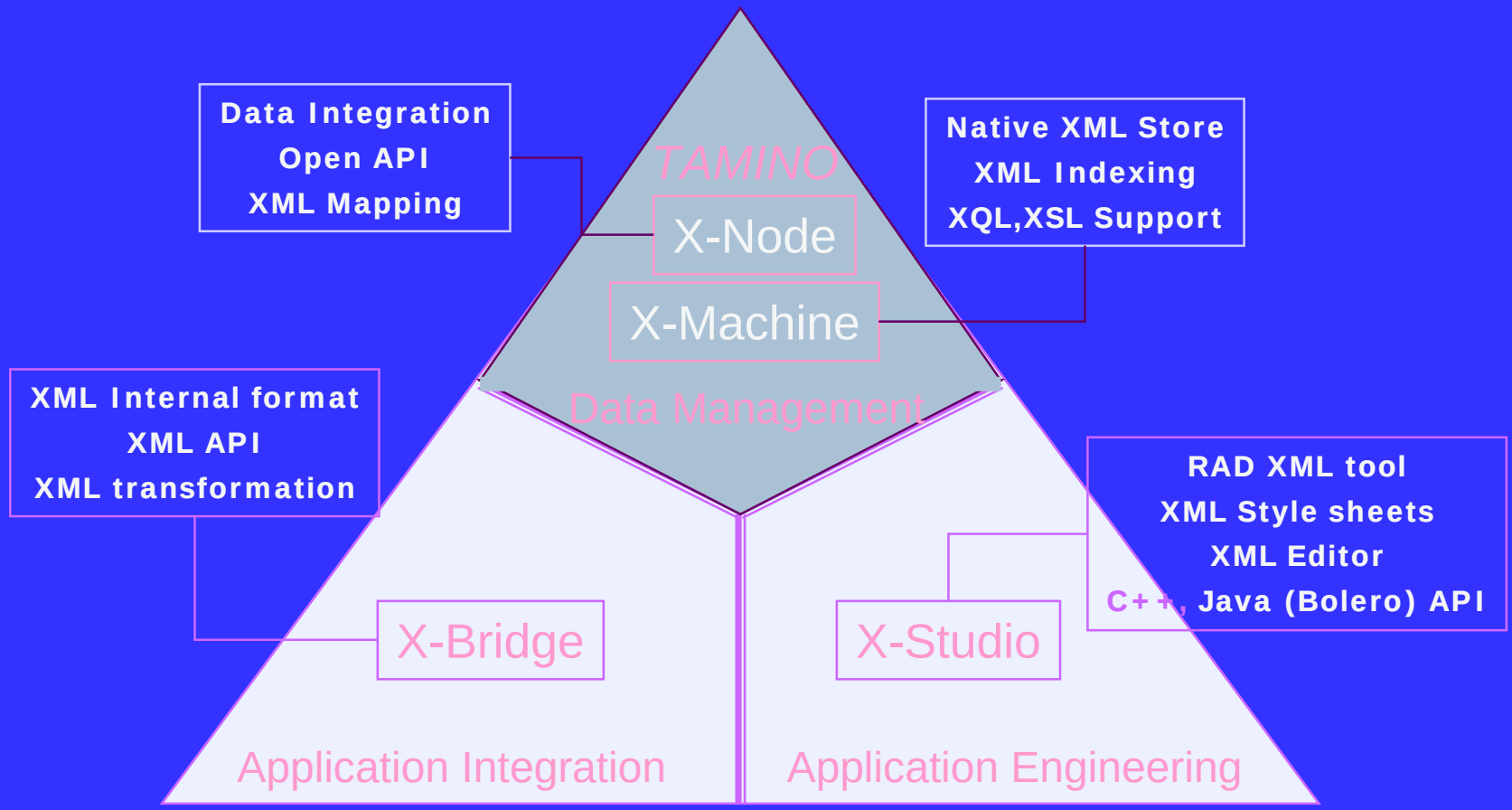
- This is the Software AG offering for XML-based data management
- Allows information archiving in native XML format
- Works using the XQL and SQL query languages
- It is an integral part of a framework (Xenon)
  - for the development and management of XML-based e-business applications
  - for their integration with existing applications

What is Tamino?

- A DBMS allowing data management in native XML format through X-Machine,
- X-Machine provides for XML tag indexing and incorporates either the parser or the research system, guaranteeing SQL support
- A data representation and management schema has been created based on the nature of XML:
  - hierarchical
  - document oriented
- The X-Node component provides the application access interface to the database, enabling the management of information from different origins in a homogenous structure

SEMIDE
EMWIS

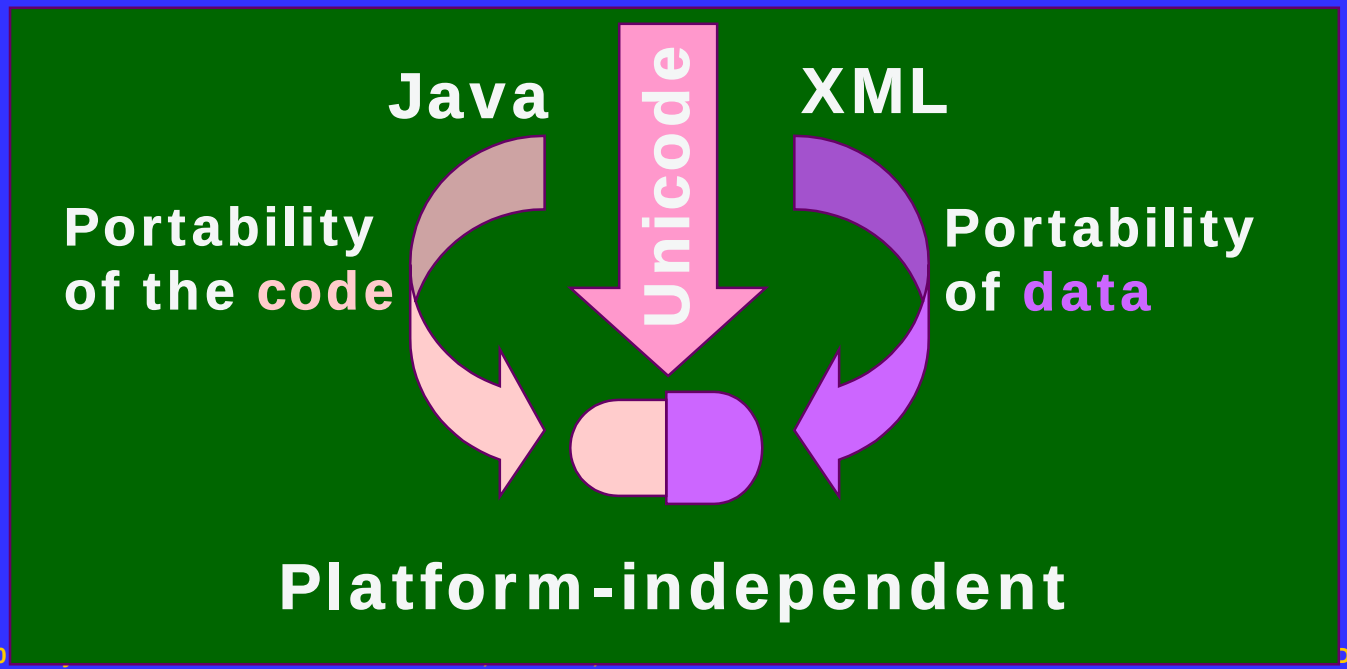
The Software AG proposal



Sun - Initiatives related to XML

- No specific XML offering, the company is however devoted to various initiatives in line with the motto:

  "XML is to data what Java is to code"



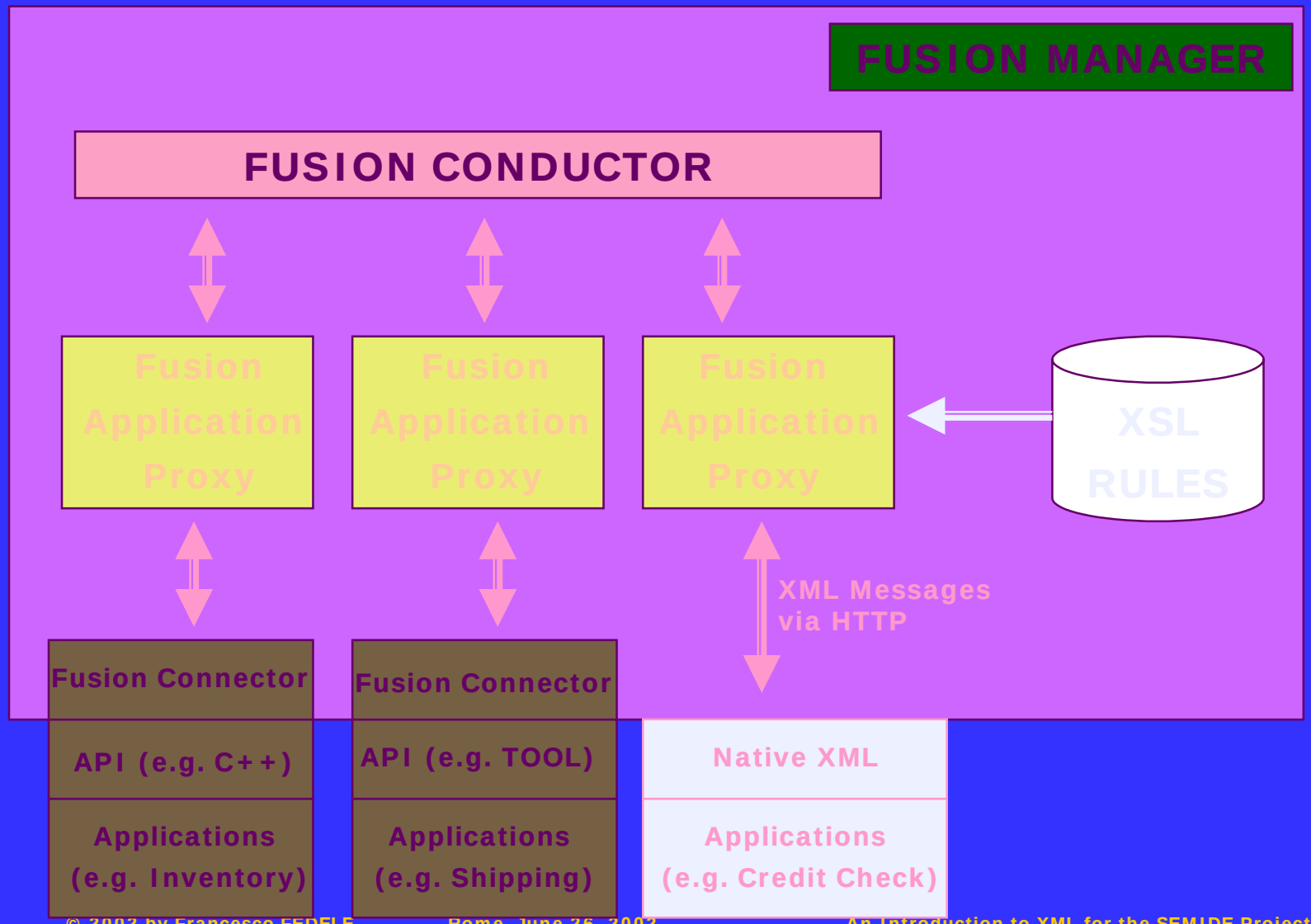
SEMIDE
EMWIS

What is Fusion?

- A suite that allows
  - point-to-point integration (application-to-application)
  - message brokering (based on Java Messaging)
  - integration of the business processes involving external systems, Custom applications and external services
- Using XML as universal native form
- Capable of providing support and integration to the ever-growing number of XML dialects (e.g. BizTalk, RosettaNet, OASIS,..)
- Capable of interacting with the most widespread DBMS

SEMIDE
EMWIS

Forte Fusion/Sun Forte



SEMIDE

EMWIS

SEMIDE

http://www.semide.org

EMWIS

http://www.emwis.org

XML support available on line

EURO-MEDITERRANEAN PARTNERSHIP

SEMIDE
EMWIS

XML support available on line

- Institutional sites related to the standards
- Sites related to the various commercial and non-commercial initiatives
- Technical sites providing tutorials of different levels
- Sites specialised in tool offers
- Commercial sites

SEMIDE
EMWIS

XML sites related to standards and initiatives

- http://www.w3c.org/xml
- XML related sites
  - http://www.xml.org
  - http://www.xml.com
  - http://www.xmlitaly.it
  - http://www.xml.it
- Sites concerning initiatives related to XML
  - www.biztalk.org
  - www.ebxml.org
  - www.goxml.com
  - www.commerceone.com
  - www.ariba.com
  - www.xmledi.com
  - www.rosettanet.org
  - www.xmlglobal.com

XML: freeware

- http://www.xmlsoftware.com/
  - XMLSOFTWARE aims to provide well organised information and resources on the Extensible Markup Language (XML), one of the most significant developments on the World Wide Web and in electronic publishing and electronic commerce
- sister sites
  - XMLINFO (http://www.xmlinfo.com/)
  - SCHEMA.NET (http://www.schema.net/)
  - XSLINFO (http://www.xslinfo.com/)

SEMIDE
EMWIS

Sites providing technical support or not

- http://www.oasis-open.org/cover/
  - the XML Cover Pages is a comprehensive online reference work for the Extensible Markup Language (XML) and its parent, the Standard Generalized Markup Language (SGML)
    - Publications
    - News
    - Software (Tools XML,XSL and others)
    - Introductions
    - Support
    - XML, XSL, XLink
    - Events
    - Related Standards
    - Special Topics
    - Application Standards

Sites providing support

- http://www.cs.caltech.edu/~adam/LOCAL/xml.html
- http://www.sun.com/
- http://www.ibm.com/developer/xml/
- http://msdn.microsoft.com/xml
- http://www.xmltree.com/
- http://metalab.unc.edu/xml
- http://www.projectcool.com/developer/xmlz

SEMIDE
EMWIS

An important international initiative: ebXML

- ebXML, a UN/CEFACT (United Nations body for Trade Facilitation and Electronic Business) and OASIS joint project
  - aim: to develop a framework allowing the use of XML in a standardised way for electronic business data exchange.
  - the 18 month work schedule is based on EDI and EDIFACT for the semantic aspects and on the existing work, carried out by CommerceOne (CBL) and the eCo framework, for the XML framework.
- First phase finished May 2001, new phase ongoing: new focus on messages (end 2002 - summer 2003)

SEMIDE

EMWIS

SEMIDE

http://www.semide.org

EMWIS

http://www.emwis.org

The XML market

EURO-MEDITERRANEAN PARTNERSHIP

SEMIDE
EMWIS

XML: application areas

- Content management
  - presentation-oriented publishing
  - one common data format
  - multiple rendering styles (XSL)
- Data interchange/EDI
  - data interchange / EDI
  - interfacing of heterogeneous products
  - inter-process communication (IPC)
- Application integration
  - application-to-application communication
  - Internet message formats (protocols)
  - client/middle tier/server
- Data aggregation/portal
  - enterprise information portals

SEMIDE
EMWIS

What can be done with XML?

- Data/document archiving
  - easy recognition of data significance, even later in time
- Dynamic research capacity
  - Dynamic creation of the pages showing search results
  - Use of more search criteria
- Personalisation of information viewing
  - Each user can choose different viewing modes, according to personal preferences
- Information viewing automation
  - Information is viewed based on XSL stylesheets
  - Changing the stylesheet also changes the display

SEMIDE
EMWIS

What is XML used for?

- Advanced publishing
  - allows automatic access to the relevant information for the user
  - personalisation
- Integration with Databases
  - modifying data using the web without losing the original format of the data
  - data may be directly translated into HTML
  - connection with existing database technologies
    - unlike HTML, the database results may be processed locally

What is XML used for?

- Web automation, e.g. automatic downloading of information blocks
  - WebMethods has developed a language (WIDL) for web automation applications
  - WIDL uses XML to capture the details of a web communication in a format that it can automate
  - the WebMethods software then acquires the information and automatically generates the necessary script for the creation of an automated application

SEMIDE
EMWIS

What is XML used for?

- Online banking
  - Information storage – account number #, bank identification, account type, availability, etc.
  - Inter-bank standardisation – enables information transfer
- Software distribution- downloads only the necessary components during setup
- Standards for software distribution via the Internet
- E-commerce

SEMIDE
EMWIS

Benefits and disadvantages of using XML

BENEFITS

- User–oriented data presentation
  - The XML+XSL combination
    - allows the separation of business logic and presentation logic
    - frees the application from links related to presentation needs
- Data exchange between applications
  - inter-application integration is possible with a fraction of the effort that is traditionally required for EDI
- Direct XML data publishing
  - the format that is readable by the machine (UNICODE) may be combined with other data and developed later on (impossible with HTML)

DISADVANTAGES

- Evolving basic standards and technology
  - the organisations in charge of defining the standards are under pressure from the commercial initiatives
- Slow adoption process

SEMIDE
EMWIS

Benefits and disadvantages of using XML

BENEFITS

- User–oriented data presentation
  - The XML+XSL combination
    - allows the separation of business logic and presentation logic
    - frees the application from links related to presentation needs
- Data exchange between applications
  - inter-application integration is possible with a fraction of the effort that is traditionally required for EDI
- Direct XML data publishing
  - the format that is readable by the machine (UNICODE) may be combined with other data and elaborated later on (impossible with HTML)

DISADVANTAGES

- Basic standards and technology in evolution
  - the organisations in charge of defining the standards are under pressure from the commercial initiatives
- Slow adoption process

SEMIDE EMWIS

SEMIDE

http://www.semide.org

EMWIS

http://www.emwis.org

Case study

Examples of application of XML, in particular to web sites

Live demo

EURO-MEDITERRANEAN PARTNERSHIP

SEMIDE EMWIS

SEMIDE

http://www.semide.org

EMWIS

http://www.emwis.org

Suggestions for use of XML in SEMIDE

Suggestions and discussion

EURO-MEDITERRANEAN PARTNERSHIP

SEMIDE
EMWIS

XML to exchange data

- Use XML to collect data for use by the National Focal Point
- Use XML to make data from a National Focal Point available to the International Focal Point
- Use XML to query databases
- Use the Web Services architecture
  - To collect data
  - To make data available to the project
  - To make data available outside the project
- Use XML as the data format for (all or some) project data

SEMIDE
EMWIS

XML to publish information

- Use XML for the content, use XSLT to transform to HTML to publish the content on the web
- Use different XSLTs to pubblish the same content in different ways and with different details
- Use XSLT to produce documents and other media from the same content
- Automatically generate content from sources
- Send updates of content to the International Focal Point

SEMIDE

EMWIS

SEMIDE

http://www.semide.org

EMWIS

http://www.emwis.org

Francesco FEDELE

francesco.fedele@cminfoland.com

fedele@effedue.com

Office +39 06 50932505

Mobile +39 329 0525896

EURO-MEDITERRANEAN PARTNERSHIP

SEMIDE

EMWIS