

XML World Euro Edition

March 26 - 28, 2001
Amsterdam



Developing XML-Enabled Java™ Applications -- How to Manage and Persist XML Objects in Java

*Frank Thelen
Product Group Executive
POET Software
frank.thelen@poet.de*

Agenda

- ◆ Documents vs. Data
- ◆ Schemata
- ◆ APIs

XML - Application Scenarios

- ◆ Document Management
 - ◆ XML Documents
 - ◆ Publishing, Presentation
 - ◆ Teamwork, Workflow, Versioning
- ◆ Data Management
 - ◆ XML Data
 - ◆ Data Transfer, Communication, Messaging, Protocol

XML - Documents vs. Data - Examples

Documents

- ◆ magazine article
- ◆ magazine ad
- ◆ book
- ◆ email
- ◆ letter
- ◆ press release
- ◆ whitepaper
- ◆ web page (XHTML)

Data

- ◆ product catalog
- ◆ invoice
- ◆ order
- ◆ flight schedule
- ◆ appointments in calendar

XML - Documents vs. Data

Documents

- ◆ Irregular and roughly structured content
- ◆ No type definitions
- ◆ "Content model"
- ◆ Large amount of "mixed content"
- ◆ Significance in the sequence of elements

Data

- ◆ Well-ordered, standardized, and more finely structured content
- ◆ Type definitions
- ◆ Schemata
- ◆ Little or no "mixed content"
- ◆ No significance in the sequence of elements

XML Data

```
<?xml version="1.0"?>
<Catalog>
  <Name>OSS Product Catalog</Name>
  <Release>1.0</Release>
  <Date>20001019</Date>
  <ProductList>
    <Product>
      <ProductCode>SW-5110-00</ProductCode>
      <ProductName>SDK for Windows</ProductName>
      <Platforms>Windows 9x/NT/2000</Platforms>
      <Price Currency="USD">248</Price>
    </Product>
    <Product>
      ...
    </Product>
  </ProductList>
</Catalog>
```

XML - Why Schemata?

- ◆ XML
 - ◆ Portable Documents/Data
 - ◆ Unconstrained Documents/Data
 - ◆ Portable Constraints
- ◆ DTD
 - ◆ Document Structures
- ◆ Schemata
 - ◆ Data Structures

Document Type Definition (DTD)

- ◆ Syntax: special
- ◆ Status:
 - ◆ Part of XML 1.0
(W3C Recommendation, 10 Feb 1998)
- ◆ Properties:
 - ◆ Similar to DTD for SGML
 - ◆ Description of Content Models
 - ◆ Very limited Datatypes
 - ◆ Not sufficient for "Data Management"

XML Schema (XSD)

- ◆ Syntax: XML 1.0
- ◆ Status:
 - ◆ W3C Candidate Recommendation, 24 Oct 2000
- ◆ Properties:
 - ◆ Separation of Elements and Type System
 - ◆ Simple Types and Complex Types
 - ◆ Derived Types (Inheritance, Polymorphism)
 - ◆ Anonymous and abstract Types/Elements
 - ◆ Keys, Unique Keys, References
 - ◆ Namespaces

APIs

- ◆ Generic Approaches
 - ◆ Abstraction: Meta Model
 - ◆ Document, Element, Entity, Node, ...
 - ◆ Application: Document Management
 - ◆ APIs: DOM, SAX
- ◆ Non-generic Approaches
 - ◆ Abstraction: Problem/Solution Domain
 - ◆ Order, Address, Stock, Product, ...
 - ◆ Application: Data Management
 - ◆ APIs: Data Binding

DOM ("Document Object Model")

- ◆ W3C Specification
- ◆ Platform- and language-neutral Interface
- ◆ Language Bindings
 - ◆ Java, C++, COM, JavaScript, Perl, ...
- ◆ Implementations
 - ◆ Xerces, XML4J, Project X, MSXML, ...
- ◆ Tree-based API
 - ◆ Node, Document, Element, etc.

DOM - Usage

- ◆ Usage
 - ◆ Manipulation and Navigation of XML instances
 - ◆ If tasks depend on context
- ◆ Pros
 - ◆ Convenient, powerful
 - ◆ Complete context
 - ◆ Read/write interface
- ◆ Cons
 - ◆ High resource consumption

DOM - Java Language Binding

- ◆ Packages

- ◆ `org.w3c.dom`

- ◆ Interface Hierarchy

- ◆ `org.w3c.dom.Node`
 - `org.w3c.dom.Attr`
 - `org.w3c.dom.CharacterData`
 - `org.w3c.dom.Document`
 - `org.w3c.dom.DocumentType`
 - `org.w3c.dom.Element`
 - `org.w3c.dom.Entity`
 - ...
- `org.w3c.dom.NodeList`
- ...

DOM - Java Platform Support

- ◆ JSR-63: Java API for XML Processing
 - ◆ Status: Release 1.1, Feb 2001
 - ◆ DOM Support: DOM Level 2 Core
 - ◆ Optional Package for J2SE/J2EE ("JAXP")
- ◆ Packages:
 - ◆ `javax.xml.parsers`

DOM - Coding

- ① Instantiate DOM parser
- ② Parse XML instance
- ③ Navigate/Manipulate DOM tree

SAX ("Simple API for XML")

- ◆ Developed by XML-DEV mailing list
- ◆ Platform- und language-neutral Interface
- ◆ Language Bindings
 - ◆ Java, C++, COM, JavaScript, Perl, ...
- ◆ Implementations
 - ◆ Xerces, XML4J, Project X, MSXML, ...
- ◆ Event-based API
 - ◆ Begin/End of Elements, etc.

SAX - Usage

- ◆ Usage
 - ◆ Large Documents
 - ◆ If tasks do not depend on context
 - ◆ For your own data structures
- ◆ Pros
 - ◆ Efficient
- ◆ Cons
 - ◆ Less convenient
 - ◆ No write-interface
 - ◆ Possible loss of information

SAX - Java Language Binding (SAX2)

◆ Packages

- ◆ `org.xml.sax`
- ◆ `org.xml.sax.helpers`
- ◆ ...

◆ Interface

- ◆

```
public interface org.xml.sax.ContentHandler {  
    public void startDocument(...);  
    public void endDocument(...);  
    public void startElement(...);  
    public void endElement(...);  
    public void characters(...);  
    ...  
}
```

SAX - Java Platform Support

- ◆ JSR-63: Java API for XML Processing
 - ◆ Status: Release 1.1, Feb 2001
 - ◆ SAX Support: SAX 2.0
 - ◆ Optional Package for J2SE/J2EE ("JAXP")
- ◆ Packages:
 - ◆ `javax.xml.parsers`

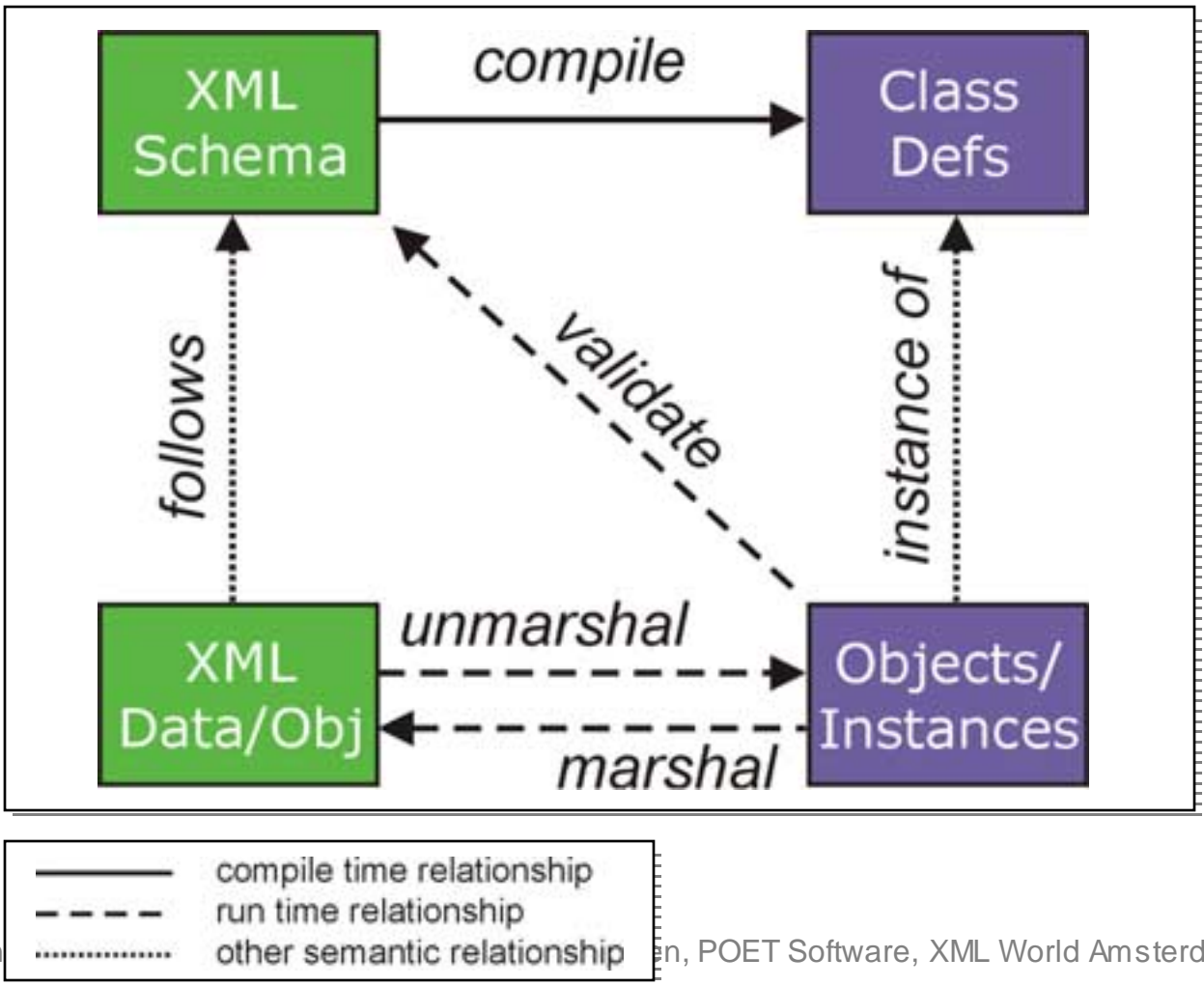
SAX - Coding (SAX2)

- ① Instantiate SAX Parser
 - ◆ `XMLReader` (formerly `Parser`)
- ② Instantiate `ContentHandler`
 - ◆ `ContentHandler` (formerly `DocumentHandler`)
- ③ Register `ContentHandler`
- ④ Parse XML instance
 - ◆ Handle Callbacks

APIs - Non-Generic Approach

- ◆ "XML Data Binding"
- ◆ Mapping of Type Systems
XML-Schema \leftrightarrow Java
 - ◆ Base Types
 - ◆ User Defined Types (Classes)
- ◆ Mapping of Objects
XML Instance \leftrightarrow Java Objects

XML Data Binding - Concept



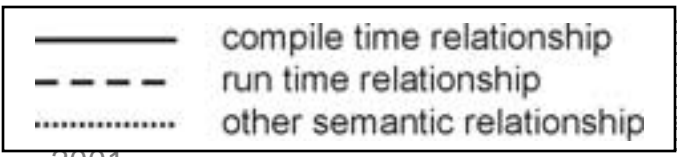
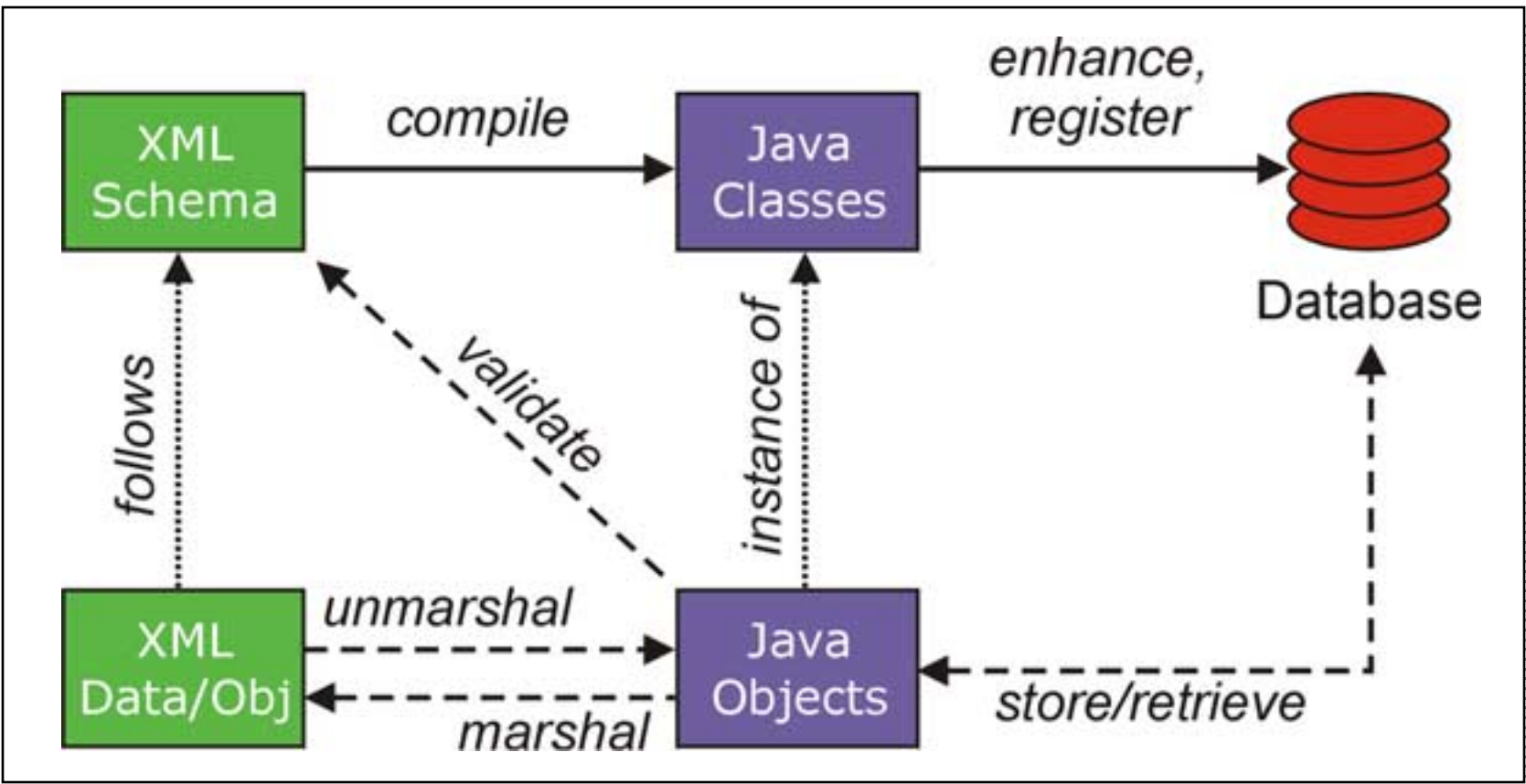
XML Data Binding - Pros/Cons

- ◆ Pros
 - ◆ Data model leads to corresponding classes
 - ◆ Specialized implementations provide higher performance, less overhead
 - ◆ User's choice of persistence model
- ◆ Cons
 - ◆ No standard as yet
 - ◆ Available implementations are still very young
 - ◆ Implementations do not always implement the latest draft

XML Data Binding - Java Platform Support

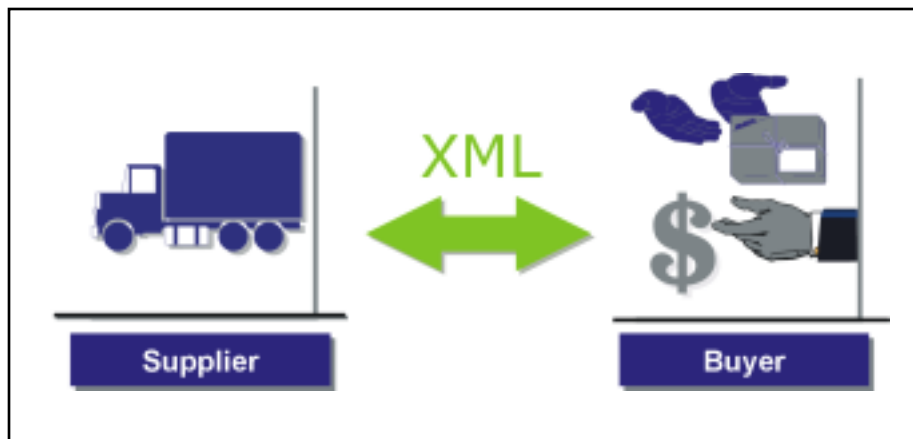
- ◆ JSR-31: XML Data Binding Specification
 - ◆ Code Name "Adelard"
 - ◆ Status:
 - ◆ Q1/2001: First Draft Spec (community review)
 - ◆ Optional Package for J2SE/J2EE ("JAXB")

XML Data Binding - Example



- Frank Thelen, POET Software, XML World Amsterdam, March

XML - Usage



Example: eProcurement

- ◆ Data Interchange
- ◆ XML-based high-level Protocol, e.g.:
 - ◆ Product Catalogue
 - ◆ Order
 - ◆ Order Confirmation
 - ◆ Invoice
 - ◆ ...

Any Questions?

Links (1)

- ◆ XML

- ◆ <http://www.w3.org/XML/>
- ◆ <http://www.xml.org/>
- ◆ <http://www.xml.com/>
- ◆ <http://www.xml-zone.com/>
- ◆ <http://www.javasoft.com/xml/>

- ◆ DOM

- ◆ <http://www.w3.org/DOM/>

- ◆ SAX

- ◆ <http://www.megginson.com/SAX/>

Links (2)

- ◆ Data Binding

- ◆ <http://java.sun.com/xml/docs/binding/DataBinding.html>
- ◆ <http://java.sun.com/xml/docs/bind.pdf>

- ◆ DTD and Schema

- ◆ <http://www.oasis-open.org/>
- ◆ <http://www.w3.org/XML/Schema.html>

- ◆ POET Software

- ◆ <http://www.poet.com/fastobjects/>
- ◆ <http://www.poet.de/fastobjects/>