Six Things I Want in an XML Binding Layer

Blaise Doughan

Team Lead, TopLink OXM
Oracle Corporation
www.oracle.com

Shaun Smith

Product Manager, TopLink
Oracle Corporation
www.oracle.com

Session TS-6224

Goal of This Talk

What You Will Gain

Learn the criteria needed to pick the best XML binding solution

Agenda

Usability

Performance

Full W3C XML Schema Support

Standards Compliance

Compatibility With Other Standards

Compatibility With your SOA

Agenda

Usability

Performance

Full W3C XML Schema Support

Standards Compliance

Compatibility With Other Standards

Compatibility With your SOA

O-X Impedance Mismatch

Factor		Java™ EE	XML
Technical	Logical Data Format	Objects, Methods, Inheritance	XSD, XPATH
	Scale	Hundreds of Megabytes	Depends
	Relationships	Memory References	Generally Use Aggregation
	Uniqueness	Internal Object Identity	Unique Identifier
Business	Key Skills	Java™ development, object modeling	XML, XSD, XPATH, XQUERY
	Tools	IDE, Source code management, Object Modeler	XML Design Tools and Viewers
Political	Corporate Org. Structure	Often integrating or using legacy application code	High expectations of data portability

Java™ Based Access of XML Data

- Direct Java API for XML Processing (JAXP)—window on data
  - Direct use of an XML parser, uses DOM nodes and/or SAX/StAX events directly
- Entity Beans/Business Objects
  - Accessed as objects or components (Enterprise JavaBeans™ specifications), transparent that the data is stored in XML
  - Need binding layer in middle tier to handle the object-XML mapping and conversion

Retrieving Data from XML

DOM Code

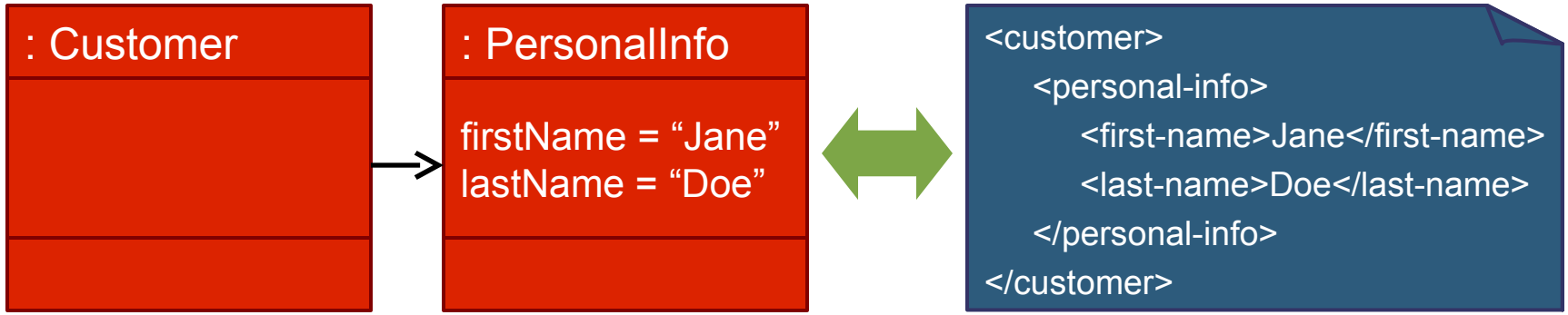
```
Node child = customerElement.getFirstChild();
while(child != null) {
    if(child.getNodeName().equals("first-name")) {
        Node textNode = childNode().getFirstChild();
        firstName = textNode.getNodeValue();
    }
    childNode.getNextSibling();
}
```

Object Code

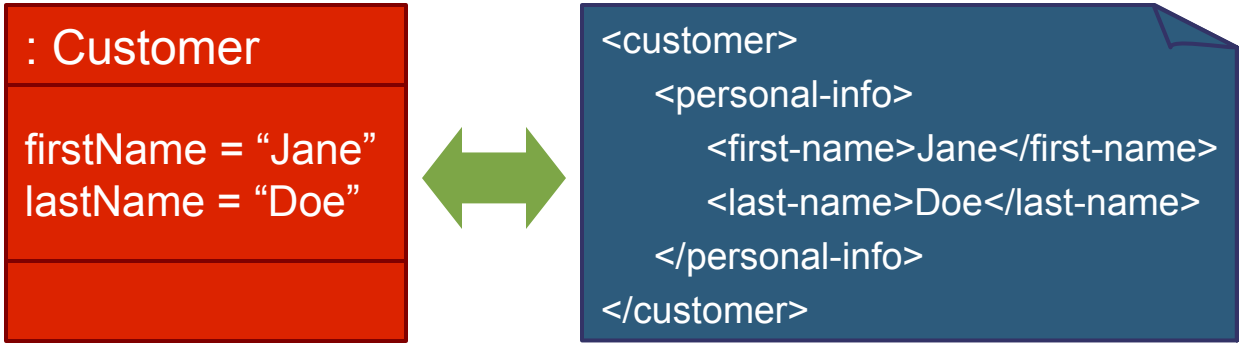
```
customer.getFirstName();
```

The Object Model

Generated Object Model

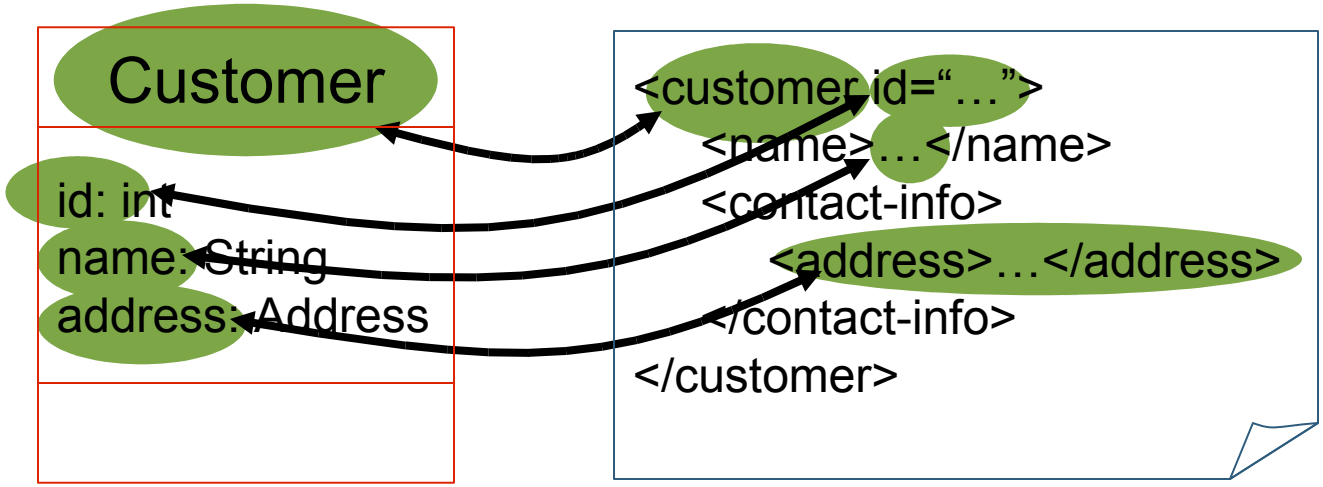


Your Own Domain Objects



Mapping

- The activity of ‘Mapping’ is the process of connecting objects/attributes to XML types/nodes



Internal Mapping Metadata

Mapping Information Using Java API for XML Binding 2.0 (JAXB) Annotations

```
@XmlRootElement
public class Customer {

    @XmlAttribute(name="id")
    public int getId() {…}
    public void setId(int id) {…}

    @XmlElement(name="billing-address")
    public Address getBillingAddress() {…}
    public void setBillingAddress(Address address) {…}

}
```

External Mapping Metadata

Mapping Information Using an XML File

- Mapping information is kept in XML descriptors and not in the objects
- External metadata means this approach is NOT at all intrusive on either the object model or the XML schema
- The object model can be mapped to multiple XML representations

Agenda

Usability

Performance

Full W3C XML Schema Support

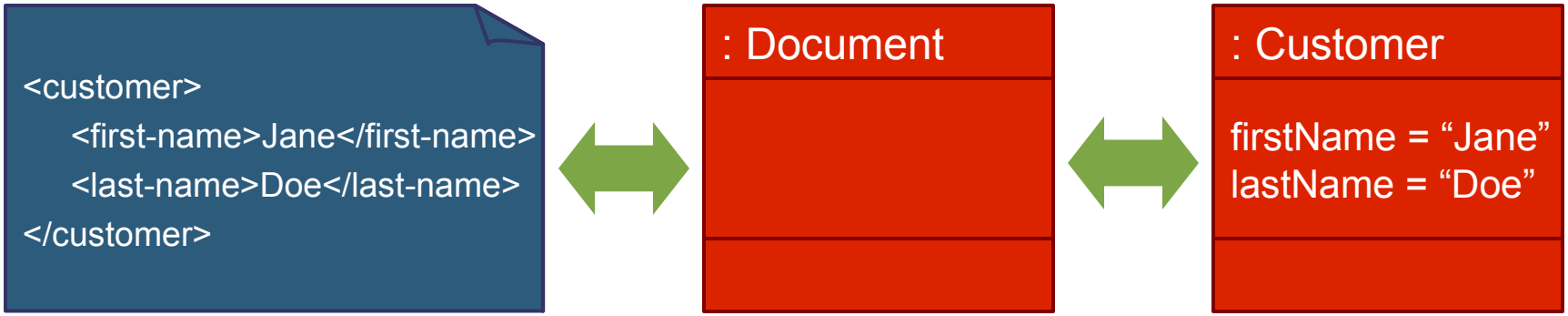
Standards Compliance

Compatibility With Other Standards

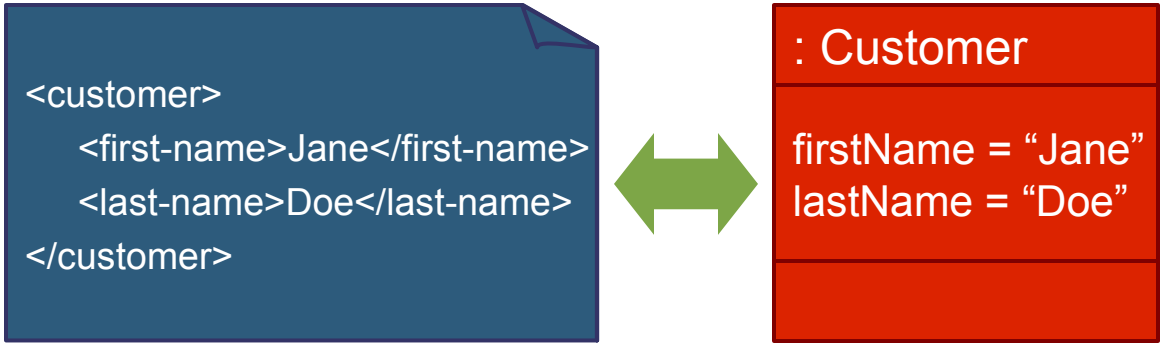
Compatibility With your SOA

The Object Model

DOM Based—Requires an Intermediate Structure



Event Based—No Intermediate Structure Required



DOM Based Binding Solutions

Advantages

- Unmapped XML content can be preserved (such as comments)
- User can be given access to the underlying “DOM” structure

Disadvantages

- Slower and requires more memory
  - Underlying “DOM” structure must be built and traversed

Event Based Binding Solutions

Advantages

- Better performance since an intermediate structure need not be built

Disadvantages

- Unmapped XML content cannot be preserved (such as comments)
- User can be given access to the underlying “DOM” structure

Agenda

Usability

Performance

Full W3C XML Schema Support

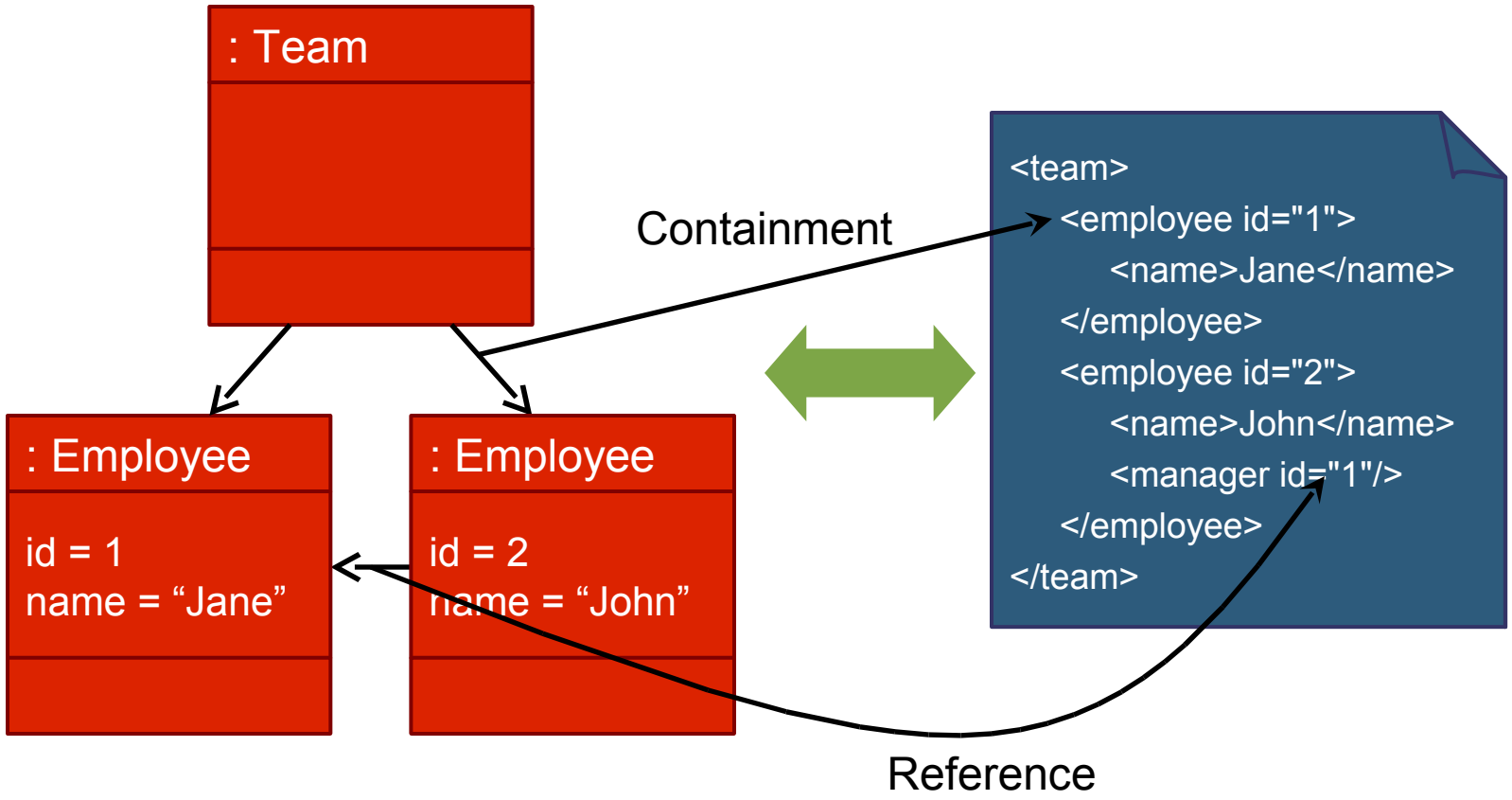
Standards Compliance

Compatibility With Other Standards

Compatibility With your SOA

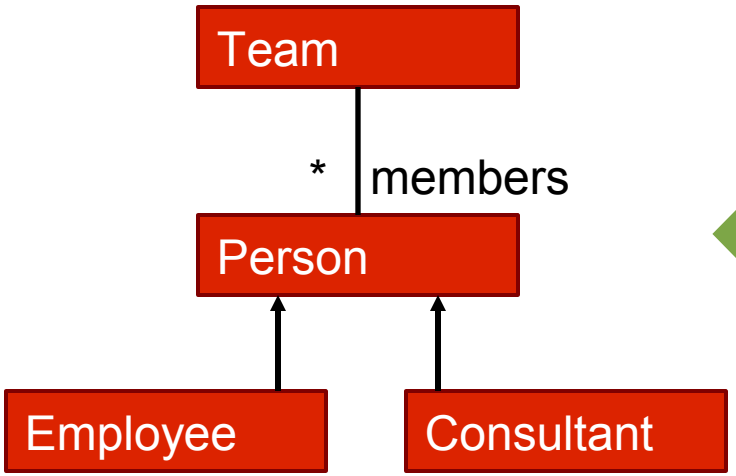
Relationships

Containment and Reference (Key-Based)



Substitution Groups

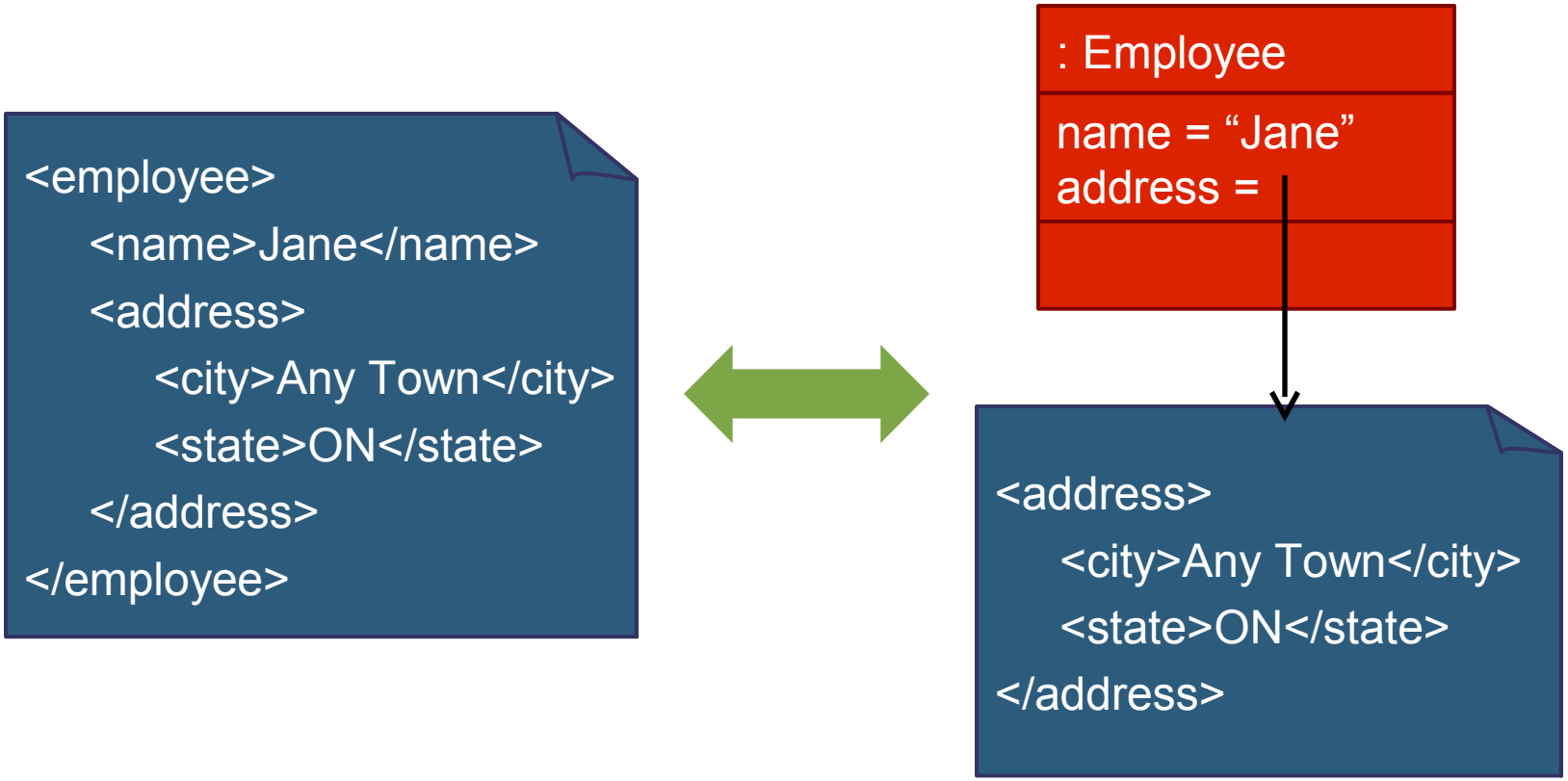
A Mechanism for Allowing Element Replacement



```
<team>
    <members>
        <employee>...</employee>
        <employee>...</employee>
        <consultant>...</consultant>
        <employee>...</employee>
    </members>
</team>
```

Partial XML Mapping

When All Else Fails, Leave it as XML



Agenda

Usability

Performance

Full W3C XML Schema Support

Standards Compliance

Compatibility With Other Standards

Compatibility With your SOA

Standard Runtime

JAXB 1.0 Standardized the Object-to-XML Runtime

```
// Instantiate the JAXB context.  The context path
// indicates which classes are involved in the XML binding
JAXBContext context =
    JAXBContext.newInstance(CONTEXT_PATH);

// Unmarshal the objects from XML
File file = new File("input.xml");
Unmarshaller unmarshaller = context.createUnmarshaller();
Customer customer = (Customer)
    unmarshaller.unmarshal(file);

// Marshal the objects to XML
Marshaller marshaller = context.createMarshaller();
marshaller.marshal(customer, System.out);
```

Standard Objects

JAXB 2.0 Standardizes the Objects

- No binding logic in the generated classes
- Metadata specified using Java annotations
- The only compile time dependencies are standard JAXB classes and interfaces
- Classes generated by one vendors compiler can be used in another vendors runtime
- JAXB 2.0 is part of Java EE 5
- JAXB 2.0 compiler to be included in Java SE 6 (Mustang)

Agenda

Usability

Performance

Full W3C XML Schema Support

Standards Compliance

Compatibility With Other Standards

Compatibility With your SOA

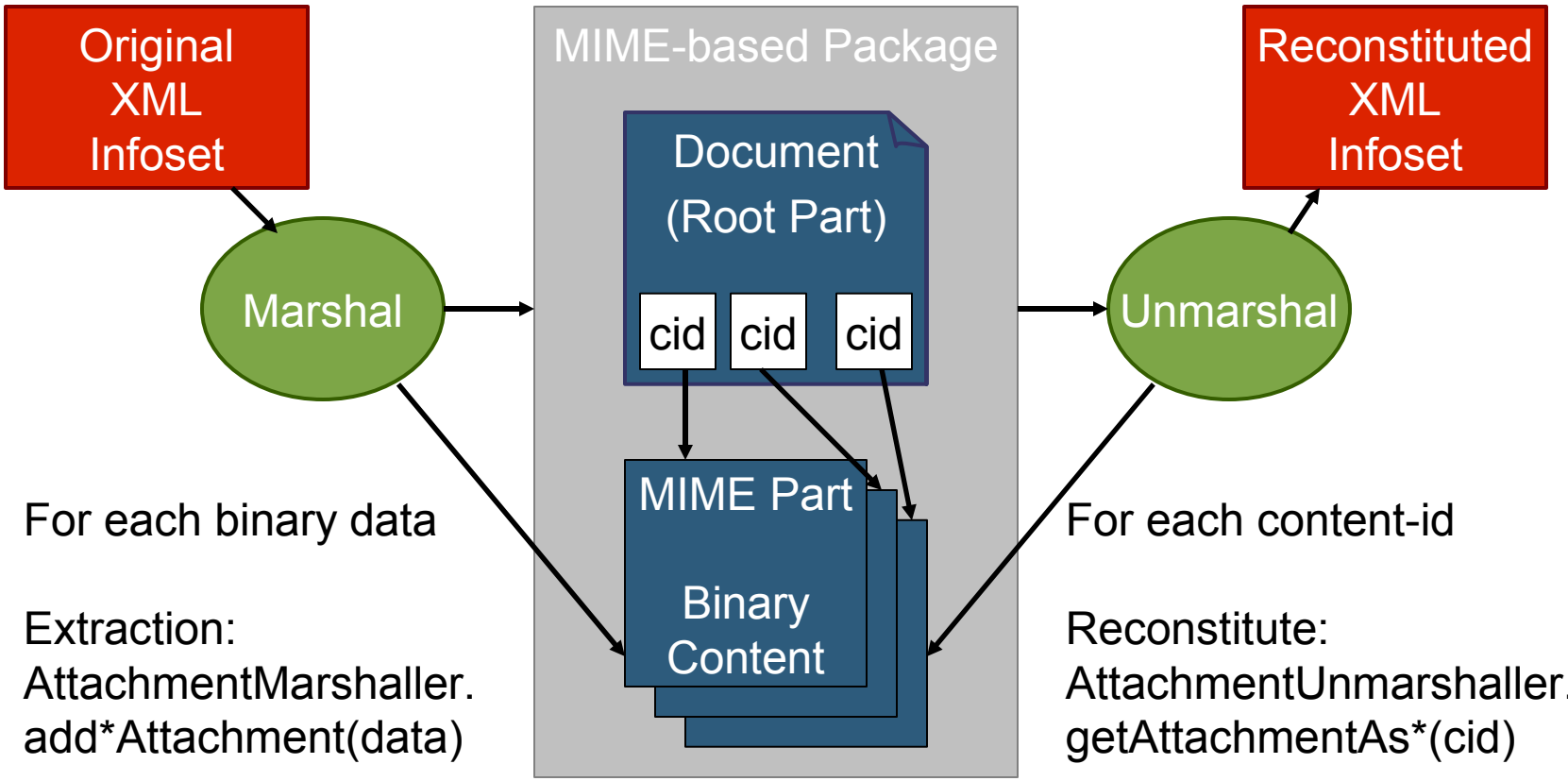
JAX-WS 2.0

Java API for XML-Based Web Services

- Successor to JAX-RPC 1.1
- Part of Java EE 5
- Uses JAXB 2.0 as the XML Binding Layer
- JAXB 2.0 compiler to be included in Java SE 6 (Mustang)
- Binding Support for WS Attachments
  - SOAP (MTOM)
  - XML-binary Optimized Packaging (XOP)
  - WS-I Attachment Profile 1.0 (WSIAP)

Binary Attachments

JAXB Marshal/Unmarshalling of Binary Attachments



Source: JAXB 2.0 specification, page 363.

JAXB 2.0 and EJB™ 3.0 Specification

Combining JAXB 2.0 and EJB 3.0 Specification Annotations

```
@XmlRootElement
@Entity
public class Customer {

    @XmlAttribute(name="id")
    @Id
    public int getId() {…}
    public void setId(int id) {…}

    @XmlElement(name="billing-address")
    @OneToOne
    @JoinColumn(name="ADDR_ID")
    public Address getBillingAddress() {…}
    public void setBillingAddress(Address address) {…}

}
```

Agenda

Usability

Performance

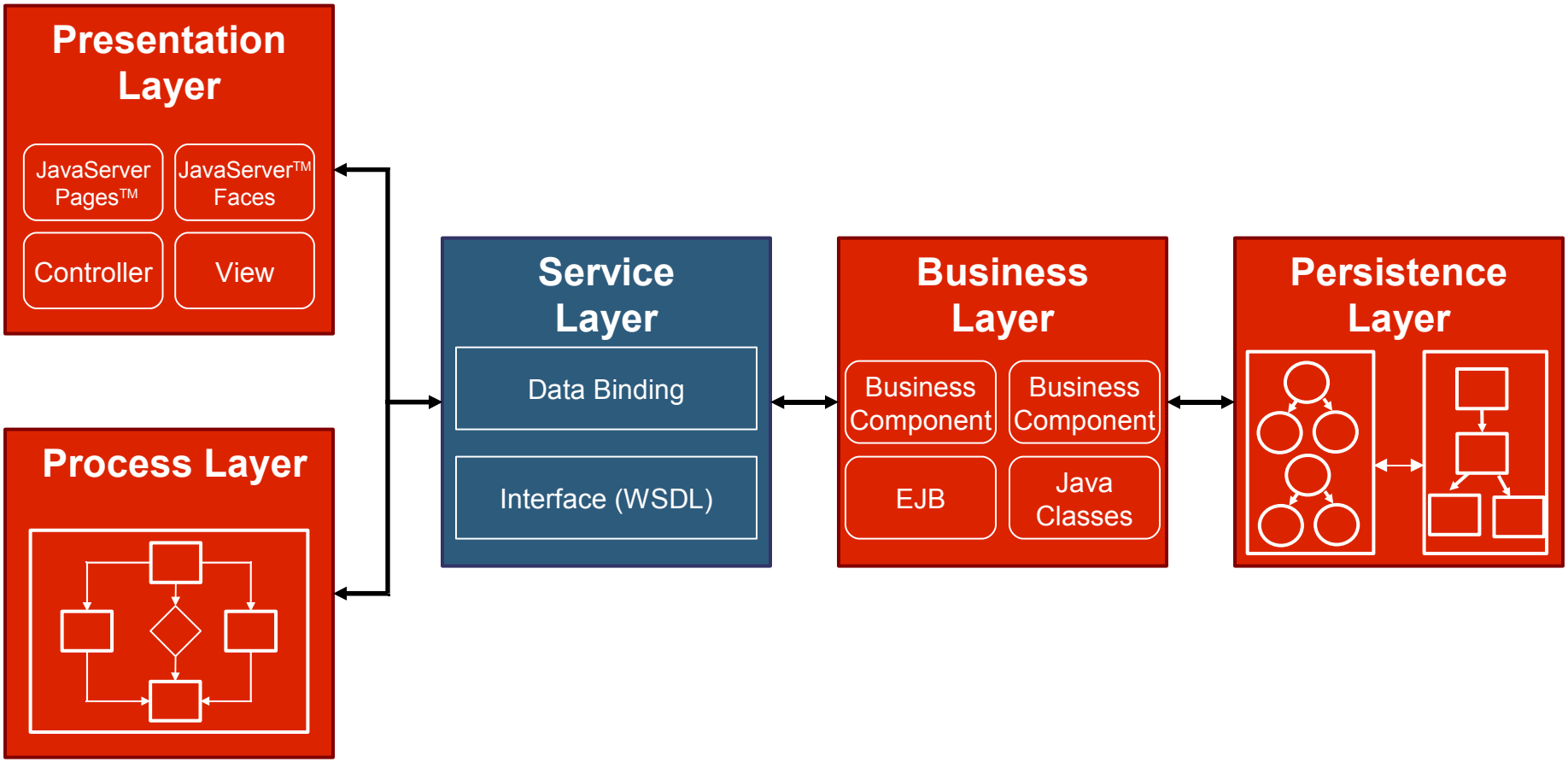
Full W3C XML Schema Support

Standards Compliance

Compatibility With Other Standards

Compatibility With your SOA

Service Oriented Architectures



Parser Independence

Write Once, Run Anywhere

- Most XML binding layers are bound to a specific version of an XML parser
- Most enterprise applications are run on application servers
- Each application includes and depends on a specific version of an XML parser
- If the binding layer and application server are dependent on different XML parsers, then it may be impossible to use them together

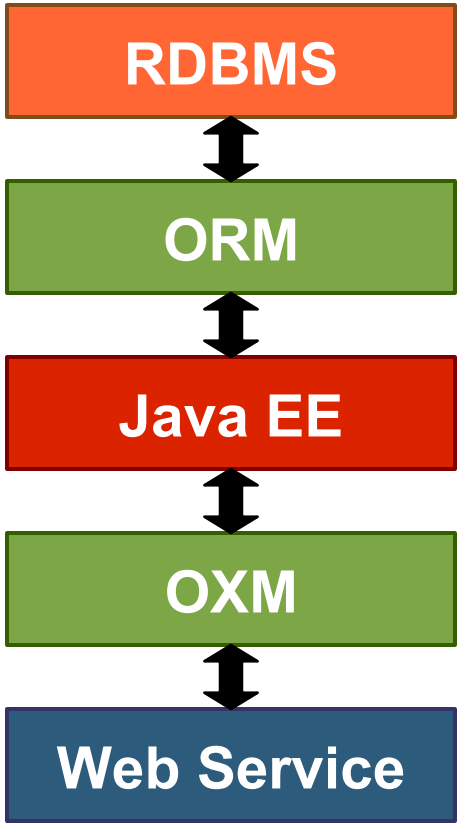
Customer Example #1

One Object Model Mapped to RDBMS and XML

Legacy database, used by many applications

Application to process insurance quotes

XML message based on an industry standard



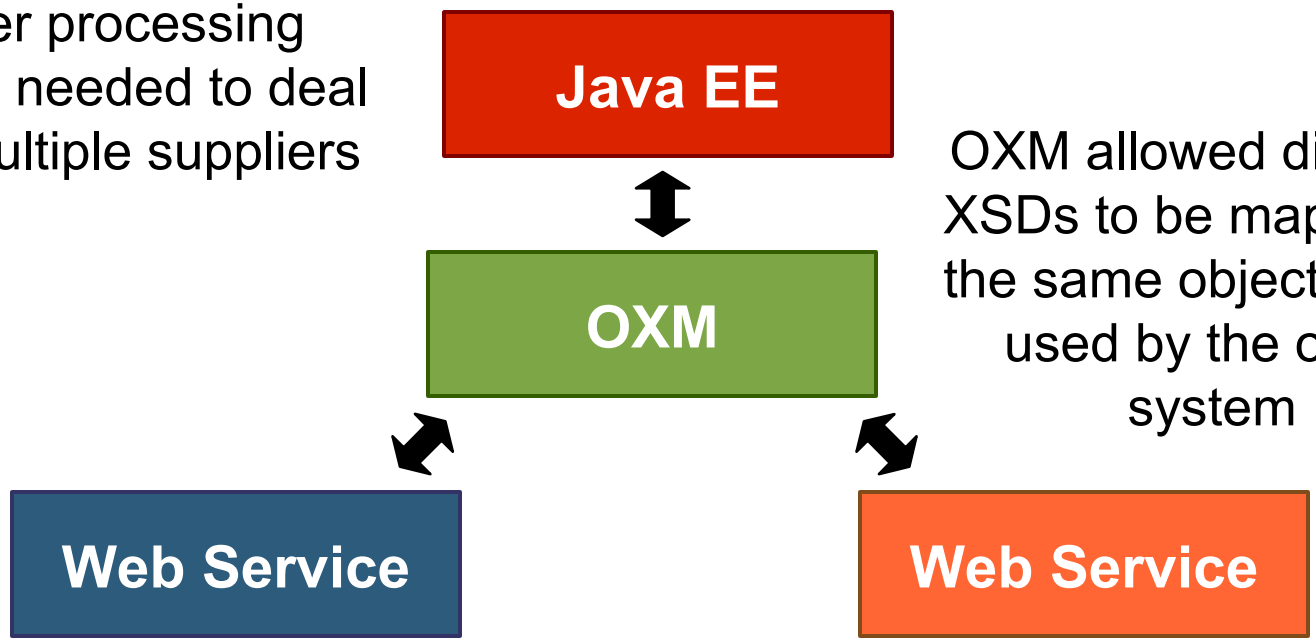
Object-to-relational mapping allows app. to access relational data as objects

Object-to-XML mapping allows the application objects to be exposed as XML

Customer Example #2

One Object Model Mapped to Multiple XML Schemas

Order processing system needed to deal with multiple suppliers



OXM allowed different XSDs to be mapped to the same object model used by the order system

Different suppliers defined proprietary XML schemas

DEMO

Blaise Doughan and Shaun Smith

Summary

- Usability
- Performance
- Full W3C XML Schema support
- Standards compliance
- Compatibility with other standards
- Compatibility with your SOA

For More Information

- TopLink Product Page—Contains links to
  - Introduction to TopLink OXM Article
  - TopLink Download
  - How-to's with step by step instructions to map the Customer Project or run the JAXB compiler
  - TopLink examples including a fully mapped version of the Customer project
- otn.oracle.com/products/ias/toplink/oxm

Q&A

Blaise Doughan and Shaun Smith

Six Things I Want in an XML Binding Layer

Blaise Doughan

Team Lead, TopLink OXM
Oracle Corporation
www.oracle.com

Shaun Smith

Product Manager, TopLink
Oracle Corporation
www.oracle.com

Session TS-6224