Using Advanced Features of JAXB

Jesper de Jong
SOFTWARE ARCHITECT

@jesperdj www.jesperdj.com

Overview

Validation

- During marshalling and unmarshalling
- Validating objects in memory

Inheritance in XML Schema and Java

Open Schemas and @XmlAnyElement

Class Binder

- Using EclipseLink MOXy

Enabling Validation During Marshalling and Unmarshalling

Validating Data

Why validate incoming data?

- The system must be able to handle data errors
- Guard integrity of stored data

```
ps-jaxb-23-validation-1
```

Validating Objects in Memory

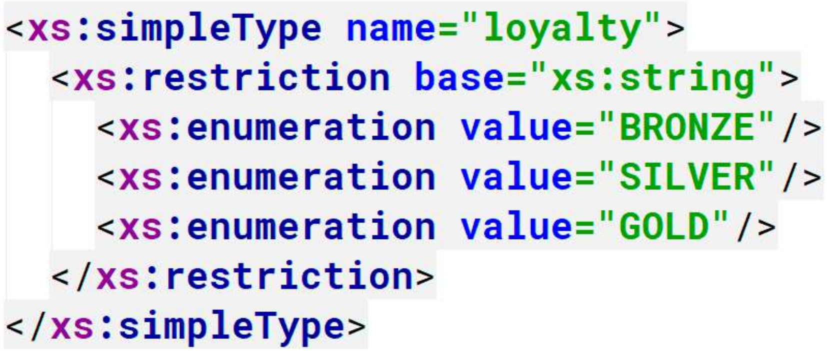
Validating Objects in Memory

ps-jaxb-24-validation-2

Using Inheritance in XML Schema and Java

Inheritance in XML Schema and Java

XML Schema: Deriving types by restriction



```
<xs:simpleType name="loyalty">
  <xs:restriction base="xs:string">
    <xs:enumeration value="BRONZE"/>
    <xs:enumeration value="SILVER"/>
    <xs:enumeration value="GOLD"/>
  </xs:restriction>
</xs:simpleType>
```

Inheritance in XML Schema and Java

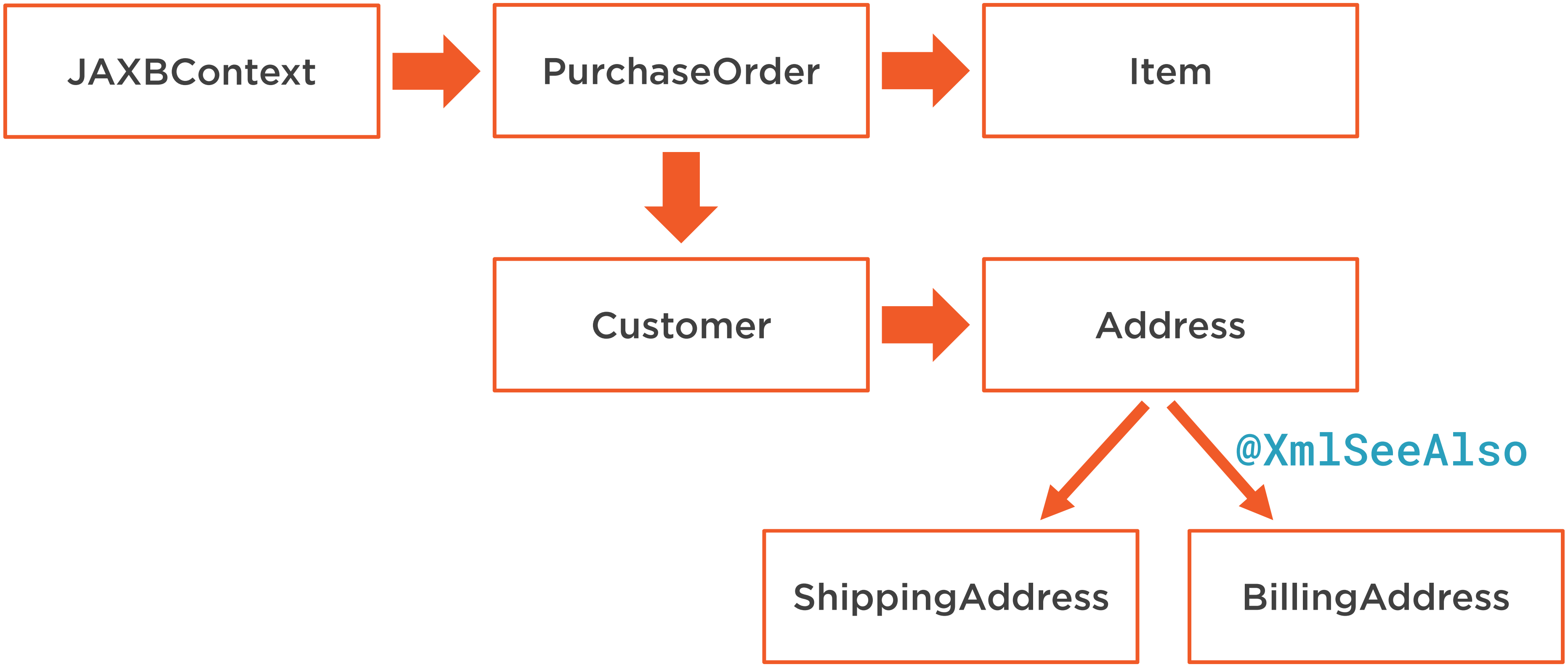
XML Schema: Deriving types by extension

↕

Java: Superclasses and subclasses

```
ps-jaxb-25-inheritance
```

JAXBContext: Finding Classes

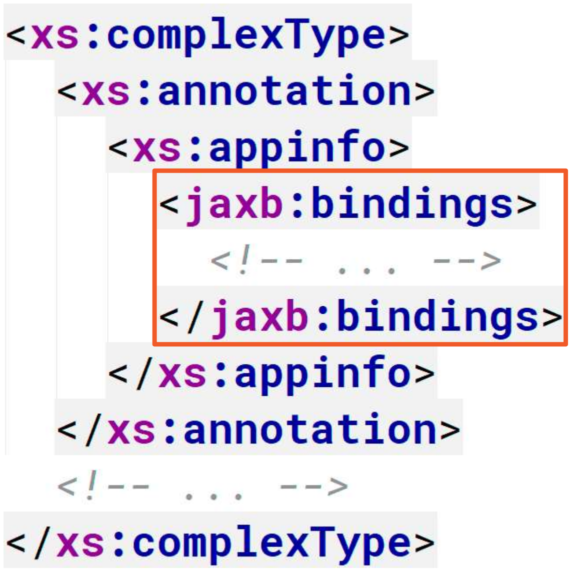


Mapping Extra Elements Using XmlAnyElement

Mapping Extra Elements

```
<xs:any namespace="##other" processContents="lax"
        minOccurs="0" maxOccurs="unbounded"/>
```

Mapping Extra Elements

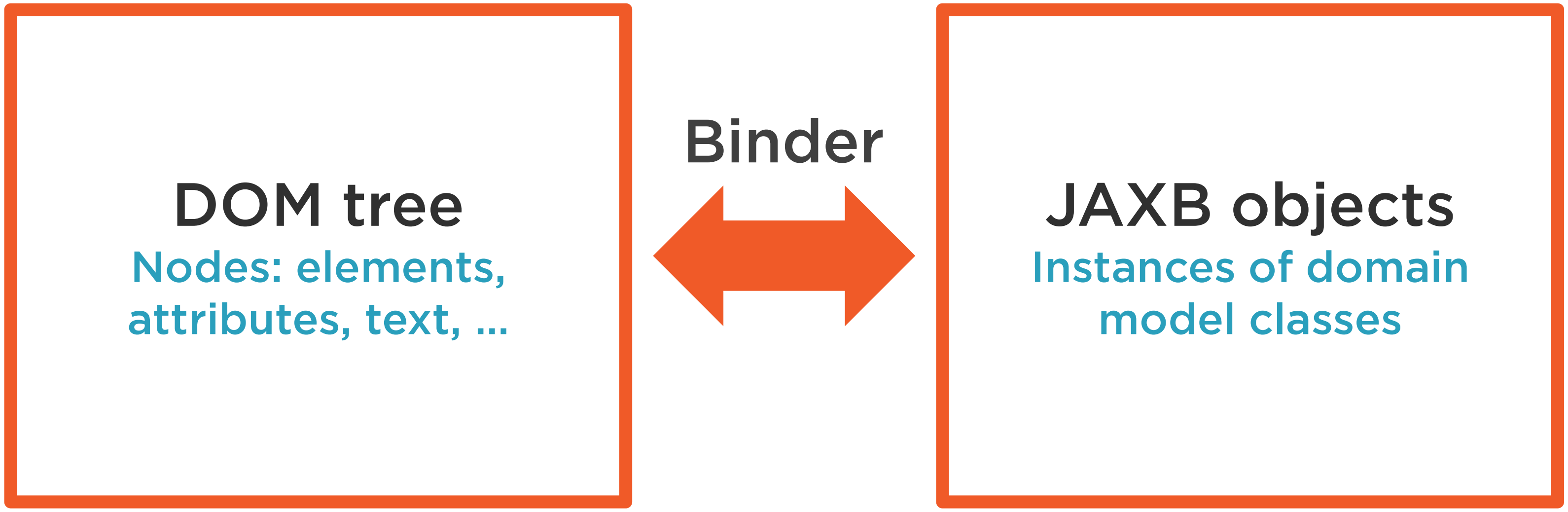


```
<xs:complexType>
    <xs:annotation>
        <xs:appinfo>
            <jaxb:bindings>
                <!-- ... -->
            </jaxb:bindings>
        </xs:appinfo>
    </xs:annotation>
    <!-- ... -->
</xs:complexType>
```

ps-jaxb-26-xmlany-1

Unmarshalling and Updating Using a Binder

Class Binder



DOM tree

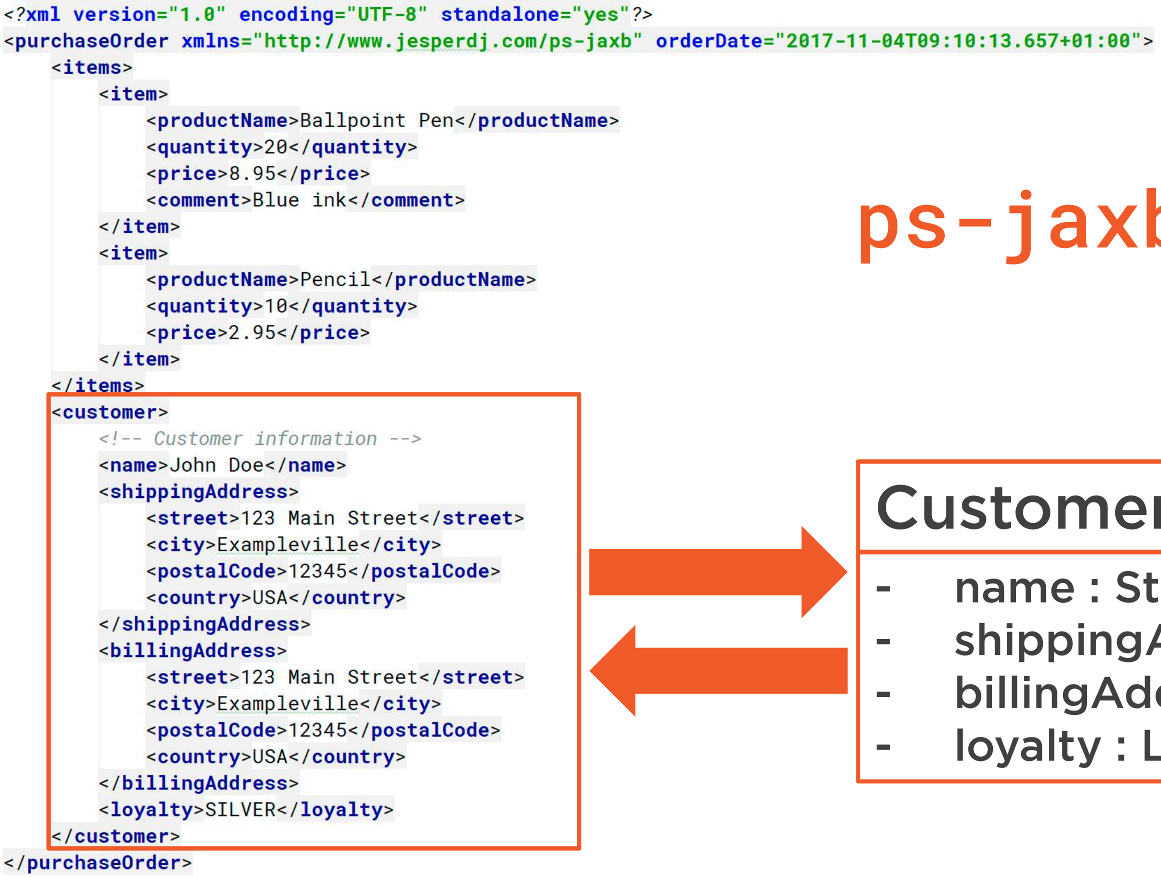
Nodes: elements, attributes, text, …

Binder

JAXB objects

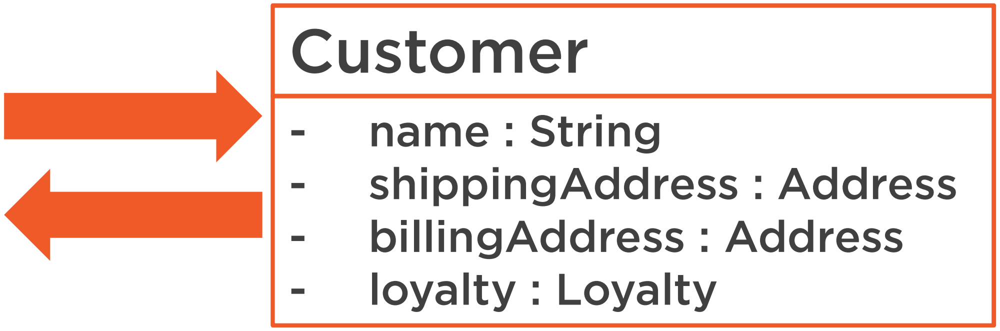
Instances of domain model classes

Class Binder



```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<purchaseOrder xmlns="http://www.jesperdj.com/ps-jaxb" orderDate="2017-11-04T09:10:13.657+01:00">
    <items>
        <item>
            <productName>Ballpoint Pen</productName>
            <quantity>20</quantity>
            <price>8.95</price>
            <comment>Blue ink</comment>
        </item>
        <item>
            <productName>Pencil</productName>
            <quantity>10</quantity>
            <price>2.95</price>
        </item>
    </items>
    <customer>
        <!-- Customer information -->
        <name>John Doe</name>
        <shippingAddress>
            <street>123 Main Street</street>
            <city>Exampleville</city>
            <postalCode>12345</postalCode>
            <country>USA</country>
        </shippingAddress>
        <billingAddress>
            <street>123 Main Street</street>
            <city>Exampleville</city>
            <postalCode>12345</postalCode>
            <country>USA</country>
        </billingAddress>
        <loyalty>SILVER</loyalty>
    </customer>
</purchaseOrder>
```

ps-jaxb-28-binder



Customer

- name : String
- shippingAddress : Address
- billingAddress : Address
- loyalty : Loyalty

EclipseLink MOXy

Alternative implementation of JAXB

http://www.eclipse.org/eclipselink/#moxy

Summary

Validation

- During marshalling and unmarshalling
- ValidationEventHandler
- In memory using JAXBSource

Inheritance

- Mapping between XML Schema and Java
- @XmlSeeAlso

Using @XmlAnyElement

Class Binder

- Map DOM tree to JAXB, maintaining link
- EclipseLink MOXy