| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/XmlIDREF.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/xml/bind/annotation/XmlID.html)   [**NEXT CLASS**](http://docs.google.com/javax/xml/bind/annotation/XmlInlineBinaryData.html) | [**FRAMES**](http://docs.google.com/index.html?javax/xml/bind/annotation/XmlIDREF.html)    [**NO FRAMES**](http://docs.google.com/XmlIDREF.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: REQUIRED | OPTIONAL | DETAIL: ELEMENT |

## **javax.xml.bind.annotation**

Annotation Type XmlIDREF

[@Retention](http://docs.google.com/java/lang/annotation/Retention.html)([value](http://docs.google.com/java/lang/annotation/Retention.html#value())=[RUNTIME](http://docs.google.com/java/lang/annotation/RetentionPolicy.html#RUNTIME))  
[@Target](http://docs.google.com/java/lang/annotation/Target.html)([value](http://docs.google.com/java/lang/annotation/Target.html#value())={[FIELD](http://docs.google.com/java/lang/annotation/ElementType.html#FIELD),[METHOD](http://docs.google.com/java/lang/annotation/ElementType.html#METHOD)})  
public @interface **XmlIDREF**

Maps a JavaBean property to XML IDREF.

To preserve referential integrity of an object graph across XML serialization followed by a XML deserialization, requires an object reference to be marshalled by reference or containment appropriately. Annotations @XmlID and @XmlIDREF together allow a customized mapping of a JavaBean property's type by containment or reference.

**Usage**

The @XmlIDREF annotation can be used with the following program elements:

* a JavaBean property
* non static, non transient field

See "Package Specification" in javax.xml.bind.package javadoc for additional common information.

The usage is subject to the following constraints:

* If the type of the field or property is a collection type, then the collection item type must contain a property or field annotated with @XmlID.
* If the field or property is single valued, then the type of the property or field must contain a property or field annotated with @XmlID.  
  Note: If the collection item type or the type of the property (for non collection type) is java.lang.Object, then the instance must contain a property/field annotated with @XmlID attribute.
* This annotation can be used with the following annotations: [XmlElement](http://docs.google.com/javax/xml/bind/annotation/XmlElement.html), [XmlAttribute](http://docs.google.com/javax/xml/bind/annotation/XmlAttribute.html), [XmlList](http://docs.google.com/javax/xml/bind/annotation/XmlList.html), and [XmlElements](http://docs.google.com/javax/xml/bind/annotation/XmlElements.html).

**Example:** Map a JavaBean property to xs:IDREF (i.e. by reference rather than by containment)

//EXAMPLE: Code fragment  
 public class Shipping {  
 @XmlIDREF public Customer getCustomer();  
 public void setCustomer(Customer customer);  
 ....  
 }  
  
 <!-- Example: XML Schema fragment -->  
 <xs:complexType name="Shipping">  
 <xs:complexContent>  
 <xs:sequence>  
 <xs:element name="customer" type="xs:IDREF"/>  
 ....  
 </xs:sequence>  
 </xs:complexContent>  
 </xs:complexType>

**Example 2:**  The following is a complete example of containment versus reference.

// By default, Customer maps to complex type xs:Customer  
 public class Customer {  
   
 // map JavaBean property type to xs:ID  
 @XmlID public String getCustomerID();  
 public void setCustomerID(String id);  
  
 // .... other properties not shown   
 }  
  
  
 // By default, Invoice maps to a complex type xs:Invoice   
 public class Invoice {  
   
 // map by reference  
 @XmlIDREF public Customer getCustomer();   
 public void setCustomer(Customer customer);  
  
 // .... other properties not shown here  
 }  
  
 // By default, Shipping maps to complex type xs:Shipping  
 public class Shipping {  
  
 // map by reference  
 @XmlIDREF public Customer getCustomer();   
 public void setCustomer(Customer customer);  
 }  
  
 // at least one class must reference Customer by containment;  
 // Customer instances won't be marshalled.  
 @XmlElement(name="CustomerData")  
 public class CustomerData {  
 // map reference to Customer by containment by default.  
 public Customer getCustomer();  
  
 // maps reference to Shipping by containment by default.   
 public Shipping getShipping();   
  
 // maps reference to Invoice by containment by default.   
 public Invoice getInvoice();   
 }  
  
 <!-- XML Schema mapping for above code frament -->  
  
 <xs:complexType name="Invoice">  
 <xs:complexContent>  
 <xs:sequence>  
 <xs:element name="customer" type="xs:IDREF"/>  
 ....  
 </xs:sequence>  
 </xs:complexContent>  
 </xs:complexType>  
  
 <xs:complexType name="Shipping">  
 <xs:complexContent>  
 <xs:sequence>  
 <xs:element name="customer" type="xs:IDREF"/>  
 ....  
 </xs:sequence>  
 </xs:complexContent>  
 </xs:complexType>  
  
 <xs:complexType name="Customer">  
 <xs:complexContent>  
 <xs:sequence>  
 ....  
 </xs:sequence>  
 <xs:attribute name="CustomerID" type="xs:ID"/>  
 </xs:complexContent>  
 </xs:complexType>  
  
 <xs:complexType name="CustomerData">  
 <xs:complexContent>  
 <xs:sequence>  
 <xs:element name="customer" type="xs:Customer"/>  
 <xs:element name="shipping" type="xs:Shipping"/>  
 <xs:element name="invoice" type="xs:Invoice"/>  
 </xs:sequence>  
 </xs:complexContent>  
 </xs:complexType>  
  
 <xs:element name"customerData" type="xs:CustomerData"/>  
  
 <!-- Instance document conforming to the above XML Schema -->  
 <customerData>  
 <customer customerID="Alice">  
 ....  
 </customer>  
  
 <shipping customer="Alice">  
 ....  
 </shipping>  
   
 <invoice customer="Alice">  
 ....  
 </invoice>  
 </customerData>

**Example 3:**  Mapping List to repeating element of type IDREF

// Code fragment  
 public class Shipping {  
 @XmlIDREF  
 @XmlElement(name="Alice")  
 public List customers;  
 }  
  
 <!-- XML schema fragment -->  
 <xs:complexType name="Shipping">  
 <xs:sequence>  
 <xs:choice minOccurs="0" maxOccurs="unbounded">  
 <xs:element name="Alice" type="xs:IDREF"/>  
 </xs:choice>  
 </xs:sequence>  
 </xs:complexType>

**Example 4:**  Mapping a List to a list of elements of type IDREF.

//Code fragment  
 public class Shipping {  
 @XmlIDREF  
 @XmlElements(  
 @XmlElement(name="Alice", type="Customer.class")  
 @XmlElement(name="John", type="InternationalCustomer.class")  
 public List customers;  
 }  
  
 <!-- XML Schema fragment -->  
 <xs:complexType name="Shipping">  
 <xs:sequence>  
 <xs:choice minOccurs="0" maxOccurs="unbounded">  
 <xs:element name="Alice" type="xs:IDREF"/>  
 <xs:element name="John" type="xs:IDREF"/>  
 </xs:choice>  
 </xs:sequence>  
 </xs:complexType>

**Since:** JAXB2.0 **See Also:**[XmlID](http://docs.google.com/javax/xml/bind/annotation/XmlID.html)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/XmlIDREF.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/xml/bind/annotation/XmlID.html)   [**NEXT CLASS**](http://docs.google.com/javax/xml/bind/annotation/XmlInlineBinaryData.html) | [**FRAMES**](http://docs.google.com/index.html?javax/xml/bind/annotation/XmlIDREF.html)    [**NO FRAMES**](http://docs.google.com/XmlIDREF.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: REQUIRED | OPTIONAL | DETAIL: ELEMENT |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).