| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/XmlElementDecl.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/XmlElement.DEFAULT.html)   [**NEXT CLASS**](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.GLOBAL.html) | [**FRAMES**](http://docs.google.com/index.html?javax/xml/bind/annotation/XmlElementDecl.html)    [**NO FRAMES**](http://docs.google.com/XmlElementDecl.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [REQUIRED](#3znysh7) | [OPTIONAL](#2et92p0) | DETAIL: [ELEMENT](#tyjcwt) |

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

Annotation Type XmlElementDecl

[@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())=[METHOD](http://docs.google.com/java/lang/annotation/ElementType.html#METHOD))  
public @interface **XmlElementDecl**

Maps a factory method to a XML element.

**Usage**

The annotation creates a mapping between an XML schema element declaration and a  *element factory method*  that returns a JAXBElement instance representing the element declaration. Typically, the element factory method is generated (and annotated) from a schema into the ObjectFactory class in a Java package that represents the binding of the element declaration's target namespace. Thus, while the annotation syntax allows @XmlElementDecl to be used on any method, semantically its use is restricted to annotation of element factory method. The usage is subject to the following constraints:

* The class containing the element factory method annotated with @XmlElementDecl must be marked with [XmlRegistry](http://docs.google.com/javax/xml/bind/annotation/XmlRegistry.html).
* The element factory method must take one parameter assignable to [Object](http://docs.google.com/java/lang/Object.html).

**Example 1:** Annotation on a factory method

// Example: code fragment  
 @XmlRegistry  
 class ObjectFactory {  
 @XmlElementDecl(name="foo")  
 JAXBElement<String> createFoo(String s) { ... }  
 }

<foo>string</foo>  
  
 // Example: code fragment corresponding to XML input  
 JAXBElement<String> o =  
 (JAXBElement<String>)unmarshaller.unmarshal(aboveDocument);  
 // print JAXBElement instance to show values  
 System.out.println(o.getName()); // prints "{}foo"  
 System.out.println(o.getValue()); // prints "string"  
 System.out.println(o.getValue().getClass()); // prints "java.lang.String"  
  
   
 <xs:element name="foo" type="xs:string"></xs:element>

**Example 2:**  Element declaration with non local scope

The following example illustrates the use of scope annotation parameter in binding of element declaration in schema derived code.

The following example may be replaced in a future revision of this javadoc.

<xs:schema>  
 <xs:complexType name="pea">  
 <xs:choice maxOccurs="unbounded">  
 <xs:element name="foo" type="xs:string"></xs:element>  
 <xs:element name="bar" type="xs:string"></xs:element>  
 </xs:choice>  
 </xs:complexType>   
 <xs:element name="foo" type="xs:int"></xs:element>  
 </xs:schema>

// Example: expected default binding  
 class Pea {  
 @XmlElementRefs({  
 @XmlElementRef(name="foo",type=JAXBElement.class)  
 @XmlElementRef(name="bar",type=JAXBElement.class)  
 })  
 List<JAXBElement<String>> fooOrBar;  
 }  
   
 @XmlRegistry  
 class ObjectFactory {  
 @XmlElementDecl(scope=Pea.class,name="foo")  
 JAXBElement createPeaFoo(String s);  
   
 @XmlElementDecl(scope=Pea.class,name="bar")  
 JAXBElement createPeaBar(String s);  
   
 @XmlElementDecl(name="foo")  
 JAXBElement createFoo(Integer i);  
 }

Without scope createFoo and createPeaFoo would become ambiguous since both of them map to a XML schema element with the same local name "foo".

**Since:** JAXB 2.0 **See Also:**[XmlRegistry](http://docs.google.com/javax/xml/bind/annotation/XmlRegistry.html)

| **Required Element Summary** | |
| --- | --- |
| [String](http://docs.google.com/java/lang/String.html) | [**name**](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.html#name())            local name of the XML element. |

| **Optional Element Summary** | |
| --- | --- |
| [String](http://docs.google.com/java/lang/String.html) | [**defaultValue**](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.html#defaultValue())            Default value of this element. |
| [String](http://docs.google.com/java/lang/String.html) | [**namespace**](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.html#namespace())            namespace name of the XML element. |
| [Class](http://docs.google.com/java/lang/Class.html) | [**scope**](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.html#scope())            scope of the mapping. |
| [String](http://docs.google.com/java/lang/String.html) | [**substitutionHeadName**](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.html#substitutionHeadName())            XML local name of a substitution group's head element. |
| [String](http://docs.google.com/java/lang/String.html) | [**substitutionHeadNamespace**](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.html#substitutionHeadNamespace())            namespace name of a substitution group's head XML element. |

| **Element Detail** |
| --- |

### name

public abstract [String](http://docs.google.com/java/lang/String.html) **name**

local name of the XML element.

**Note to reviewers:**  There is no default name; since the annotation is on a factory method, it is not clear that the method name can be derived from the factory method name.

**See Also:**[namespace()](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.html#namespace())

### scope

public abstract [Class](http://docs.google.com/java/lang/Class.html) **scope**

scope of the mapping.

If this is not [XmlElementDecl.GLOBAL](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.GLOBAL.html), then this element declaration mapping is only active within the specified class.

**Default:**javax.xml.bind.annotation.XmlElementDecl.GLOBAL.class

### namespace

public abstract [String](http://docs.google.com/java/lang/String.html) **namespace**

namespace name of the XML element.

If the value is "##default", then the value is the namespace name for the package of the class containing this factory method.

**See Also:**[name()](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.html#name()) **Default:**"##default"

### substitutionHeadNamespace

public abstract [String](http://docs.google.com/java/lang/String.html) **substitutionHeadNamespace**

namespace name of a substitution group's head XML element.

This specifies the namespace name of the XML element whose local name is specified by substitutionHeadName().

If susbtitutionHeadName() is "", then this value can only be "##default". But the value is ignored since since this element is not part of susbtitution group when the value of susbstitutionHeadName() is "".

If susbtitutionHeadName() is not "" and the value is "##default", then the namespace name is the namespace name to which the package of the containing class, marked with [XmlRegistry](http://docs.google.com/javax/xml/bind/annotation/XmlRegistry.html), is mapped.

If susbtitutionHeadName() is not "" and the value is not "##default", then the value is the namespace name.

**See Also:**[substitutionHeadName()](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.html#substitutionHeadName()) **Default:**"##default"

### substitutionHeadName

public abstract [String](http://docs.google.com/java/lang/String.html) **substitutionHeadName**

XML local name of a substitution group's head element.

If the value is "", then this element is not part of any substitution group.

**See Also:**[substitutionHeadNamespace()](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.html#substitutionHeadNamespace()) **Default:**""

### defaultValue

public abstract [String](http://docs.google.com/java/lang/String.html) **defaultValue**

Default value of this element.

The '�' value specified as a default of this annotation element is used as a poor-man's substitute for null to allow implementations to recognize the 'no default value' state.

**Default:**"\u0000"

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/XmlElementDecl.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/XmlElement.DEFAULT.html)   [**NEXT CLASS**](http://docs.google.com/javax/xml/bind/annotation/XmlElementDecl.GLOBAL.html) | [**FRAMES**](http://docs.google.com/index.html?javax/xml/bind/annotation/XmlElementDecl.html)    [**NO FRAMES**](http://docs.google.com/XmlElementDecl.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [REQUIRED](#3znysh7) | [OPTIONAL](#2et92p0) | DETAIL: [ELEMENT](#tyjcwt) |

[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).