| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Marshaller.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/MarshalException.html)   [**NEXT CLASS**](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) | [**FRAMES**](http://docs.google.com/index.html?javax/xml/bind/Marshaller.html)    [**NO FRAMES**](http://docs.google.com/Marshaller.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3dy6vkm) | [FIELD](#1t3h5sf) | CONSTR | [METHOD](#4d34og8) | DETAIL: [FIELD](#2s8eyo1) | CONSTR | [METHOD](#1ksv4uv) |

## **javax.xml.bind**

Interface Marshaller

**All Known Implementing Classes:** [AbstractMarshallerImpl](http://docs.google.com/javax/xml/bind/helpers/AbstractMarshallerImpl.html)

public interface **Marshaller**

The Marshaller class is responsible for governing the process of serializing Java content trees back into XML data. It provides the basic marshalling methods:

*Assume the following setup code for all following code fragments:*

JAXBContext jc = JAXBContext.newInstance( "com.acme.foo" );  
 Unmarshaller u = jc.createUnmarshaller();  
 Object element = u.unmarshal( new File( "foo.xml" ) );  
 Marshaller m = jc.createMarshaller();

Marshalling to a File:

OutputStream os = new FileOutputStream( "nosferatu.xml" );  
 m.marshal( element, os );

Marshalling to a SAX ContentHandler:

// assume MyContentHandler instanceof ContentHandler  
 m.marshal( element, new MyContentHandler() );

Marshalling to a DOM Node:

DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();  
 dbf.setNamespaceAware(true);  
 DocumentBuilder db = dbf.newDocumentBuilder();  
 Document doc = db.newDocument();  
  
 m.marshal( element, doc );

Marshalling to a java.io.OutputStream:

m.marshal( element, System.out );

Marshalling to a java.io.Writer:

m.marshal( element, new PrintWriter( System.out ) );

Marshalling to a javax.xml.transform.SAXResult:

// assume MyContentHandler instanceof ContentHandler  
 SAXResult result = new SAXResult( new MyContentHandler() );  
  
 m.marshal( element, result );

Marshalling to a javax.xml.transform.DOMResult:

DOMResult result = new DOMResult();  
   
 m.marshal( element, result );

Marshalling to a javax.xml.transform.StreamResult:

StreamResult result = new StreamResult( System.out );  
   
 m.marshal( element, result );

Marshalling to a javax.xml.stream.XMLStreamWriter:

XMLStreamWriter xmlStreamWriter =   
 XMLOutputFactory.newInstance().createXMLStreamWriter( ... );  
   
 m.marshal( element, xmlStreamWriter );

Marshalling to a javax.xml.stream.XMLEventWriter:

XMLEventWriter xmlEventWriter =   
 XMLOutputFactory.newInstance().createXMLEventWriter( ... );  
   
 m.marshal( element, xmlEventWriter );

**Marshalling content tree rooted by a JAXB element**

The first parameter of the overloaded Marshaller.marshal(java.lang.Object, ...) methods must be a JAXB element as computed by [JAXBIntrospector.isElement(java.lang.Object)](http://docs.google.com/javax/xml/bind/JAXBIntrospector.html#isElement(java.lang.Object)); otherwise, a Marshaller.marshal method must throw a [MarshalException](http://docs.google.com/javax/xml/bind/MarshalException.html). There exist two mechanisms to enable marshalling an instance that is not a JAXB element. One method is to wrap the instance as a value of a [JAXBElement](http://docs.google.com/javax/xml/bind/JAXBElement.html), and pass the wrapper element as the first parameter to a Marshaller.marshal method. For java to schema binding, it is also possible to simply annotate the instance's class with @[XmlRootElement](http://docs.google.com/javax/xml/bind/annotation/XmlRootElement.html).

**Encoding**

By default, the Marshaller will use UTF-8 encoding when generating XML data to a java.io.OutputStream, or a java.io.Writer. Use the [setProperty](http://docs.google.com/javax/xml/bind/Marshaller.html#setProperty(java.lang.String,%20java.lang.Object)) API to change the output encoding used during these marshal operations. Client applications are expected to supply a valid character encoding name as defined in the [W3C XML 1.0 Recommendation](http://www.w3.org/TR/2000/REC-xml-20001006#charencoding) and supported by your  [Java Platform](http://java.sun.com/j2se/1.3/docs/api/java/lang/package-summary.html#charenc).

**Validation and Well-Formedness**

Client applications are not required to validate the Java content tree prior to calling any of the marshal API's. Furthermore, there is no requirement that the Java content tree be valid with respect to its original schema in order to marshal it back into XML data. Different JAXB Providers will support marshalling invalid Java content trees at varying levels, however all JAXB Providers must be able to marshal a valid content tree back to XML data. A JAXB Provider must throw a MarshalException when it is unable to complete the marshal operation due to invalid content. Some JAXB Providers will fully allow marshalling invalid content, others will fail on the first validation error.

Even when schema validation is not explictly enabled for the marshal operation, it is possible that certain types of validation events will be detected during the operation. Validation events will be reported to the registered event handler. If the client application has not registered an event handler prior to invoking one of the marshal API's, then events will be delivered to a default event handler which will terminate the marshal operation after encountering the first error or fatal error. Note that for JAXB 2.0 and later versions, [DefaultValidationEventHandler](http://docs.google.com/javax/xml/bind/helpers/DefaultValidationEventHandler.html) is no longer used.

**Supported Properties**

All JAXB Providers are required to support the following set of properties. Some providers may support additional properties.

jaxb.encoding - value must be a java.lang.String The output encoding to use when marshalling the XML data. The Marshaller will use "UTF-8" by default if this property is not specified. jaxb.formatted.output - value must be a java.lang.Boolean This property controls whether or not the Marshaller will format the resulting XML data with line breaks and indentation. A true value for this property indicates human readable indented xml data, while a false value indicates unformatted xml data. The Marshaller will default to false (unformatted) if this property is not specified. jaxb.schemaLocation - value must be a java.lang.String This property allows the client application to specify an xsi:schemaLocation attribute in the generated XML data. The format of the schemaLocation attribute value is discussed in an easy to understand, non-normative form in [Section 5.6 of the W3C XML Schema Part 0: Primer](http://www.w3.org/TR/xmlschema-0/#schemaLocation) and specified in  [Section 2.6 of the W3C XML Schema Part 1: Structures](http://www.w3.org/TR/xmlschema-1/#Instance_Document_Constructions). jaxb.noNamespaceSchemaLocation - value must be a java.lang.String This property allows the client application to specify an xsi:noNamespaceSchemaLocation attribute in the generated XML data. The format of the schemaLocation attribute value is discussed in an easy to understand, non-normative form in [Section 5.6 of the W3C XML Schema Part 0: Primer](http://www.w3.org/TR/xmlschema-0/#schemaLocation) and specified in  [Section 2.6 of the W3C XML Schema Part 1: Structures](http://www.w3.org/TR/xmlschema-1/#Instance_Document_Constructions). jaxb.fragment - value must be a java.lang.Boolean This property determines whether or not document level events will be generated by the Marshaller. If the property is not specified, the default is false. This property has different implications depending on which marshal api you are using - when this property is set to true:

* [marshal(Object,ContentHandler)](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20org.xml.sax.ContentHandler)) - the Marshaller won't invoke [ContentHandler.startDocument()](http://docs.google.com/org/xml/sax/ContentHandler.html#startDocument()) and [ContentHandler.endDocument()](http://docs.google.com/org/xml/sax/ContentHandler.html#endDocument()).
* [marshal(Object,Node)](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20org.w3c.dom.Node)) - the property has no effect on this API.
* [marshal(Object,OutputStream)](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20java.io.OutputStream)) - the Marshaller won't generate an xml declaration.
* [marshal(Object,Writer)](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20java.io.Writer)) - the Marshaller won't generate an xml declaration.
* [marshal(Object,Result)](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20javax.xml.transform.Result)) - depends on the kind of Result object, see semantics for Node, ContentHandler, and Stream APIs
* [marshal(Object,XMLEventWriter)](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20javax.xml.stream.XMLEventWriter)) - the Marshaller will not generate [XMLStreamConstants.START\_DOCUMENT](http://docs.google.com/javax/xml/stream/XMLStreamConstants.html#START_DOCUMENT) and [XMLStreamConstants.END\_DOCUMENT](http://docs.google.com/javax/xml/stream/XMLStreamConstants.html#END_DOCUMENT) events.
* [marshal(Object,XMLStreamWriter)](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20javax.xml.stream.XMLStreamWriter)) - the Marshaller will not generate [XMLStreamConstants.START\_DOCUMENT](http://docs.google.com/javax/xml/stream/XMLStreamConstants.html#START_DOCUMENT) and [XMLStreamConstants.END\_DOCUMENT](http://docs.google.com/javax/xml/stream/XMLStreamConstants.html#END_DOCUMENT) events.

**Marshal Event Callbacks**

"The [Marshaller](http://docs.google.com/javax/xml/bind/Marshaller.html) provides two styles of callback mechanisms that allow application specific processing during key points in the unmarshalling process. In 'class defined' event callbacks, application specific code placed in JAXB mapped classes is triggered during marshalling. 'External listeners' allow for centralized processing of marshal events in one callback method rather than by type event callbacks.

Class defined event callback methods allow any JAXB mapped class to specify its own specific callback methods by defining methods with the following method signatures:

// Invoked by Marshaller after it has created an instance of this object.  
 boolean beforeMarshal(Marshaller, Object parent);  
   
 // Invoked by Marshaller after it has marshalled all properties of this object.  
 void afterMmarshal(Marshaller, Object parent);

The class defined event callback methods should be used when the callback method requires access to non-public methods and/or fields of the class.

The external listener callback mechanism enables the registration of a [Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) instance with a [setListener(Listener)](http://docs.google.com/javax/xml/bind/Marshaller.html#setListener(javax.xml.bind.Marshaller.Listener)). The external listener receives all callback events, allowing for more centralized processing than per class defined callback methods.

The 'class defined' and external listener event callback methods are independent of each other, both can be called for one event. The invocation ordering when both listener callback methods exist is defined in [Marshaller.Listener.beforeMarshal(Object)](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html#beforeMarshal(java.lang.Object)) and [Marshaller.Listener.afterMarshal(Object)](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html#afterMarshal(java.lang.Object)).

An event callback method throwing an exception terminates the current marshal process.

**Since:** JAXB1.0 **See Also:**[JAXBContext](http://docs.google.com/javax/xml/bind/JAXBContext.html), [Validator](http://docs.google.com/javax/xml/bind/Validator.html), [Unmarshaller](http://docs.google.com/javax/xml/bind/Unmarshaller.html)

| **Nested Class Summary** | |
| --- | --- |
| static class | [**Marshaller.Listener**](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html)    Register an instance of an implementation of this class with a [Marshaller](http://docs.google.com/javax/xml/bind/Marshaller.html) to externally listen for marshal events. |

| **Field Summary** | |
| --- | --- |
| static [String](http://docs.google.com/java/lang/String.html) | [**JAXB\_ENCODING**](http://docs.google.com/javax/xml/bind/Marshaller.html#JAXB_ENCODING)            The name of the property used to specify the output encoding in the marshalled XML data. |
| static [String](http://docs.google.com/java/lang/String.html) | [**JAXB\_FORMATTED\_OUTPUT**](http://docs.google.com/javax/xml/bind/Marshaller.html#JAXB_FORMATTED_OUTPUT)            The name of the property used to specify whether or not the marshalled XML data is formatted with linefeeds and indentation. |
| static [String](http://docs.google.com/java/lang/String.html) | [**JAXB\_FRAGMENT**](http://docs.google.com/javax/xml/bind/Marshaller.html#JAXB_FRAGMENT)            The name of the property used to specify whether or not the marshaller will generate document level events (ie calling startDocument or endDocument). |
| static [String](http://docs.google.com/java/lang/String.html) | [**JAXB\_NO\_NAMESPACE\_SCHEMA\_LOCATION**](http://docs.google.com/javax/xml/bind/Marshaller.html#JAXB_NO_NAMESPACE_SCHEMA_LOCATION)            The name of the property used to specify the xsi:noNamespaceSchemaLocation attribute value to place in the marshalled XML output. |
| static [String](http://docs.google.com/java/lang/String.html) | [**JAXB\_SCHEMA\_LOCATION**](http://docs.google.com/javax/xml/bind/Marshaller.html#JAXB_SCHEMA_LOCATION)            The name of the property used to specify the xsi:schemaLocation attribute value to place in the marshalled XML output. |

| **Method Summary** | |
| --- | --- |
| | <A extends [XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html)>  A | | --- | | [**getAdapter**](http://docs.google.com/javax/xml/bind/Marshaller.html#getAdapter(java.lang.Class))([Class](http://docs.google.com/java/lang/Class.html)<A> type)            Gets the adapter associated with the specified type. |
| [AttachmentMarshaller](http://docs.google.com/javax/xml/bind/attachment/AttachmentMarshaller.html) | [**getAttachmentMarshaller**](http://docs.google.com/javax/xml/bind/Marshaller.html#getAttachmentMarshaller())() |
| [ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) | [**getEventHandler**](http://docs.google.com/javax/xml/bind/Marshaller.html#getEventHandler())()            Return the current event handler or the default event handler if one hasn't been set. |
| [Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) | [**getListener**](http://docs.google.com/javax/xml/bind/Marshaller.html#getListener())()            Return [Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) registered with this [Marshaller](http://docs.google.com/javax/xml/bind/Marshaller.html). |
| [Node](http://docs.google.com/org/w3c/dom/Node.html) | [**getNode**](http://docs.google.com/javax/xml/bind/Marshaller.html#getNode(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) contentTree)            Get a DOM tree view of the content tree(Optional). |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getProperty**](http://docs.google.com/javax/xml/bind/Marshaller.html#getProperty(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Get the particular property in the underlying implementation of Marshaller. |
| [Schema](http://docs.google.com/javax/xml/validation/Schema.html) | [**getSchema**](http://docs.google.com/javax/xml/bind/Marshaller.html#getSchema())()            Get the JAXP 1.3 [Schema](http://docs.google.com/javax/xml/validation/Schema.html) object being used to perform marshal-time validation. |
| void | [**marshal**](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20org.xml.sax.ContentHandler))([Object](http://docs.google.com/java/lang/Object.html) jaxbElement, [ContentHandler](http://docs.google.com/org/xml/sax/ContentHandler.html) handler)            Marshal the content tree rooted at jaxbElement into SAX2 events. |
| void | [**marshal**](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20org.w3c.dom.Node))([Object](http://docs.google.com/java/lang/Object.html) jaxbElement, [Node](http://docs.google.com/org/w3c/dom/Node.html) node)            Marshal the content tree rooted at jaxbElement into a DOM tree. |
| void | [**marshal**](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20java.io.OutputStream))([Object](http://docs.google.com/java/lang/Object.html) jaxbElement, [OutputStream](http://docs.google.com/java/io/OutputStream.html) os)            Marshal the content tree rooted at jaxbElement into an output stream. |
| void | [**marshal**](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20javax.xml.transform.Result))([Object](http://docs.google.com/java/lang/Object.html) jaxbElement, [Result](http://docs.google.com/javax/xml/transform/Result.html) result)            Marshal the content tree rooted at jaxbElement into the specified javax.xml.transform.Result. |
| void | [**marshal**](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20java.io.Writer))([Object](http://docs.google.com/java/lang/Object.html) jaxbElement, [Writer](http://docs.google.com/java/io/Writer.html) writer)            Marshal the content tree rooted at jaxbElement into a Writer. |
| void | [**marshal**](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20javax.xml.stream.XMLEventWriter))([Object](http://docs.google.com/java/lang/Object.html) jaxbElement, [XMLEventWriter](http://docs.google.com/javax/xml/stream/XMLEventWriter.html) writer)            Marshal the content tree rooted at jaxbElement into a [XMLEventWriter](http://docs.google.com/javax/xml/stream/XMLEventWriter.html). |
| void | [**marshal**](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20javax.xml.stream.XMLStreamWriter))([Object](http://docs.google.com/java/lang/Object.html) jaxbElement, [XMLStreamWriter](http://docs.google.com/javax/xml/stream/XMLStreamWriter.html) writer)            Marshal the content tree rooted at jaxbElement into a [XMLStreamWriter](http://docs.google.com/javax/xml/stream/XMLStreamWriter.html). |
| | <A extends [XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html)>  void | | --- | | [**setAdapter**](http://docs.google.com/javax/xml/bind/Marshaller.html#setAdapter(java.lang.Class,%20A))([Class](http://docs.google.com/java/lang/Class.html)<A> type, A adapter)            Associates a configured instance of [XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html) with this marshaller. |
| void | [**setAdapter**](http://docs.google.com/javax/xml/bind/Marshaller.html#setAdapter(javax.xml.bind.annotation.adapters.XmlAdapter))([XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html) adapter)            Associates a configured instance of [XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html) with this marshaller. |
| void | [**setAttachmentMarshaller**](http://docs.google.com/javax/xml/bind/Marshaller.html#setAttachmentMarshaller(javax.xml.bind.attachment.AttachmentMarshaller))([AttachmentMarshaller](http://docs.google.com/javax/xml/bind/attachment/AttachmentMarshaller.html) am)            Associate a context that enables binary data within an XML document to be transmitted as XML-binary optimized attachment. |
| void | [**setEventHandler**](http://docs.google.com/javax/xml/bind/Marshaller.html#setEventHandler(javax.xml.bind.ValidationEventHandler))([ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) handler)            Allow an application to register a validation event handler. |
| void | [**setListener**](http://docs.google.com/javax/xml/bind/Marshaller.html#setListener(javax.xml.bind.Marshaller.Listener))([Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) listener)             Register marshal event callback [Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) with this [Marshaller](http://docs.google.com/javax/xml/bind/Marshaller.html). |
| void | [**setProperty**](http://docs.google.com/javax/xml/bind/Marshaller.html#setProperty(java.lang.String,%20java.lang.Object))([String](http://docs.google.com/java/lang/String.html) name, [Object](http://docs.google.com/java/lang/Object.html) value)            Set the particular property in the underlying implementation of Marshaller. |
| void | [**setSchema**](http://docs.google.com/javax/xml/bind/Marshaller.html#setSchema(javax.xml.validation.Schema))([Schema](http://docs.google.com/javax/xml/validation/Schema.html) schema)            Specify the JAXP 1.3 [Schema](http://docs.google.com/javax/xml/validation/Schema.html) object that should be used to validate subsequent marshal operations against. |

| **Field Detail** |
| --- |

### JAXB\_ENCODING

static final [String](http://docs.google.com/java/lang/String.html) **JAXB\_ENCODING**

The name of the property used to specify the output encoding in the marshalled XML data.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#javax.xml.bind.Marshaller.JAXB_ENCODING)

### JAXB\_FORMATTED\_OUTPUT

static final [String](http://docs.google.com/java/lang/String.html) **JAXB\_FORMATTED\_OUTPUT**

The name of the property used to specify whether or not the marshalled XML data is formatted with linefeeds and indentation.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#javax.xml.bind.Marshaller.JAXB_FORMATTED_OUTPUT)

### JAXB\_SCHEMA\_LOCATION

static final [String](http://docs.google.com/java/lang/String.html) **JAXB\_SCHEMA\_LOCATION**

The name of the property used to specify the xsi:schemaLocation attribute value to place in the marshalled XML output.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#javax.xml.bind.Marshaller.JAXB_SCHEMA_LOCATION)

### JAXB\_NO\_NAMESPACE\_SCHEMA\_LOCATION

static final [String](http://docs.google.com/java/lang/String.html) **JAXB\_NO\_NAMESPACE\_SCHEMA\_LOCATION**

The name of the property used to specify the xsi:noNamespaceSchemaLocation attribute value to place in the marshalled XML output.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#javax.xml.bind.Marshaller.JAXB_NO_NAMESPACE_SCHEMA_LOCATION)

### JAXB\_FRAGMENT

static final [String](http://docs.google.com/java/lang/String.html) **JAXB\_FRAGMENT**

The name of the property used to specify whether or not the marshaller will generate document level events (ie calling startDocument or endDocument).

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#javax.xml.bind.Marshaller.JAXB_FRAGMENT)

| **Method Detail** |
| --- |

### marshal

void **marshal**([Object](http://docs.google.com/java/lang/Object.html) jaxbElement,  
 [Result](http://docs.google.com/javax/xml/transform/Result.html) result)  
 throws [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html)

Marshal the content tree rooted at jaxbElement into the specified javax.xml.transform.Result.

All JAXB Providers must at least support [DOMResult](http://docs.google.com/javax/xml/transform/dom/DOMResult.html), [SAXResult](http://docs.google.com/javax/xml/transform/sax/SAXResult.html), and [StreamResult](http://docs.google.com/javax/xml/transform/stream/StreamResult.html). It can support other derived classes of Result as well.

**Parameters:**jaxbElement - The root of content tree to be marshalled.result - XML will be sent to this Result **Throws:** [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html) - If any unexpected problem occurs during the marshalling. [MarshalException](http://docs.google.com/javax/xml/bind/MarshalException.html) - If the [ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) returns false from its handleEvent method or the Marshaller is unable to marshal obj (or any object reachable from obj). See  [Marshalling a JAXB element](#3znysh7). [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If any of the method parameters are null

### marshal

void **marshal**([Object](http://docs.google.com/java/lang/Object.html) jaxbElement,  
 [OutputStream](http://docs.google.com/java/io/OutputStream.html) os)  
 throws [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html)

Marshal the content tree rooted at jaxbElement into an output stream.

**Parameters:**jaxbElement - The root of content tree to be marshalled.os - XML will be added to this stream. **Throws:** [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html) - If any unexpected problem occurs during the marshalling. [MarshalException](http://docs.google.com/javax/xml/bind/MarshalException.html) - If the [ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) returns false from its handleEvent method or the Marshaller is unable to marshal obj (or any object reachable from obj). See  [Marshalling a JAXB element](#3znysh7). [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If any of the method parameters are null

### marshal

void **marshal**([Object](http://docs.google.com/java/lang/Object.html) jaxbElement,  
 [Writer](http://docs.google.com/java/io/Writer.html) writer)  
 throws [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html)

Marshal the content tree rooted at jaxbElement into a Writer.

**Parameters:**jaxbElement - The root of content tree to be marshalled.writer - XML will be sent to this writer. **Throws:** [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html) - If any unexpected problem occurs during the marshalling. [MarshalException](http://docs.google.com/javax/xml/bind/MarshalException.html) - If the [ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) returns false from its handleEvent method or the Marshaller is unable to marshal obj (or any object reachable from obj). See  [Marshalling a JAXB element](#3znysh7). [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If any of the method parameters are null

### marshal

void **marshal**([Object](http://docs.google.com/java/lang/Object.html) jaxbElement,  
 [ContentHandler](http://docs.google.com/org/xml/sax/ContentHandler.html) handler)  
 throws [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html)

Marshal the content tree rooted at jaxbElement into SAX2 events.

**Parameters:**jaxbElement - The root of content tree to be marshalled.handler - XML will be sent to this handler as SAX2 events. **Throws:** [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html) - If any unexpected problem occurs during the marshalling. [MarshalException](http://docs.google.com/javax/xml/bind/MarshalException.html) - If the [ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) returns false from its handleEvent method or the Marshaller is unable to marshal obj (or any object reachable from obj). See  [Marshalling a JAXB element](#3znysh7). [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If any of the method parameters are null

### marshal

void **marshal**([Object](http://docs.google.com/java/lang/Object.html) jaxbElement,  
 [Node](http://docs.google.com/org/w3c/dom/Node.html) node)  
 throws [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html)

Marshal the content tree rooted at jaxbElement into a DOM tree.

**Parameters:**jaxbElement - The content tree to be marshalled.node - DOM nodes will be added as children of this node. This parameter must be a Node that accepts children ([Document](http://docs.google.com/org/w3c/dom/Document.html), [DocumentFragment](http://docs.google.com/org/w3c/dom/DocumentFragment.html), or [Element](http://docs.google.com/org/w3c/dom/Element.html)) **Throws:** [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html) - If any unexpected problem occurs during the marshalling. [MarshalException](http://docs.google.com/javax/xml/bind/MarshalException.html) - If the [ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) returns false from its handleEvent method or the Marshaller is unable to marshal jaxbElement (or any object reachable from jaxbElement). See  [Marshalling a JAXB element](#3znysh7). [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If any of the method parameters are null

### marshal

void **marshal**([Object](http://docs.google.com/java/lang/Object.html) jaxbElement,  
 [XMLStreamWriter](http://docs.google.com/javax/xml/stream/XMLStreamWriter.html) writer)  
 throws [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html)

Marshal the content tree rooted at jaxbElement into a [XMLStreamWriter](http://docs.google.com/javax/xml/stream/XMLStreamWriter.html).

**Parameters:**jaxbElement - The content tree to be marshalled.writer - XML will be sent to this writer. **Throws:** [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html) - If any unexpected problem occurs during the marshalling. [MarshalException](http://docs.google.com/javax/xml/bind/MarshalException.html) - If the [ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) returns false from its handleEvent method or the Marshaller is unable to marshal obj (or any object reachable from obj). See  [Marshalling a JAXB element](#3znysh7). [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If any of the method parameters are null**Since:** JAXB 2.0

### marshal

void **marshal**([Object](http://docs.google.com/java/lang/Object.html) jaxbElement,  
 [XMLEventWriter](http://docs.google.com/javax/xml/stream/XMLEventWriter.html) writer)  
 throws [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html)

Marshal the content tree rooted at jaxbElement into a [XMLEventWriter](http://docs.google.com/javax/xml/stream/XMLEventWriter.html).

**Parameters:**jaxbElement - The content tree rooted at jaxbElement to be marshalled.writer - XML will be sent to this writer. **Throws:** [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html) - If any unexpected problem occurs during the marshalling. [MarshalException](http://docs.google.com/javax/xml/bind/MarshalException.html) - If the [ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) returns false from its handleEvent method or the Marshaller is unable to marshal obj (or any object reachable from obj). See  [Marshalling a JAXB element](#3znysh7). [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If any of the method parameters are null**Since:** JAXB 2.0

### getNode

[Node](http://docs.google.com/org/w3c/dom/Node.html) **getNode**([Object](http://docs.google.com/java/lang/Object.html) contentTree)  
 throws [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html)

Get a DOM tree view of the content tree(Optional). If the returned DOM tree is updated, these changes are also visible in the content tree. Use [marshal(Object, org.w3c.dom.Node)](http://docs.google.com/javax/xml/bind/Marshaller.html#marshal(java.lang.Object,%20org.w3c.dom.Node)) to force a deep copy of the content tree to a DOM representation.

**Parameters:**contentTree - - JAXB Java representation of XML content **Returns:**the DOM tree view of the contentTree **Throws:** [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - If the JAXB provider implementation does not support a DOM view of the content tree [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If any of the method parameters are null [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html) - If any unexpected problem occurs

### setProperty

void **setProperty**([String](http://docs.google.com/java/lang/String.html) name,  
 [Object](http://docs.google.com/java/lang/Object.html) value)  
 throws [PropertyException](http://docs.google.com/javax/xml/bind/PropertyException.html)

Set the particular property in the underlying implementation of Marshaller. This method can only be used to set one of the standard JAXB defined properties above or a provider specific property. Attempting to set an undefined property will result in a PropertyException being thrown. See  [Supported Properties](#2et92p0).

**Parameters:**name - the name of the property to be set. This value can either be specified using one of the constant fields or a user supplied string.value - the value of the property to be set **Throws:** [PropertyException](http://docs.google.com/javax/xml/bind/PropertyException.html) - when there is an error processing the given property or value [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If the name parameter is null

### getProperty

[Object](http://docs.google.com/java/lang/Object.html) **getProperty**([String](http://docs.google.com/java/lang/String.html) name)  
 throws [PropertyException](http://docs.google.com/javax/xml/bind/PropertyException.html)

Get the particular property in the underlying implementation of Marshaller. This method can only be used to get one of the standard JAXB defined properties above or a provider specific property. Attempting to get an undefined property will result in a PropertyException being thrown. See  [Supported Properties](#2et92p0).

**Parameters:**name - the name of the property to retrieve **Returns:**the value of the requested property **Throws:** [PropertyException](http://docs.google.com/javax/xml/bind/PropertyException.html) - when there is an error retrieving the given property or value property name [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If the name parameter is null

### setEventHandler

void **setEventHandler**([ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) handler)  
 throws [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html)

Allow an application to register a validation event handler.

The validation event handler will be called by the JAXB Provider if any validation errors are encountered during calls to any of the marshal API's. If the client application does not register a validation event handler before invoking one of the marshal methods, then validation events will be handled by the default event handler which will terminate the marshal operation after the first error or fatal error is encountered.

Calling this method with a null parameter will cause the Marshaller to revert back to the default default event handler.

**Parameters:**handler - the validation event handler **Throws:** [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html) - if an error was encountered while setting the event handler

### getEventHandler

[ValidationEventHandler](http://docs.google.com/javax/xml/bind/ValidationEventHandler.html) **getEventHandler**()  
 throws [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html)

Return the current event handler or the default event handler if one hasn't been set.

**Returns:**the current ValidationEventHandler or the default event handler if it hasn't been set **Throws:** [JAXBException](http://docs.google.com/javax/xml/bind/JAXBException.html) - if an error was encountered while getting the current event handler

### setAdapter

void **setAdapter**([XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html) adapter)

Associates a configured instance of [XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html) with this marshaller.

This is a convenience method that invokes setAdapter(adapter.getClass(),adapter);.

**Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the adapter parameter is null. [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if invoked agains a JAXB 1.0 implementation.**Since:** JAXB 2.0 **See Also:**[setAdapter(Class,XmlAdapter)](http://docs.google.com/javax/xml/bind/Marshaller.html#setAdapter(java.lang.Class,%20A))

### setAdapter

<A extends [XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html)> void **setAdapter**([Class](http://docs.google.com/java/lang/Class.html)<A> type,  
 A adapter)

Associates a configured instance of [XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html) with this marshaller.

Every marshaller internally maintains a [Map](http://docs.google.com/java/util/Map.html)<[Class](http://docs.google.com/java/lang/Class.html),[XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html)>, which it uses for marshalling classes whose fields/methods are annotated with [XmlJavaTypeAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlJavaTypeAdapter.html).

This method allows applications to use a configured instance of [XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html). When an instance of an adapter is not given, a marshaller will create one by invoking its default constructor.

**Parameters:**type - The type of the adapter. The specified instance will be used when [XmlJavaTypeAdapter.value()](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlJavaTypeAdapter.html#value()) refers to this type.adapter - The instance of the adapter to be used. If null, it will un-register the current adapter set for this type. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the type parameter is null. [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if invoked agains a JAXB 1.0 implementation.**Since:** JAXB 2.0

### getAdapter

<A extends [XmlAdapter](http://docs.google.com/javax/xml/bind/annotation/adapters/XmlAdapter.html)> A **getAdapter**([Class](http://docs.google.com/java/lang/Class.html)<A> type)

Gets the adapter associated with the specified type. This is the reverse operation of the [setAdapter(javax.xml.bind.annotation.adapters.XmlAdapter)](http://docs.google.com/javax/xml/bind/Marshaller.html#setAdapter(javax.xml.bind.annotation.adapters.XmlAdapter)) method.

**Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the type parameter is null. [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - if invoked agains a JAXB 1.0 implementation.**Since:** JAXB 2.0

### setAttachmentMarshaller

void **setAttachmentMarshaller**([AttachmentMarshaller](http://docs.google.com/javax/xml/bind/attachment/AttachmentMarshaller.html) am)

Associate a context that enables binary data within an XML document to be transmitted as XML-binary optimized attachment. The attachment is referenced from the XML document content model by content-id URIs(cid) references stored within the xml document.

**Throws:** [IllegalStateException](http://docs.google.com/java/lang/IllegalStateException.html) - if attempt to concurrently call this method during a marshal operation.

### getAttachmentMarshaller

[AttachmentMarshaller](http://docs.google.com/javax/xml/bind/attachment/AttachmentMarshaller.html) **getAttachmentMarshaller**()

### setSchema

void **setSchema**([Schema](http://docs.google.com/javax/xml/validation/Schema.html) schema)

Specify the JAXP 1.3 [Schema](http://docs.google.com/javax/xml/validation/Schema.html) object that should be used to validate subsequent marshal operations against. Passing null into this method will disable validation.

This method allows the caller to validate the marshalled XML as it's marshalled.

Initially this property is set to null.

**Parameters:**schema - Schema object to validate marshal operations against or null to disable validation **Throws:** [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - could be thrown if this method is invoked on an Marshaller created from a JAXBContext referencing JAXB 1.0 mapped classes**Since:** JAXB2.0

### getSchema

[Schema](http://docs.google.com/javax/xml/validation/Schema.html) **getSchema**()

Get the JAXP 1.3 [Schema](http://docs.google.com/javax/xml/validation/Schema.html) object being used to perform marshal-time validation. If there is no Schema set on the marshaller, then this method will return null indicating that marshal-time validation will not be performed.

**Returns:**the Schema object being used to perform marshal-time validation or null if not present. **Throws:** [UnsupportedOperationException](http://docs.google.com/java/lang/UnsupportedOperationException.html) - could be thrown if this method is invoked on an Marshaller created from a JAXBContext referencing JAXB 1.0 mapped classes**Since:** JAXB2.0

### setListener

void **setListener**([Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) listener)

Register marshal event callback [Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) with this [Marshaller](http://docs.google.com/javax/xml/bind/Marshaller.html).

There is only one Listener per Marshaller. Setting a Listener replaces the previous set Listener. One can unregister current Listener by setting listener to null.

**Parameters:**listener - an instance of a class that implements [Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html)**Since:** JAXB2.0

### getListener

[Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) **getListener**()

Return [Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) registered with this [Marshaller](http://docs.google.com/javax/xml/bind/Marshaller.html).

**Returns:**registered [Marshaller.Listener](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) or null if no Listener is registered with this Marshaller.**Since:** JAXB2.0

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Marshaller.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/MarshalException.html)   [**NEXT CLASS**](http://docs.google.com/javax/xml/bind/Marshaller.Listener.html) | [**FRAMES**](http://docs.google.com/index.html?javax/xml/bind/Marshaller.html)    [**NO FRAMES**](http://docs.google.com/Marshaller.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3dy6vkm) | [FIELD](#1t3h5sf) | CONSTR | [METHOD](#4d34og8) | DETAIL: [FIELD](#2s8eyo1) | CONSTR | [METHOD](#1ksv4uv) |

[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).