| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/XMLReader.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/org/xml/sax/XMLFilter.html)   NEXT CLASS | [**FRAMES**](http://docs.google.com/index.html?org/xml/sax/XMLReader.html)    [**NO FRAMES**](http://docs.google.com/XMLReader.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **org.xml.sax**

Interface XMLReader

**All Known Subinterfaces:** [XMLFilter](http://docs.google.com/org/xml/sax/XMLFilter.html) **All Known Implementing Classes:** [ParserAdapter](http://docs.google.com/org/xml/sax/helpers/ParserAdapter.html), [XMLFilterImpl](http://docs.google.com/org/xml/sax/helpers/XMLFilterImpl.html)

public interface **XMLReader**

Interface for reading an XML document using callbacks.

*This module, both source code and documentation, is in the Public Domain, and comes with* ***NO WARRANTY****.* See <http://www.saxproject.org> for further information.

**Note:** despite its name, this interface does *not* extend the standard Java [Reader](http://docs.google.com/java/io/Reader.html) interface, because reading XML is a fundamentally different activity than reading character data.

XMLReader is the interface that an XML parser's SAX2 driver must implement. This interface allows an application to set and query features and properties in the parser, to register event handlers for document processing, and to initiate a document parse.

All SAX interfaces are assumed to be synchronous: the [parse](http://docs.google.com/org/xml/sax/XMLReader.html#parse(org.xml.sax.InputSource)) methods must not return until parsing is complete, and readers must wait for an event-handler callback to return before reporting the next event.

This interface replaces the (now deprecated) SAX 1.0 [Parser](http://docs.google.com/org/xml/sax/Parser.html) interface. The XMLReader interface contains two important enhancements over the old Parser interface (as well as some minor ones):

1. it adds a standard way to query and set features and properties; and
2. it adds Namespace support, which is required for many higher-level XML standards.

There are adapters available to convert a SAX1 Parser to a SAX2 XMLReader and vice-versa.

**Since:** SAX 2.0 **See Also:**[XMLFilter](http://docs.google.com/org/xml/sax/XMLFilter.html), [ParserAdapter](http://docs.google.com/org/xml/sax/helpers/ParserAdapter.html), [XMLReaderAdapter](http://docs.google.com/org/xml/sax/helpers/XMLReaderAdapter.html)

| **Method Summary** | |
| --- | --- |
| [ContentHandler](http://docs.google.com/org/xml/sax/ContentHandler.html) | [**getContentHandler**](http://docs.google.com/org/xml/sax/XMLReader.html#getContentHandler())()            Return the current content handler. |
| [DTDHandler](http://docs.google.com/org/xml/sax/DTDHandler.html) | [**getDTDHandler**](http://docs.google.com/org/xml/sax/XMLReader.html#getDTDHandler())()            Return the current DTD handler. |
| [EntityResolver](http://docs.google.com/org/xml/sax/EntityResolver.html) | [**getEntityResolver**](http://docs.google.com/org/xml/sax/XMLReader.html#getEntityResolver())()            Return the current entity resolver. |
| [ErrorHandler](http://docs.google.com/org/xml/sax/ErrorHandler.html) | [**getErrorHandler**](http://docs.google.com/org/xml/sax/XMLReader.html#getErrorHandler())()            Return the current error handler. |
| boolean | [**getFeature**](http://docs.google.com/org/xml/sax/XMLReader.html#getFeature(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Look up the value of a feature flag. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getProperty**](http://docs.google.com/org/xml/sax/XMLReader.html#getProperty(java.lang.String))([String](http://docs.google.com/java/lang/String.html) name)            Look up the value of a property. |
| void | [**parse**](http://docs.google.com/org/xml/sax/XMLReader.html#parse(org.xml.sax.InputSource))([InputSource](http://docs.google.com/org/xml/sax/InputSource.html) input)            Parse an XML document. |
| void | [**parse**](http://docs.google.com/org/xml/sax/XMLReader.html#parse(java.lang.String))([String](http://docs.google.com/java/lang/String.html) systemId)            Parse an XML document from a system identifier (URI). |
| void | [**setContentHandler**](http://docs.google.com/org/xml/sax/XMLReader.html#setContentHandler(org.xml.sax.ContentHandler))([ContentHandler](http://docs.google.com/org/xml/sax/ContentHandler.html) handler)            Allow an application to register a content event handler. |
| void | [**setDTDHandler**](http://docs.google.com/org/xml/sax/XMLReader.html#setDTDHandler(org.xml.sax.DTDHandler))([DTDHandler](http://docs.google.com/org/xml/sax/DTDHandler.html) handler)            Allow an application to register a DTD event handler. |
| void | [**setEntityResolver**](http://docs.google.com/org/xml/sax/XMLReader.html#setEntityResolver(org.xml.sax.EntityResolver))([EntityResolver](http://docs.google.com/org/xml/sax/EntityResolver.html) resolver)            Allow an application to register an entity resolver. |
| void | [**setErrorHandler**](http://docs.google.com/org/xml/sax/XMLReader.html#setErrorHandler(org.xml.sax.ErrorHandler))([ErrorHandler](http://docs.google.com/org/xml/sax/ErrorHandler.html) handler)            Allow an application to register an error event handler. |
| void | [**setFeature**](http://docs.google.com/org/xml/sax/XMLReader.html#setFeature(java.lang.String,%20boolean))([String](http://docs.google.com/java/lang/String.html) name, boolean value)            Set the value of a feature flag. |
| void | [**setProperty**](http://docs.google.com/org/xml/sax/XMLReader.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 value of a property. |

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

### getFeature

boolean **getFeature**([String](http://docs.google.com/java/lang/String.html) name)  
 throws [SAXNotRecognizedException](http://docs.google.com/org/xml/sax/SAXNotRecognizedException.html),  
 [SAXNotSupportedException](http://docs.google.com/org/xml/sax/SAXNotSupportedException.html)

Look up the value of a feature flag.

The feature name is any fully-qualified URI. It is possible for an XMLReader to recognize a feature name but temporarily be unable to return its value. Some feature values may be available only in specific contexts, such as before, during, or after a parse. Also, some feature values may not be programmatically accessible. (In the case of an adapter for SAX1 [Parser](http://docs.google.com/org/xml/sax/Parser.html), there is no implementation-independent way to expose whether the underlying parser is performing validation, expanding external entities, and so forth.)

All XMLReaders are required to recognize the http://xml.org/sax/features/namespaces and the http://xml.org/sax/features/namespace-prefixes feature names.

Typical usage is something like this:

XMLReader r = new MySAXDriver();  
  
 // try to activate validation  
 try {  
 r.setFeature("http://xml.org/sax/features/validation", true);  
 } catch (SAXException e) {  
 System.err.println("Cannot activate validation.");   
 }  
  
 // register event handlers  
 r.setContentHandler(new MyContentHandler());  
 r.setErrorHandler(new MyErrorHandler());  
  
 // parse the first document  
 try {  
 r.parse("http://www.foo.com/mydoc.xml");  
 } catch (IOException e) {  
 System.err.println("I/O exception reading XML document");  
 } catch (SAXException e) {  
 System.err.println("XML exception reading document.");  
 }

Implementors are free (and encouraged) to invent their own features, using names built on their own URIs.

**Parameters:**name - The feature name, which is a fully-qualified URI. **Returns:**The current value of the feature (true or false). **Throws:** [SAXNotRecognizedException](http://docs.google.com/org/xml/sax/SAXNotRecognizedException.html) - If the feature value can't be assigned or retrieved. [SAXNotSupportedException](http://docs.google.com/org/xml/sax/SAXNotSupportedException.html) - When the XMLReader recognizes the feature name but cannot determine its value at this time.**See Also:**[setFeature(java.lang.String, boolean)](http://docs.google.com/org/xml/sax/XMLReader.html#setFeature(java.lang.String,%20boolean))

### setFeature

void **setFeature**([String](http://docs.google.com/java/lang/String.html) name,  
 boolean value)  
 throws [SAXNotRecognizedException](http://docs.google.com/org/xml/sax/SAXNotRecognizedException.html),  
 [SAXNotSupportedException](http://docs.google.com/org/xml/sax/SAXNotSupportedException.html)

Set the value of a feature flag.

The feature name is any fully-qualified URI. It is possible for an XMLReader to expose a feature value but to be unable to change the current value. Some feature values may be immutable or mutable only in specific contexts, such as before, during, or after a parse.

All XMLReaders are required to support setting http://xml.org/sax/features/namespaces to true and http://xml.org/sax/features/namespace-prefixes to false.

**Parameters:**name - The feature name, which is a fully-qualified URI.value - The requested value of the feature (true or false). **Throws:** [SAXNotRecognizedException](http://docs.google.com/org/xml/sax/SAXNotRecognizedException.html) - If the feature value can't be assigned or retrieved. [SAXNotSupportedException](http://docs.google.com/org/xml/sax/SAXNotSupportedException.html) - When the XMLReader recognizes the feature name but cannot set the requested value.**See Also:**[getFeature(java.lang.String)](http://docs.google.com/org/xml/sax/XMLReader.html#getFeature(java.lang.String))

### getProperty

[Object](http://docs.google.com/java/lang/Object.html) **getProperty**([String](http://docs.google.com/java/lang/String.html) name)  
 throws [SAXNotRecognizedException](http://docs.google.com/org/xml/sax/SAXNotRecognizedException.html),  
 [SAXNotSupportedException](http://docs.google.com/org/xml/sax/SAXNotSupportedException.html)

Look up the value of a property.

The property name is any fully-qualified URI. It is possible for an XMLReader to recognize a property name but temporarily be unable to return its value. Some property values may be available only in specific contexts, such as before, during, or after a parse.

XMLReaders are not required to recognize any specific property names, though an initial core set is documented for SAX2.

Implementors are free (and encouraged) to invent their own properties, using names built on their own URIs.

**Parameters:**name - The property name, which is a fully-qualified URI. **Returns:**The current value of the property. **Throws:** [SAXNotRecognizedException](http://docs.google.com/org/xml/sax/SAXNotRecognizedException.html) - If the property value can't be assigned or retrieved. [SAXNotSupportedException](http://docs.google.com/org/xml/sax/SAXNotSupportedException.html) - When the XMLReader recognizes the property name but cannot determine its value at this time.**See Also:**[setProperty(java.lang.String, java.lang.Object)](http://docs.google.com/org/xml/sax/XMLReader.html#setProperty(java.lang.String,%20java.lang.Object))

### setProperty

void **setProperty**([String](http://docs.google.com/java/lang/String.html) name,  
 [Object](http://docs.google.com/java/lang/Object.html) value)  
 throws [SAXNotRecognizedException](http://docs.google.com/org/xml/sax/SAXNotRecognizedException.html),  
 [SAXNotSupportedException](http://docs.google.com/org/xml/sax/SAXNotSupportedException.html)

Set the value of a property.

The property name is any fully-qualified URI. It is possible for an XMLReader to recognize a property name but to be unable to change the current value. Some property values may be immutable or mutable only in specific contexts, such as before, during, or after a parse.

XMLReaders are not required to recognize setting any specific property names, though a core set is defined by SAX2.

This method is also the standard mechanism for setting extended handlers.

**Parameters:**name - The property name, which is a fully-qualified URI.value - The requested value for the property. **Throws:** [SAXNotRecognizedException](http://docs.google.com/org/xml/sax/SAXNotRecognizedException.html) - If the property value can't be assigned or retrieved. [SAXNotSupportedException](http://docs.google.com/org/xml/sax/SAXNotSupportedException.html) - When the XMLReader recognizes the property name but cannot set the requested value.

### setEntityResolver

void **setEntityResolver**([EntityResolver](http://docs.google.com/org/xml/sax/EntityResolver.html) resolver)

Allow an application to register an entity resolver.

If the application does not register an entity resolver, the XMLReader will perform its own default resolution.

Applications may register a new or different resolver in the middle of a parse, and the SAX parser must begin using the new resolver immediately.

**Parameters:**resolver - The entity resolver.**See Also:**[getEntityResolver()](http://docs.google.com/org/xml/sax/XMLReader.html#getEntityResolver())

### getEntityResolver

[EntityResolver](http://docs.google.com/org/xml/sax/EntityResolver.html) **getEntityResolver**()

Return the current entity resolver.

**Returns:**The current entity resolver, or null if none has been registered.**See Also:**[setEntityResolver(org.xml.sax.EntityResolver)](http://docs.google.com/org/xml/sax/XMLReader.html#setEntityResolver(org.xml.sax.EntityResolver))

### setDTDHandler

void **setDTDHandler**([DTDHandler](http://docs.google.com/org/xml/sax/DTDHandler.html) handler)

Allow an application to register a DTD event handler.

If the application does not register a DTD handler, all DTD events reported by the SAX parser will be silently ignored.

Applications may register a new or different handler in the middle of a parse, and the SAX parser must begin using the new handler immediately.

**Parameters:**handler - The DTD handler.**See Also:**[getDTDHandler()](http://docs.google.com/org/xml/sax/XMLReader.html#getDTDHandler())

### getDTDHandler

[DTDHandler](http://docs.google.com/org/xml/sax/DTDHandler.html) **getDTDHandler**()

Return the current DTD handler.

**Returns:**The current DTD handler, or null if none has been registered.**See Also:**[setDTDHandler(org.xml.sax.DTDHandler)](http://docs.google.com/org/xml/sax/XMLReader.html#setDTDHandler(org.xml.sax.DTDHandler))

### setContentHandler

void **setContentHandler**([ContentHandler](http://docs.google.com/org/xml/sax/ContentHandler.html) handler)

Allow an application to register a content event handler.

If the application does not register a content handler, all content events reported by the SAX parser will be silently ignored.

Applications may register a new or different handler in the middle of a parse, and the SAX parser must begin using the new handler immediately.

**Parameters:**handler - The content handler.**See Also:**[getContentHandler()](http://docs.google.com/org/xml/sax/XMLReader.html#getContentHandler())

### getContentHandler

[ContentHandler](http://docs.google.com/org/xml/sax/ContentHandler.html) **getContentHandler**()

Return the current content handler.

**Returns:**The current content handler, or null if none has been registered.**See Also:**[setContentHandler(org.xml.sax.ContentHandler)](http://docs.google.com/org/xml/sax/XMLReader.html#setContentHandler(org.xml.sax.ContentHandler))

### setErrorHandler

void **setErrorHandler**([ErrorHandler](http://docs.google.com/org/xml/sax/ErrorHandler.html) handler)

Allow an application to register an error event handler.

If the application does not register an error handler, all error events reported by the SAX parser will be silently ignored; however, normal processing may not continue. It is highly recommended that all SAX applications implement an error handler to avoid unexpected bugs.

Applications may register a new or different handler in the middle of a parse, and the SAX parser must begin using the new handler immediately.

**Parameters:**handler - The error handler.**See Also:**[getErrorHandler()](http://docs.google.com/org/xml/sax/XMLReader.html#getErrorHandler())

### getErrorHandler

[ErrorHandler](http://docs.google.com/org/xml/sax/ErrorHandler.html) **getErrorHandler**()

Return the current error handler.

**Returns:**The current error handler, or null if none has been registered.**See Also:**[setErrorHandler(org.xml.sax.ErrorHandler)](http://docs.google.com/org/xml/sax/XMLReader.html#setErrorHandler(org.xml.sax.ErrorHandler))

### parse

void **parse**([InputSource](http://docs.google.com/org/xml/sax/InputSource.html) input)  
 throws [IOException](http://docs.google.com/java/io/IOException.html),  
 [SAXException](http://docs.google.com/org/xml/sax/SAXException.html)

Parse an XML document.

The application can use this method to instruct the XML reader to begin parsing an XML document from any valid input source (a character stream, a byte stream, or a URI).

Applications may not invoke this method while a parse is in progress (they should create a new XMLReader instead for each nested XML document). Once a parse is complete, an application may reuse the same XMLReader object, possibly with a different input source. Configuration of the XMLReader object (such as handler bindings and values established for feature flags and properties) is unchanged by completion of a parse, unless the definition of that aspect of the configuration explicitly specifies other behavior. (For example, feature flags or properties exposing characteristics of the document being parsed.)

During the parse, the XMLReader will provide information about the XML document through the registered event handlers.

This method is synchronous: it will not return until parsing has ended. If a client application wants to terminate parsing early, it should throw an exception.

**Parameters:**input - The input source for the top-level of the XML document. **Throws:** [SAXException](http://docs.google.com/org/xml/sax/SAXException.html) - Any SAX exception, possibly wrapping another exception. [IOException](http://docs.google.com/java/io/IOException.html) - An IO exception from the parser, possibly from a byte stream or character stream supplied by the application.**See Also:**[InputSource](http://docs.google.com/org/xml/sax/InputSource.html), [parse(java.lang.String)](http://docs.google.com/org/xml/sax/XMLReader.html#parse(java.lang.String)), [setEntityResolver(org.xml.sax.EntityResolver)](http://docs.google.com/org/xml/sax/XMLReader.html#setEntityResolver(org.xml.sax.EntityResolver)), [setDTDHandler(org.xml.sax.DTDHandler)](http://docs.google.com/org/xml/sax/XMLReader.html#setDTDHandler(org.xml.sax.DTDHandler)), [setContentHandler(org.xml.sax.ContentHandler)](http://docs.google.com/org/xml/sax/XMLReader.html#setContentHandler(org.xml.sax.ContentHandler)), [setErrorHandler(org.xml.sax.ErrorHandler)](http://docs.google.com/org/xml/sax/XMLReader.html#setErrorHandler(org.xml.sax.ErrorHandler))

### parse

void **parse**([String](http://docs.google.com/java/lang/String.html) systemId)  
 throws [IOException](http://docs.google.com/java/io/IOException.html),  
 [SAXException](http://docs.google.com/org/xml/sax/SAXException.html)

Parse an XML document from a system identifier (URI).

This method is a shortcut for the common case of reading a document from a system identifier. It is the exact equivalent of the following:

parse(new InputSource(systemId));

If the system identifier is a URL, it must be fully resolved by the application before it is passed to the parser.

**Parameters:**systemId - The system identifier (URI). **Throws:** [SAXException](http://docs.google.com/org/xml/sax/SAXException.html) - Any SAX exception, possibly wrapping another exception. [IOException](http://docs.google.com/java/io/IOException.html) - An IO exception from the parser, possibly from a byte stream or character stream supplied by the application.**See Also:**[parse(org.xml.sax.InputSource)](http://docs.google.com/org/xml/sax/XMLReader.html#parse(org.xml.sax.InputSource))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/XMLReader.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/org/xml/sax/XMLFilter.html)   NEXT CLASS | [**FRAMES**](http://docs.google.com/index.html?org/xml/sax/XMLReader.html)    [**NO FRAMES**](http://docs.google.com/XMLReader.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[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).