| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AbstractInterruptibleChannel.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   [**NEXT CLASS**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html) | [**FRAMES**](http://docs.google.com/index.html?java/nio/channels/spi/AbstractInterruptibleChannel.html)    [**NO FRAMES**](http://docs.google.com/AbstractInterruptibleChannel.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#2s8eyo1) |

## **java.nio.channels.spi**

Class AbstractInterruptibleChannel

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **java.nio.channels.spi.AbstractInterruptibleChannel**

**All Implemented Interfaces:** [Closeable](http://docs.google.com/java/io/Closeable.html), [Channel](http://docs.google.com/java/nio/channels/Channel.html), [InterruptibleChannel](http://docs.google.com/java/nio/channels/InterruptibleChannel.html) **Direct Known Subclasses:** [FileChannel](http://docs.google.com/java/nio/channels/FileChannel.html), [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html)

public abstract class **AbstractInterruptibleChannel**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Channel](http://docs.google.com/java/nio/channels/Channel.html), [InterruptibleChannel](http://docs.google.com/java/nio/channels/InterruptibleChannel.html)

Base implementation class for interruptible channels.

This class encapsulates the low-level machinery required to implement the asynchronous closing and interruption of channels. A concrete channel class must invoke the [begin](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#begin()) and [end](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#end(boolean)) methods before and after, respectively, invoking an I/O operation that might block indefinitely. In order to ensure that the [end](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#end(boolean)) method is always invoked, these methods should be used within a try ... finally block:

boolean completed = false;  
 try {  
 begin();  
 completed = ...; // Perform blocking I/O operation  
 return ...; // Return result  
 } finally {  
 end(completed);  
 }

The completed argument to the [end](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#end(boolean)) method tells whether or not the I/O operation actually completed, that is, whether it had any effect that would be visible to the invoker. In the case of an operation that reads bytes, for example, this argument should be true if, and only if, some bytes were actually transferred into the invoker's target buffer.

A concrete channel class must also implement the [implCloseChannel](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#implCloseChannel()) method in such a way that if it is invoked while another thread is blocked in a native I/O operation upon the channel then that operation will immediately return, either by throwing an exception or by returning normally. If a thread is interrupted or the channel upon which it is blocked is asynchronously closed then the channel's [end](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#end(boolean)) method will throw the appropriate exception.

This class performs the synchronization required to implement the [Channel](http://docs.google.com/java/nio/channels/Channel.html) specification. Implementations of the [implCloseChannel](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#implCloseChannel()) method need not synchronize against other threads that might be attempting to close the channel.

**Since:** 1.4

| **Constructor Summary** | |
| --- | --- |
| protected | [**AbstractInterruptibleChannel**](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#AbstractInterruptibleChannel())()            Initializes a new instance of this class. |

| **Method Summary** | |
| --- | --- |
| protected  void | [**begin**](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#begin())()            Marks the beginning of an I/O operation that might block indefinitely. |
| void | [**close**](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#close())()            Closes this channel. |
| protected  void | [**end**](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#end(boolean))(boolean completed)            Marks the end of an I/O operation that might block indefinitely. |
| protected abstract  void | [**implCloseChannel**](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#implCloseChannel())()            Closes this channel. |
| boolean | [**isOpen**](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#isOpen())()            Tells whether or not this channel is open. |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Constructor Detail** |
| --- |

### AbstractInterruptibleChannel

protected **AbstractInterruptibleChannel**()

Initializes a new instance of this class.

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

### close

public final void **close**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Closes this channel.

If the channel has already been closed then this method returns immediately. Otherwise it marks the channel as closed and then invokes the [implCloseChannel](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#implCloseChannel()) method in order to complete the close operation.

**Specified by:**[close](http://docs.google.com/java/io/Closeable.html#close()) in interface [Closeable](http://docs.google.com/java/io/Closeable.html)**Specified by:**[close](http://docs.google.com/java/nio/channels/Channel.html#close()) in interface [Channel](http://docs.google.com/java/nio/channels/Channel.html)**Specified by:**[close](http://docs.google.com/java/nio/channels/InterruptibleChannel.html#close()) in interface [InterruptibleChannel](http://docs.google.com/java/nio/channels/InterruptibleChannel.html) **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - If an I/O error occurs

### implCloseChannel

protected abstract void **implCloseChannel**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Closes this channel.

This method is invoked by the [close](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#close()) method in order to perform the actual work of closing the channel. This method is only invoked if the channel has not yet been closed, and it is never invoked more than once.

An implementation of this method must arrange for any other thread that is blocked in an I/O operation upon this channel to return immediately, either by throwing an exception or by returning normally.

**Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - If an I/O error occurs while closing the channel

### isOpen

public final boolean **isOpen**()

**Description copied from interface:** [**Channel**](http://docs.google.com/java/nio/channels/Channel.html#isOpen()) Tells whether or not this channel is open.

**Specified by:**[isOpen](http://docs.google.com/java/nio/channels/Channel.html#isOpen()) in interface [Channel](http://docs.google.com/java/nio/channels/Channel.html) **Returns:**true if, and only if, this channel is open

### begin

protected final void **begin**()

Marks the beginning of an I/O operation that might block indefinitely.

This method should be invoked in tandem with the [end](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#end(boolean)) method, using a try ... finally block as shown [above](#3znysh7), in order to implement asynchronous closing and interruption for this channel.

### end

protected final void **end**(boolean completed)  
 throws [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html)

Marks the end of an I/O operation that might block indefinitely.

This method should be invoked in tandem with the [begin](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#begin()) method, using a try ... finally block as shown [above](#3znysh7), in order to implement asynchronous closing and interruption for this channel.

**Parameters:**completed - true if, and only if, the I/O operation completed successfully, that is, had some effect that would be visible to the operation's invoker **Throws:** [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html) - If the channel was asynchronously closed [ClosedByInterruptException](http://docs.google.com/java/nio/channels/ClosedByInterruptException.html) - If the thread blocked in the I/O operation was interrupted

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AbstractInterruptibleChannel.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   [**NEXT CLASS**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html) | [**FRAMES**](http://docs.google.com/index.html?java/nio/channels/spi/AbstractInterruptibleChannel.html)    [**NO FRAMES**](http://docs.google.com/AbstractInterruptibleChannel.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#2s8eyo1) |

[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).