| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AbstractSelectableChannel.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/java/nio/channels/spi/AbstractInterruptibleChannel.html)   [**NEXT CLASS**](http://docs.google.com/java/nio/channels/spi/AbstractSelectionKey.html) | [**FRAMES**](http://docs.google.com/index.html?java/nio/channels/spi/AbstractSelectableChannel.html)    [**NO FRAMES**](http://docs.google.com/AbstractSelectableChannel.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#3rdcrjn) |

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

Class AbstractSelectableChannel

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

**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:** [DatagramChannel](http://docs.google.com/java/nio/channels/DatagramChannel.html), [Pipe.SinkChannel](http://docs.google.com/java/nio/channels/Pipe.SinkChannel.html), [Pipe.SourceChannel](http://docs.google.com/java/nio/channels/Pipe.SourceChannel.html), [ServerSocketChannel](http://docs.google.com/java/nio/channels/ServerSocketChannel.html), [SocketChannel](http://docs.google.com/java/nio/channels/SocketChannel.html)

public abstract class **AbstractSelectableChannel**extends [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html)

Base implementation class for selectable channels.

This class defines methods that handle the mechanics of channel registration, deregistration, and closing. It maintains the current blocking mode of this channel as well as its current set of selection keys. It performs all of the synchronization required to implement the [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) specification. Implementations of the abstract protected methods defined in this class need not synchronize against other threads that might be engaged in the same operations.

**Since:** 1.4

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

| **Method Summary** | |
| --- | --- |
| [Object](http://docs.google.com/java/lang/Object.html) | [**blockingLock**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#blockingLock())()            Retrieves the object upon which the [configureBlocking](http://docs.google.com/java/nio/channels/SelectableChannel.html#configureBlocking(boolean)) and [register](http://docs.google.com/java/nio/channels/SelectableChannel.html#register(java.nio.channels.Selector,%20int,%20java.lang.Object)) methods synchronize. |
| [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) | [**configureBlocking**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#configureBlocking(boolean))(boolean block)            Adjusts this channel's blocking mode. |
| protected  void | [**implCloseChannel**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#implCloseChannel())()            Closes this channel. |
| protected abstract  void | [**implCloseSelectableChannel**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#implCloseSelectableChannel())()            Closes this selectable channel. |
| protected abstract  void | [**implConfigureBlocking**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#implConfigureBlocking(boolean))(boolean block)            Adjusts this channel's blocking mode. |
| boolean | [**isBlocking**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#isBlocking())()            Tells whether or not every I/O operation on this channel will block until it completes. |
| boolean | [**isRegistered**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#isRegistered())()            Tells whether or not this channel is currently registered with any selectors. |
| [SelectionKey](http://docs.google.com/java/nio/channels/SelectionKey.html) | [**keyFor**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#keyFor(java.nio.channels.Selector))([Selector](http://docs.google.com/java/nio/channels/Selector.html) sel)            Retrieves the key representing the channel's registration with the given selector. |
| [SelectorProvider](http://docs.google.com/java/nio/channels/spi/SelectorProvider.html) | [**provider**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#provider())()            Returns the provider that created this channel. |
| [SelectionKey](http://docs.google.com/java/nio/channels/SelectionKey.html) | [**register**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#register(java.nio.channels.Selector,%20int,%20java.lang.Object))([Selector](http://docs.google.com/java/nio/channels/Selector.html) sel, int ops, [Object](http://docs.google.com/java/lang/Object.html) att)            Registers this channel with the given selector, returning a selection key. |

| **Methods inherited from class java.nio.channels.**[**SelectableChannel**](http://docs.google.com/java/nio/channels/SelectableChannel.html) |
| --- |
| [register](http://docs.google.com/java/nio/channels/SelectableChannel.html#register(java.nio.channels.Selector,%20int)), [validOps](http://docs.google.com/java/nio/channels/SelectableChannel.html#validOps()) |

| **Methods inherited from class java.nio.channels.spi.**[**AbstractInterruptibleChannel**](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html) |
| --- |
| [begin](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#begin()), [close](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#close()), [end](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#end(boolean)), [isOpen](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#isOpen()) |

| **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)) |

| **Methods inherited from interface java.nio.channels.**[**Channel**](http://docs.google.com/java/nio/channels/Channel.html) |
| --- |
| [close](http://docs.google.com/java/nio/channels/Channel.html#close()), [isOpen](http://docs.google.com/java/nio/channels/Channel.html#isOpen()) |

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

### AbstractSelectableChannel

protected **AbstractSelectableChannel**([SelectorProvider](http://docs.google.com/java/nio/channels/spi/SelectorProvider.html) provider)

Initializes a new instance of this class.

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

### provider

public final [SelectorProvider](http://docs.google.com/java/nio/channels/spi/SelectorProvider.html) **provider**()

Returns the provider that created this channel.

**Specified by:**[provider](http://docs.google.com/java/nio/channels/SelectableChannel.html#provider()) in class [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) **Returns:**The provider that created this channel

### isRegistered

public final boolean **isRegistered**()

**Description copied from class:** [**SelectableChannel**](http://docs.google.com/java/nio/channels/SelectableChannel.html#isRegistered()) Tells whether or not this channel is currently registered with any selectors. A newly-created channel is not registered.

Due to the inherent delay between key cancellation and channel deregistration, a channel may remain registered for some time after all of its keys have been cancelled. A channel may also remain registered for some time after it is closed.

**Specified by:**[isRegistered](http://docs.google.com/java/nio/channels/SelectableChannel.html#isRegistered()) in class [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) **Returns:**true if, and only if, this channel is registered

### keyFor

public final [SelectionKey](http://docs.google.com/java/nio/channels/SelectionKey.html) **keyFor**([Selector](http://docs.google.com/java/nio/channels/Selector.html) sel)

**Description copied from class:** [**SelectableChannel**](http://docs.google.com/java/nio/channels/SelectableChannel.html#keyFor(java.nio.channels.Selector)) Retrieves the key representing the channel's registration with the given selector.

**Specified by:**[keyFor](http://docs.google.com/java/nio/channels/SelectableChannel.html#keyFor(java.nio.channels.Selector)) in class [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) **Returns:**The key returned when this channel was last registered with the given selector, or null if this channel is not currently registered with that selector

### register

public final [SelectionKey](http://docs.google.com/java/nio/channels/SelectionKey.html) **register**([Selector](http://docs.google.com/java/nio/channels/Selector.html) sel,  
 int ops,  
 [Object](http://docs.google.com/java/lang/Object.html) att)  
 throws [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html)

Registers this channel with the given selector, returning a selection key.

This method first verifies that this channel is open and that the given initial interest set is valid.

If this channel is already registered with the given selector then the selection key representing that registration is returned after setting its interest set to the given value.

Otherwise this channel has not yet been registered with the given selector, so the [register](http://docs.google.com/java/nio/channels/spi/AbstractSelector.html#register(java.nio.channels.spi.AbstractSelectableChannel,%20int,%20java.lang.Object)) method of the selector is invoked while holding the appropriate locks. The resulting key is added to this channel's key set before being returned.

**Specified by:**[register](http://docs.google.com/java/nio/channels/SelectableChannel.html#register(java.nio.channels.Selector,%20int,%20java.lang.Object)) in class [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) **Parameters:**sel - The selector with which this channel is to be registeredops - The interest set for the resulting keyatt - The attachment for the resulting key; may be null **Returns:**A key representing the registration of this channel with the given selector **Throws:** [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed

### implCloseChannel

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

Closes this channel.

This method, which is specified in the [AbstractInterruptibleChannel](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html) class and is invoked by the [close](http://docs.google.com/java/nio/channels/Channel.html#close()) method, in turn invokes the [implCloseSelectableChannel](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#implCloseSelectableChannel()) method in order to perform the actual work of closing this channel. It then cancels all of this channel's keys.

**Specified by:**[implCloseChannel](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html#implCloseChannel()) in class [AbstractInterruptibleChannel](http://docs.google.com/java/nio/channels/spi/AbstractInterruptibleChannel.html) **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - If an I/O error occurs while closing the channel

### implCloseSelectableChannel

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

Closes this selectable channel.

This method is invoked by the [close](http://docs.google.com/java/nio/channels/Channel.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)

### isBlocking

public final boolean **isBlocking**()

**Description copied from class:** [**SelectableChannel**](http://docs.google.com/java/nio/channels/SelectableChannel.html#isBlocking()) Tells whether or not every I/O operation on this channel will block until it completes. A newly-created channel is always in blocking mode.

If this channel is closed then the value returned by this method is not specified.

**Specified by:**[isBlocking](http://docs.google.com/java/nio/channels/SelectableChannel.html#isBlocking()) in class [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) **Returns:**true if, and only if, this channel is in blocking mode

### blockingLock

public final [Object](http://docs.google.com/java/lang/Object.html) **blockingLock**()

**Description copied from class:** [**SelectableChannel**](http://docs.google.com/java/nio/channels/SelectableChannel.html#blockingLock()) Retrieves the object upon which the [configureBlocking](http://docs.google.com/java/nio/channels/SelectableChannel.html#configureBlocking(boolean)) and [register](http://docs.google.com/java/nio/channels/SelectableChannel.html#register(java.nio.channels.Selector,%20int,%20java.lang.Object)) methods synchronize. This is often useful in the implementation of adaptors that require a specific blocking mode to be maintained for a short period of time.

**Specified by:**[blockingLock](http://docs.google.com/java/nio/channels/SelectableChannel.html#blockingLock()) in class [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) **Returns:**The blocking-mode lock object

### configureBlocking

public final [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) **configureBlocking**(boolean block)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Adjusts this channel's blocking mode.

If the given blocking mode is different from the current blocking mode then this method invokes the [implConfigureBlocking](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#implConfigureBlocking(boolean)) method, while holding the appropriate locks, in order to change the mode.

**Specified by:**[configureBlocking](http://docs.google.com/java/nio/channels/SelectableChannel.html#configureBlocking(boolean)) in class [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) **Parameters:**block - If true then this channel will be placed in blocking mode; if false then it will be placed non-blocking mode **Returns:**This selectable channel **Throws:** [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed [IOException](http://docs.google.com/java/io/IOException.html) - If an I/O error occurs

### implConfigureBlocking

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

Adjusts this channel's blocking mode.

This method is invoked by the [configureBlocking](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#configureBlocking(boolean)) method in order to perform the actual work of changing the blocking mode. This method is only invoked if the new mode is different from the current mode.

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

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AbstractSelectableChannel.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/java/nio/channels/spi/AbstractInterruptibleChannel.html)   [**NEXT CLASS**](http://docs.google.com/java/nio/channels/spi/AbstractSelectionKey.html) | [**FRAMES**](http://docs.google.com/index.html?java/nio/channels/spi/AbstractSelectableChannel.html)    [**NO FRAMES**](http://docs.google.com/AbstractSelectableChannel.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#3rdcrjn) |

[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).