| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DatagramChannel.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/ConnectionPendingException.html)   [**NEXT CLASS**](http://docs.google.com/java/nio/channels/FileChannel.html) | [**FRAMES**](http://docs.google.com/index.html?java/nio/channels/DatagramChannel.html)    [**NO FRAMES**](http://docs.google.com/DatagramChannel.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#17dp8vu) | [METHOD](#26in1rg) |

## **java.nio.channels**

Class DatagramChannel

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

**All Implemented Interfaces:** [Closeable](http://docs.google.com/java/io/Closeable.html), [ByteChannel](http://docs.google.com/java/nio/channels/ByteChannel.html), [Channel](http://docs.google.com/java/nio/channels/Channel.html), [GatheringByteChannel](http://docs.google.com/java/nio/channels/GatheringByteChannel.html), [InterruptibleChannel](http://docs.google.com/java/nio/channels/InterruptibleChannel.html), [ReadableByteChannel](http://docs.google.com/java/nio/channels/ReadableByteChannel.html), [ScatteringByteChannel](http://docs.google.com/java/nio/channels/ScatteringByteChannel.html), [WritableByteChannel](http://docs.google.com/java/nio/channels/WritableByteChannel.html)

public abstract class **DatagramChannel**extends [AbstractSelectableChannel](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html)implements [ByteChannel](http://docs.google.com/java/nio/channels/ByteChannel.html), [ScatteringByteChannel](http://docs.google.com/java/nio/channels/ScatteringByteChannel.html), [GatheringByteChannel](http://docs.google.com/java/nio/channels/GatheringByteChannel.html)

A selectable channel for datagram-oriented sockets.

Datagram channels are not a complete abstraction of network datagram sockets. Binding and the manipulation of socket options must be done through an associated [DatagramSocket](http://docs.google.com/java/net/DatagramSocket.html) object obtained by invoking the [socket](http://docs.google.com/java/nio/channels/DatagramChannel.html#socket()) method. It is not possible to create a channel for an arbitrary, pre-existing datagram socket, nor is it possible to specify the [DatagramSocketImpl](http://docs.google.com/java/net/DatagramSocketImpl.html) object to be used by a datagram socket associated with a datagram channel.

A datagram channel is created by invoking the [open](http://docs.google.com/java/nio/channels/DatagramChannel.html#open()) method of this class. A newly-created datagram channel is open but not connected. A datagram channel need not be connected in order for the [send](http://docs.google.com/java/nio/channels/DatagramChannel.html#send(java.nio.ByteBuffer,%20java.net.SocketAddress)) and [receive](http://docs.google.com/java/nio/channels/DatagramChannel.html#receive(java.nio.ByteBuffer)) methods to be used. A datagram channel may be connected, by invoking its [connect](http://docs.google.com/java/nio/channels/DatagramChannel.html#connect(java.net.SocketAddress)) method, in order to avoid the overhead of the security checks are otherwise performed as part of every send and receive operation. A datagram channel must be connected in order to use the [read](http://docs.google.com/java/nio/channels/DatagramChannel.html#read(java.nio.ByteBuffer)) and [write](http://docs.google.com/java/nio/channels/DatagramChannel.html#write(java.nio.ByteBuffer)) methods, since those methods do not accept or return socket addresses.

Once connected, a datagram channel remains connected until it is disconnected or closed. Whether or not a datagram channel is connected may be determined by invoking its [isConnected](http://docs.google.com/java/nio/channels/DatagramChannel.html#isConnected()) method.

Datagram channels are safe for use by multiple concurrent threads. They support concurrent reading and writing, though at most one thread may be reading and at most one thread may be writing at any given time.

**Since:** 1.4

| **Constructor Summary** | |
| --- | --- |
| protected | [**DatagramChannel**](http://docs.google.com/java/nio/channels/DatagramChannel.html#DatagramChannel(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** | |
| --- | --- |
| abstract  [DatagramChannel](http://docs.google.com/java/nio/channels/DatagramChannel.html) | [**connect**](http://docs.google.com/java/nio/channels/DatagramChannel.html#connect(java.net.SocketAddress))([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) remote)            Connects this channel's socket. |
| abstract  [DatagramChannel](http://docs.google.com/java/nio/channels/DatagramChannel.html) | [**disconnect**](http://docs.google.com/java/nio/channels/DatagramChannel.html#disconnect())()            Disconnects this channel's socket. |
| abstract  boolean | [**isConnected**](http://docs.google.com/java/nio/channels/DatagramChannel.html#isConnected())()            Tells whether or not this channel's socket is connected. |
| static [DatagramChannel](http://docs.google.com/java/nio/channels/DatagramChannel.html) | [**open**](http://docs.google.com/java/nio/channels/DatagramChannel.html#open())()            Opens a datagram channel. |
| abstract  int | [**read**](http://docs.google.com/java/nio/channels/DatagramChannel.html#read(java.nio.ByteBuffer))([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html) dst)            Reads a datagram from this channel. |
| long | [**read**](http://docs.google.com/java/nio/channels/DatagramChannel.html#read(java.nio.ByteBuffer%5B%5D))([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html)[] dsts)            Reads a datagram from this channel. |
| abstract  long | [**read**](http://docs.google.com/java/nio/channels/DatagramChannel.html#read(java.nio.ByteBuffer%5B%5D,%20int,%20int))([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html)[] dsts, int offset, int length)            Reads a datagram from this channel. |
| abstract  [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) | [**receive**](http://docs.google.com/java/nio/channels/DatagramChannel.html#receive(java.nio.ByteBuffer))([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html) dst)            Receives a datagram via this channel. |
| abstract  int | [**send**](http://docs.google.com/java/nio/channels/DatagramChannel.html#send(java.nio.ByteBuffer,%20java.net.SocketAddress))([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html) src, [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) target)            Sends a datagram via this channel. |
| abstract  [DatagramSocket](http://docs.google.com/java/net/DatagramSocket.html) | [**socket**](http://docs.google.com/java/nio/channels/DatagramChannel.html#socket())()            Retrieves a datagram socket associated with this channel. |
| int | [**validOps**](http://docs.google.com/java/nio/channels/DatagramChannel.html#validOps())()            Returns an operation set identifying this channel's supported operations. |
| abstract  int | [**write**](http://docs.google.com/java/nio/channels/DatagramChannel.html#write(java.nio.ByteBuffer))([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html) src)            Writes a datagram to this channel. |
| long | [**write**](http://docs.google.com/java/nio/channels/DatagramChannel.html#write(java.nio.ByteBuffer%5B%5D))([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html)[] srcs)            Writes a datagram to this channel. |
| abstract  long | [**write**](http://docs.google.com/java/nio/channels/DatagramChannel.html#write(java.nio.ByteBuffer%5B%5D,%20int,%20int))([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html)[] srcs, int offset, int length)            Writes a datagram to this channel. |

| **Methods inherited from class java.nio.channels.spi.**[**AbstractSelectableChannel**](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html) |
| --- |
| [blockingLock](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#blockingLock()), [configureBlocking](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#configureBlocking(boolean)), [implCloseChannel](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#implCloseChannel()), [implCloseSelectableChannel](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#implCloseSelectableChannel()), [implConfigureBlocking](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#implConfigureBlocking(boolean)), [isBlocking](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#isBlocking()), [isRegistered](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#isRegistered()), [keyFor](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#keyFor(java.nio.channels.Selector)), [provider](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#provider()), [register](http://docs.google.com/java/nio/channels/spi/AbstractSelectableChannel.html#register(java.nio.channels.Selector,%20int,%20java.lang.Object)) |

| **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)) |

| **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** |
| --- |

### DatagramChannel

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

Initializes a new instance of this class.

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

### open

public static [DatagramChannel](http://docs.google.com/java/nio/channels/DatagramChannel.html) **open**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Opens a datagram channel.

The new channel is created by invoking the [openDatagramChannel](http://docs.google.com/java/nio/channels/spi/SelectorProvider.html#openDatagramChannel()) method of the system-wide default [SelectorProvider](http://docs.google.com/java/nio/channels/spi/SelectorProvider.html) object. The channel will not be connected.

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

### validOps

public final int **validOps**()

Returns an operation set identifying this channel's supported operations.

Datagram channels support reading and writing, so this method returns ([SelectionKey.OP\_READ](http://docs.google.com/java/nio/channels/SelectionKey.html#OP_READ) | [SelectionKey.OP\_WRITE](http://docs.google.com/java/nio/channels/SelectionKey.html#OP_WRITE)).

**Specified by:**[validOps](http://docs.google.com/java/nio/channels/SelectableChannel.html#validOps()) in class [SelectableChannel](http://docs.google.com/java/nio/channels/SelectableChannel.html) **Returns:**The valid-operation set

### socket

public abstract [DatagramSocket](http://docs.google.com/java/net/DatagramSocket.html) **socket**()

Retrieves a datagram socket associated with this channel.

The returned object will not declare any public methods that are not declared in the [DatagramSocket](http://docs.google.com/java/net/DatagramSocket.html) class.

**Returns:**A datagram socket associated with this channel

### isConnected

public abstract boolean **isConnected**()

Tells whether or not this channel's socket is connected.

**Returns:**true if, and only if, this channel's socket is connected

### connect

public abstract [DatagramChannel](http://docs.google.com/java/nio/channels/DatagramChannel.html) **connect**([SocketAddress](http://docs.google.com/java/net/SocketAddress.html) remote)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Connects this channel's socket.

The channel's socket is configured so that it only receives datagrams from, and sends datagrams to, the given remote *peer* address. Once connected, datagrams may not be received from or sent to any other address. A datagram socket remains connected until it is explicitly disconnected or until it is closed.

This method performs exactly the same security checks as the [connect](http://docs.google.com/java/net/DatagramSocket.html#connect(java.net.InetAddress,%20int)) method of the [DatagramSocket](http://docs.google.com/java/net/DatagramSocket.html) class. That is, if a security manager has been installed then this method verifies that its [checkAccept](http://docs.google.com/java/lang/SecurityManager.html#checkAccept(java.lang.String,%20int)) and [checkConnect](http://docs.google.com/java/lang/SecurityManager.html#checkConnect(java.lang.String,%20int)) methods permit datagrams to be received from and sent to, respectively, the given remote address.

This method may be invoked at any time. It will not have any effect on read or write operations that are already in progress at the moment that it is invoked.

**Parameters:**remote - The remote address to which this channel is to be connected **Returns:**This datagram channel **Throws:** [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html) - If another thread closes this channel while the connect operation is in progress [ClosedByInterruptException](http://docs.google.com/java/nio/channels/ClosedByInterruptException.html) - If another thread interrupts the current thread while the connect operation is in progress, thereby closing the channel and setting the current thread's interrupt status [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - If a security manager has been installed and it does not permit access to the given remote address [IOException](http://docs.google.com/java/io/IOException.html) - If some other I/O error occurs

### disconnect

public abstract [DatagramChannel](http://docs.google.com/java/nio/channels/DatagramChannel.html) **disconnect**()  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Disconnects this channel's socket.

The channel's socket is configured so that it can receive datagrams from, and sends datagrams to, any remote address so long as the security manager, if installed, permits it.

This method may be invoked at any time. It will not have any effect on read or write operations that are already in progress at the moment that it is invoked.

If this channel's socket is not connected, or if the channel is closed, then invoking this method has no effect.

**Returns:**This datagram channel **Throws:** [IOException](http://docs.google.com/java/io/IOException.html) - If some other I/O error occurs

### receive

public abstract [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) **receive**([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html) dst)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Receives a datagram via this channel.

If a datagram is immediately available, or if this channel is in blocking mode and one eventually becomes available, then the datagram is copied into the given byte buffer and its source address is returned. If this channel is in non-blocking mode and a datagram is not immediately available then this method immediately returns null.

The datagram is transferred into the given byte buffer starting at its current position, as if by a regular [read](http://docs.google.com/java/nio/channels/ReadableByteChannel.html#read(java.nio.ByteBuffer)) operation. If there are fewer bytes remaining in the buffer than are required to hold the datagram then the remainder of the datagram is silently discarded.

This method performs exactly the same security checks as the [receive](http://docs.google.com/java/net/DatagramSocket.html#receive(java.net.DatagramPacket)) method of the [DatagramSocket](http://docs.google.com/java/net/DatagramSocket.html) class. That is, if the socket is not connected to a specific remote address and a security manager has been installed then for each datagram received this method verifies that the source's address and port number are permitted by the security manager's [checkAccept](http://docs.google.com/java/lang/SecurityManager.html#checkAccept(java.lang.String,%20int)) method. The overhead of this security check can be avoided by first connecting the socket via the [connect](http://docs.google.com/java/nio/channels/DatagramChannel.html#connect(java.net.SocketAddress)) method.

This method may be invoked at any time. If another thread has already initiated a read operation upon this channel, however, then an invocation of this method will block until the first operation is complete.

**Parameters:**dst - The buffer into which the datagram is to be transferred **Returns:**The datagram's source address, or null if this channel is in non-blocking mode and no datagram was immediately available **Throws:** [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html) - If another thread closes this channel while the read operation is in progress [ClosedByInterruptException](http://docs.google.com/java/nio/channels/ClosedByInterruptException.html) - If another thread interrupts the current thread while the read operation is in progress, thereby closing the channel and setting the current thread's interrupt status [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - If a security manager has been installed and it does not permit datagrams to be accepted from the datagram's sender [IOException](http://docs.google.com/java/io/IOException.html) - If some other I/O error occurs

### send

public abstract int **send**([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html) src,  
 [SocketAddress](http://docs.google.com/java/net/SocketAddress.html) target)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Sends a datagram via this channel.

If this channel is in non-blocking mode and there is sufficient room in the underlying output buffer, or if this channel is in blocking mode and sufficient room becomes available, then the remaining bytes in the given buffer are transmitted as a single datagram to the given target address.

The datagram is transferred from the byte buffer as if by a regular [write](http://docs.google.com/java/nio/channels/WritableByteChannel.html#write(java.nio.ByteBuffer)) operation.

This method performs exactly the same security checks as the [send](http://docs.google.com/java/net/DatagramSocket.html#send(java.net.DatagramPacket)) method of the [DatagramSocket](http://docs.google.com/java/net/DatagramSocket.html) class. That is, if the socket is not connected to a specific remote address and a security manager has been installed then for each datagram sent this method verifies that the target address and port number are permitted by the security manager's [checkConnect](http://docs.google.com/java/lang/SecurityManager.html#checkConnect(java.lang.String,%20int)) method. The overhead of this security check can be avoided by first connecting the socket via the [connect](http://docs.google.com/java/nio/channels/DatagramChannel.html#connect(java.net.SocketAddress)) method.

This method may be invoked at any time. If another thread has already initiated a write operation upon this channel, however, then an invocation of this method will block until the first operation is complete.

**Parameters:**src - The buffer containing the datagram to be senttarget - The address to which the datagram is to be sent **Returns:**The number of bytes sent, which will be either the number of bytes that were remaining in the source buffer when this method was invoked or, if this channel is non-blocking, may be zero if there was insufficient room for the datagram in the underlying output buffer **Throws:** [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html) - If another thread closes this channel while the read operation is in progress [ClosedByInterruptException](http://docs.google.com/java/nio/channels/ClosedByInterruptException.html) - If another thread interrupts the current thread while the read operation is in progress, thereby closing the channel and setting the current thread's interrupt status [SecurityException](http://docs.google.com/java/lang/SecurityException.html) - If a security manager has been installed and it does not permit datagrams to be sent to the given address [IOException](http://docs.google.com/java/io/IOException.html) - If some other I/O error occurs

### read

public abstract int **read**([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html) dst)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Reads a datagram from this channel.

This method may only be invoked if this channel's socket is connected, and it only accepts datagrams from the socket's peer. If there are more bytes in the datagram than remain in the given buffer then the remainder of the datagram is silently discarded. Otherwise this method behaves exactly as specified in the [ReadableByteChannel](http://docs.google.com/java/nio/channels/ReadableByteChannel.html) interface.

**Specified by:**[read](http://docs.google.com/java/nio/channels/ReadableByteChannel.html#read(java.nio.ByteBuffer)) in interface [ReadableByteChannel](http://docs.google.com/java/nio/channels/ReadableByteChannel.html) **Parameters:**dst - The buffer into which bytes are to be transferred **Returns:**The number of bytes read, possibly zero, or -1 if the channel has reached end-of-stream **Throws:** [NotYetConnectedException](http://docs.google.com/java/nio/channels/NotYetConnectedException.html) - If this channel's socket is not connected [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html) - If another thread closes this channel while the read operation is in progress [ClosedByInterruptException](http://docs.google.com/java/nio/channels/ClosedByInterruptException.html) - If another thread interrupts the current thread while the read operation is in progress, thereby closing the channel and setting the current thread's interrupt status [IOException](http://docs.google.com/java/io/IOException.html) - If some other I/O error occurs

### read

public abstract long **read**([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html)[] dsts,  
 int offset,  
 int length)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Reads a datagram from this channel.

This method may only be invoked if this channel's socket is connected, and it only accepts datagrams from the socket's peer. If there are more bytes in the datagram than remain in the given buffers then the remainder of the datagram is silently discarded. Otherwise this method behaves exactly as specified in the [ScatteringByteChannel](http://docs.google.com/java/nio/channels/ScatteringByteChannel.html) interface.

**Specified by:**[read](http://docs.google.com/java/nio/channels/ScatteringByteChannel.html#read(java.nio.ByteBuffer%5B%5D,%20int,%20int)) in interface [ScatteringByteChannel](http://docs.google.com/java/nio/channels/ScatteringByteChannel.html) **Parameters:**dsts - The buffers into which bytes are to be transferredoffset - The offset within the buffer array of the first buffer into which bytes are to be transferred; must be non-negative and no larger than dsts.lengthlength - The maximum number of buffers to be accessed; must be non-negative and no larger than dsts.length - offset **Returns:**The number of bytes read, possibly zero, or -1 if the channel has reached end-of-stream **Throws:** [NotYetConnectedException](http://docs.google.com/java/nio/channels/NotYetConnectedException.html) - If this channel's socket is not connected [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html) - If another thread closes this channel while the read operation is in progress [ClosedByInterruptException](http://docs.google.com/java/nio/channels/ClosedByInterruptException.html) - If another thread interrupts the current thread while the read operation is in progress, thereby closing the channel and setting the current thread's interrupt status [IOException](http://docs.google.com/java/io/IOException.html) - If some other I/O error occurs

### read

public final long **read**([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html)[] dsts)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Reads a datagram from this channel.

This method may only be invoked if this channel's socket is connected, and it only accepts datagrams from the socket's peer. If there are more bytes in the datagram than remain in the given buffers then the remainder of the datagram is silently discarded. Otherwise this method behaves exactly as specified in the [ScatteringByteChannel](http://docs.google.com/java/nio/channels/ScatteringByteChannel.html) interface.

**Specified by:**[read](http://docs.google.com/java/nio/channels/ScatteringByteChannel.html#read(java.nio.ByteBuffer%5B%5D)) in interface [ScatteringByteChannel](http://docs.google.com/java/nio/channels/ScatteringByteChannel.html) **Parameters:**dsts - The buffers into which bytes are to be transferred **Returns:**The number of bytes read, possibly zero, or -1 if the channel has reached end-of-stream **Throws:** [NotYetConnectedException](http://docs.google.com/java/nio/channels/NotYetConnectedException.html) - If this channel's socket is not connected [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html) - If another thread closes this channel while the read operation is in progress [ClosedByInterruptException](http://docs.google.com/java/nio/channels/ClosedByInterruptException.html) - If another thread interrupts the current thread while the read operation is in progress, thereby closing the channel and setting the current thread's interrupt status [IOException](http://docs.google.com/java/io/IOException.html) - If some other I/O error occurs

### write

public abstract int **write**([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html) src)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a datagram to this channel.

This method may only be invoked if this channel's socket is connected, in which case it sends datagrams directly to the socket's peer. Otherwise it behaves exactly as specified in the [WritableByteChannel](http://docs.google.com/java/nio/channels/WritableByteChannel.html) interface.

**Specified by:**[write](http://docs.google.com/java/nio/channels/WritableByteChannel.html#write(java.nio.ByteBuffer)) in interface [WritableByteChannel](http://docs.google.com/java/nio/channels/WritableByteChannel.html) **Parameters:**src - The buffer from which bytes are to be retrieved **Returns:**The number of bytes written, possibly zero **Throws:** [NotYetConnectedException](http://docs.google.com/java/nio/channels/NotYetConnectedException.html) - If this channel's socket is not connected [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html) - If another thread closes this channel while the write operation is in progress [ClosedByInterruptException](http://docs.google.com/java/nio/channels/ClosedByInterruptException.html) - If another thread interrupts the current thread while the write operation is in progress, thereby closing the channel and setting the current thread's interrupt status [IOException](http://docs.google.com/java/io/IOException.html) - If some other I/O error occurs

### write

public abstract long **write**([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html)[] srcs,  
 int offset,  
 int length)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a datagram to this channel.

This method may only be invoked if this channel's socket is connected, in which case it sends datagrams directly to the socket's peer. Otherwise it behaves exactly as specified in the [GatheringByteChannel](http://docs.google.com/java/nio/channels/GatheringByteChannel.html) interface.

**Specified by:**[write](http://docs.google.com/java/nio/channels/GatheringByteChannel.html#write(java.nio.ByteBuffer%5B%5D,%20int,%20int)) in interface [GatheringByteChannel](http://docs.google.com/java/nio/channels/GatheringByteChannel.html) **Parameters:**srcs - The buffers from which bytes are to be retrievedoffset - The offset within the buffer array of the first buffer from which bytes are to be retrieved; must be non-negative and no larger than srcs.lengthlength - The maximum number of buffers to be accessed; must be non-negative and no larger than srcs.length - offset **Returns:**The number of bytes sent, which will be either the number of bytes that were remaining in the source buffer when this method was invoked or, if this channel is non-blocking, may be zero if there was insufficient room for the datagram in the underlying output buffer **Throws:** [NotYetConnectedException](http://docs.google.com/java/nio/channels/NotYetConnectedException.html) - If this channel's socket is not connected [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html) - If another thread closes this channel while the write operation is in progress [ClosedByInterruptException](http://docs.google.com/java/nio/channels/ClosedByInterruptException.html) - If another thread interrupts the current thread while the write operation is in progress, thereby closing the channel and setting the current thread's interrupt status [IOException](http://docs.google.com/java/io/IOException.html) - If some other I/O error occurs

### write

public final long **write**([ByteBuffer](http://docs.google.com/java/nio/ByteBuffer.html)[] srcs)  
 throws [IOException](http://docs.google.com/java/io/IOException.html)

Writes a datagram to this channel.

This method may only be invoked if this channel's socket is connected, in which case it sends datagrams directly to the socket's peer. Otherwise it behaves exactly as specified in the [GatheringByteChannel](http://docs.google.com/java/nio/channels/GatheringByteChannel.html) interface.

**Specified by:**[write](http://docs.google.com/java/nio/channels/GatheringByteChannel.html#write(java.nio.ByteBuffer%5B%5D)) in interface [GatheringByteChannel](http://docs.google.com/java/nio/channels/GatheringByteChannel.html) **Parameters:**srcs - The buffers from which bytes are to be retrieved **Returns:**The number of bytes sent, which will be either the number of bytes that were remaining in the source buffer when this method was invoked or, if this channel is non-blocking, may be zero if there was insufficient room for the datagram in the underlying output buffer **Throws:** [NotYetConnectedException](http://docs.google.com/java/nio/channels/NotYetConnectedException.html) - If this channel's socket is not connected [ClosedChannelException](http://docs.google.com/java/nio/channels/ClosedChannelException.html) - If this channel is closed [AsynchronousCloseException](http://docs.google.com/java/nio/channels/AsynchronousCloseException.html) - If another thread closes this channel while the write operation is in progress [ClosedByInterruptException](http://docs.google.com/java/nio/channels/ClosedByInterruptException.html) - If another thread interrupts the current thread while the write operation is in progress, thereby closing the channel and setting the current thread's interrupt status [IOException](http://docs.google.com/java/io/IOException.html) - If some other 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/DatagramChannel.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/ConnectionPendingException.html)   [**NEXT CLASS**](http://docs.google.com/java/nio/channels/FileChannel.html) | [**FRAMES**](http://docs.google.com/index.html?java/nio/channels/DatagramChannel.html)    [**NO FRAMES**](http://docs.google.com/DatagramChannel.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#17dp8vu) | [METHOD](#26in1rg) |

[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).