| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/SystemFlavorMap.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/awt/datatransfer/StringSelection.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/datatransfer/Transferable.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/datatransfer/SystemFlavorMap.html)    [**NO FRAMES**](http://docs.google.com/SystemFlavorMap.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#tyjcwt) |

## **java.awt.datatransfer**

Class SystemFlavorMap

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **java.awt.datatransfer.SystemFlavorMap**

**All Implemented Interfaces:** [FlavorMap](http://docs.google.com/java/awt/datatransfer/FlavorMap.html), [FlavorTable](http://docs.google.com/java/awt/datatransfer/FlavorTable.html)

public final class **SystemFlavorMap**extends [Object](http://docs.google.com/java/lang/Object.html)implements [FlavorMap](http://docs.google.com/java/awt/datatransfer/FlavorMap.html), [FlavorTable](http://docs.google.com/java/awt/datatransfer/FlavorTable.html)

The SystemFlavorMap is a configurable map between "natives" (Strings), which correspond to platform-specific data formats, and "flavors" (DataFlavors), which correspond to platform-independent MIME types. This mapping is used by the data transfer subsystem to transfer data between Java and native applications, and between Java applications in separate VMs.

In the Sun reference implementation, the default SystemFlavorMap is initialized by the file jre/lib/flavormap.properties and the contents of the URL referenced by the AWT property AWT.DnD.flavorMapFileURL. See flavormap.properties for details.

**Since:** 1.2

| **Method Summary** | |
| --- | --- |
| void | [**addFlavorForUnencodedNative**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#addFlavorForUnencodedNative(java.lang.String,%20java.awt.datatransfer.DataFlavor))([String](http://docs.google.com/java/lang/String.html) nat, [DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) flav)            Adds a mapping from a single String native to a single DataFlavor. |
| void | [**addUnencodedNativeForFlavor**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#addUnencodedNativeForFlavor(java.awt.datatransfer.DataFlavor,%20java.lang.String))([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) flav, [String](http://docs.google.com/java/lang/String.html) nat)            Adds a mapping from the specified DataFlavor (and all DataFlavors equal to the specified DataFlavor) to the specified String native. |
| static [DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) | [**decodeDataFlavor**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#decodeDataFlavor(java.lang.String))([String](http://docs.google.com/java/lang/String.html) nat)            Decodes a String native for use as a DataFlavor. |
| static [String](http://docs.google.com/java/lang/String.html) | [**decodeJavaMIMEType**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#decodeJavaMIMEType(java.lang.String))([String](http://docs.google.com/java/lang/String.html) nat)            Decodes a String native for use as a Java MIME type. |
| static [String](http://docs.google.com/java/lang/String.html) | [**encodeDataFlavor**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#encodeDataFlavor(java.awt.datatransfer.DataFlavor))([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) flav)            Encodes a DataFlavor for use as a String native. |
| static [String](http://docs.google.com/java/lang/String.html) | [**encodeJavaMIMEType**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#encodeJavaMIMEType(java.lang.String))([String](http://docs.google.com/java/lang/String.html) mimeType)            Encodes a MIME type for use as a String native. |
| static [FlavorMap](http://docs.google.com/java/awt/datatransfer/FlavorMap.html) | [**getDefaultFlavorMap**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#getDefaultFlavorMap())()            Returns the default FlavorMap for this thread's ClassLoader. |
| [List](http://docs.google.com/java/util/List.html)<[DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)> | [**getFlavorsForNative**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#getFlavorsForNative(java.lang.String))([String](http://docs.google.com/java/lang/String.html) nat)            Returns a List of DataFlavors to which the specified String native can be translated by the data transfer subsystem. |
| [Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)> | [**getFlavorsForNatives**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#getFlavorsForNatives(java.lang.String%5B%5D))([String](http://docs.google.com/java/lang/String.html)[] natives)            Returns a Map of the specified String natives to their most preferred DataFlavor. |
| [List](http://docs.google.com/java/util/List.html)<[String](http://docs.google.com/java/lang/String.html)> | [**getNativesForFlavor**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#getNativesForFlavor(java.awt.datatransfer.DataFlavor))([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) flav)            Returns a List of String natives to which the specified DataFlavor can be translated by the data transfer subsystem. |
| [Map](http://docs.google.com/java/util/Map.html)<[DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html),[String](http://docs.google.com/java/lang/String.html)> | [**getNativesForFlavors**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#getNativesForFlavors(java.awt.datatransfer.DataFlavor%5B%5D))([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)[] flavors)            Returns a Map of the specified DataFlavors to their most preferred String native. |
| static boolean | [**isJavaMIMEType**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#isJavaMIMEType(java.lang.String))([String](http://docs.google.com/java/lang/String.html) str)            Returns whether the specified String is an encoded Java MIME type. |
| void | [**setFlavorsForNative**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#setFlavorsForNative(java.lang.String,%20java.awt.datatransfer.DataFlavor%5B%5D))([String](http://docs.google.com/java/lang/String.html) nat, [DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)[] flavors)            Discards the current mappings for the specified String native, and creates new mappings to the specified DataFlavors. |
| void | [**setNativesForFlavor**](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#setNativesForFlavor(java.awt.datatransfer.DataFlavor,%20java.lang.String%5B%5D))([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) flav, [String](http://docs.google.com/java/lang/String.html)[] natives)            Discards the current mappings for the specified DataFlavor and all DataFlavors equal to the specified DataFlavor, and creates new mappings to the specified String natives. |

| **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)) |

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

### getDefaultFlavorMap

public static [FlavorMap](http://docs.google.com/java/awt/datatransfer/FlavorMap.html) **getDefaultFlavorMap**()

Returns the default FlavorMap for this thread's ClassLoader.

### getNativesForFlavor

public [List](http://docs.google.com/java/util/List.html)<[String](http://docs.google.com/java/lang/String.html)> **getNativesForFlavor**([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) flav)

Returns a List of String natives to which the specified DataFlavor can be translated by the data transfer subsystem. The List will be sorted from best native to worst. That is, the first native will best reflect data in the specified flavor to the underlying native platform.

If the specified DataFlavor is previously unknown to the data transfer subsystem and the data transfer subsystem is unable to translate this DataFlavor to any existing native, then invoking this method will establish a mapping in both directions between the specified DataFlavor and an encoded version of its MIME type as its native.

**Specified by:**[getNativesForFlavor](http://docs.google.com/java/awt/datatransfer/FlavorTable.html#getNativesForFlavor(java.awt.datatransfer.DataFlavor)) in interface [FlavorTable](http://docs.google.com/java/awt/datatransfer/FlavorTable.html) **Parameters:**flav - the DataFlavor whose corresponding natives should be returned. If null is specified, all natives currently known to the data transfer subsystem are returned in a non-deterministic order. **Returns:**a java.util.List of java.lang.String objects which are platform-specific representations of platform- specific data formats**Since:** 1.4 **See Also:**[encodeDataFlavor(java.awt.datatransfer.DataFlavor)](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#encodeDataFlavor(java.awt.datatransfer.DataFlavor))

### getFlavorsForNative

public [List](http://docs.google.com/java/util/List.html)<[DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)> **getFlavorsForNative**([String](http://docs.google.com/java/lang/String.html) nat)

Returns a List of DataFlavors to which the specified String native can be translated by the data transfer subsystem. The List will be sorted from best DataFlavor to worst. That is, the first DataFlavor will best reflect data in the specified native to a Java application.

If the specified native is previously unknown to the data transfer subsystem, and that native has been properly encoded, then invoking this method will establish a mapping in both directions between the specified native and a DataFlavor whose MIME type is a decoded version of the native.

If the specified native is not a properly encoded native and the mappings for this native have not been altered with setFlavorsForNative, then the contents of the List is platform dependent, but null cannot be returned.

**Specified by:**[getFlavorsForNative](http://docs.google.com/java/awt/datatransfer/FlavorTable.html#getFlavorsForNative(java.lang.String)) in interface [FlavorTable](http://docs.google.com/java/awt/datatransfer/FlavorTable.html) **Parameters:**nat - the native whose corresponding DataFlavors should be returned. If null is specified, all DataFlavors currently known to the data transfer subsystem are returned in a non-deterministic order. **Returns:**a java.util.List of DataFlavor objects into which platform-specific data in the specified, platform-specific native can be translated**Since:** 1.4 **See Also:**[encodeJavaMIMEType(java.lang.String)](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#encodeJavaMIMEType(java.lang.String))

### getNativesForFlavors

public [Map](http://docs.google.com/java/util/Map.html)<[DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html),[String](http://docs.google.com/java/lang/String.html)> **getNativesForFlavors**([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)[] flavors)

Returns a Map of the specified DataFlavors to their most preferred String native. Each native value will be the same as the first native in the List returned by getNativesForFlavor for the specified flavor.

If a specified DataFlavor is previously unknown to the data transfer subsystem, then invoking this method will establish a mapping in both directions between the specified DataFlavor and an encoded version of its MIME type as its native.

**Specified by:**[getNativesForFlavors](http://docs.google.com/java/awt/datatransfer/FlavorMap.html#getNativesForFlavors(java.awt.datatransfer.DataFlavor%5B%5D)) in interface [FlavorMap](http://docs.google.com/java/awt/datatransfer/FlavorMap.html) **Parameters:**flavors - an array of DataFlavors which will be the key set of the returned Map. If null is specified, a mapping of all DataFlavors known to the data transfer subsystem to their most preferred String natives will be returned. **Returns:**a java.util.Map of DataFlavors to String natives**See Also:**[getNativesForFlavor(java.awt.datatransfer.DataFlavor)](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#getNativesForFlavor(java.awt.datatransfer.DataFlavor)), [encodeDataFlavor(java.awt.datatransfer.DataFlavor)](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#encodeDataFlavor(java.awt.datatransfer.DataFlavor))

### getFlavorsForNatives

public [Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)> **getFlavorsForNatives**([String](http://docs.google.com/java/lang/String.html)[] natives)

Returns a Map of the specified String natives to their most preferred DataFlavor. Each DataFlavor value will be the same as the first DataFlavor in the List returned by getFlavorsForNative for the specified native.

If a specified native is previously unknown to the data transfer subsystem, and that native has been properly encoded, then invoking this method will establish a mapping in both directions between the specified native and a DataFlavor whose MIME type is a decoded version of the native.

**Specified by:**[getFlavorsForNatives](http://docs.google.com/java/awt/datatransfer/FlavorMap.html#getFlavorsForNatives(java.lang.String%5B%5D)) in interface [FlavorMap](http://docs.google.com/java/awt/datatransfer/FlavorMap.html) **Parameters:**natives - an array of Strings which will be the key set of the returned Map. If null is specified, a mapping of all supported String natives to their most preferred DataFlavors will be returned. **Returns:**a java.util.Map of String natives to DataFlavors**See Also:**[getFlavorsForNative(java.lang.String)](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#getFlavorsForNative(java.lang.String)), [encodeJavaMIMEType(java.lang.String)](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#encodeJavaMIMEType(java.lang.String))

### addUnencodedNativeForFlavor

public void **addUnencodedNativeForFlavor**([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) flav,  
 [String](http://docs.google.com/java/lang/String.html) nat)

Adds a mapping from the specified DataFlavor (and all DataFlavors equal to the specified DataFlavor) to the specified String native. Unlike getNativesForFlavor, the mapping will only be established in one direction, and the native will not be encoded. To establish a two-way mapping, call addFlavorForUnencodedNative as well. The new mapping will be of lower priority than any existing mapping. This method has no effect if a mapping from the specified or equal DataFlavor to the specified String native already exists.

**Parameters:**flav - the DataFlavor key for the mappingnat - the String native value for the mapping **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if flav or nat is null**Since:** 1.4 **See Also:**[addFlavorForUnencodedNative(java.lang.String, java.awt.datatransfer.DataFlavor)](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#addFlavorForUnencodedNative(java.lang.String,%20java.awt.datatransfer.DataFlavor))

### setNativesForFlavor

public void **setNativesForFlavor**([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) flav,  
 [String](http://docs.google.com/java/lang/String.html)[] natives)

Discards the current mappings for the specified DataFlavor and all DataFlavors equal to the specified DataFlavor, and creates new mappings to the specified String natives. Unlike getNativesForFlavor, the mappings will only be established in one direction, and the natives will not be encoded. To establish two-way mappings, call setFlavorsForNative as well. The first native in the array will represent the highest priority mapping. Subsequent natives will represent mappings of decreasing priority.

If the array contains several elements that reference equal String natives, this method will establish new mappings for the first of those elements and ignore the rest of them.

It is recommended that client code not reset mappings established by the data transfer subsystem. This method should only be used for application-level mappings.

**Parameters:**flav - the DataFlavor key for the mappingsnatives - the String native values for the mappings **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if flav or natives is null or if natives contains null elements**Since:** 1.4 **See Also:**[setFlavorsForNative(java.lang.String, java.awt.datatransfer.DataFlavor[])](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#setFlavorsForNative(java.lang.String,%20java.awt.datatransfer.DataFlavor%5B%5D))

### addFlavorForUnencodedNative

public void **addFlavorForUnencodedNative**([String](http://docs.google.com/java/lang/String.html) nat,  
 [DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) flav)

Adds a mapping from a single String native to a single DataFlavor. Unlike getFlavorsForNative, the mapping will only be established in one direction, and the native will not be encoded. To establish a two-way mapping, call addUnencodedNativeForFlavor as well. The new mapping will be of lower priority than any existing mapping. This method has no effect if a mapping from the specified String native to the specified or equal DataFlavor already exists.

**Parameters:**nat - the String native key for the mappingflav - the DataFlavor value for the mapping **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if nat or flav is null**Since:** 1.4 **See Also:**[addUnencodedNativeForFlavor(java.awt.datatransfer.DataFlavor, java.lang.String)](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#addUnencodedNativeForFlavor(java.awt.datatransfer.DataFlavor,%20java.lang.String))

### setFlavorsForNative

public void **setFlavorsForNative**([String](http://docs.google.com/java/lang/String.html) nat,  
 [DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)[] flavors)

Discards the current mappings for the specified String native, and creates new mappings to the specified DataFlavors. Unlike getFlavorsForNative, the mappings will only be established in one direction, and the natives need not be encoded. To establish two-way mappings, call setNativesForFlavor as well. The first DataFlavor in the array will represent the highest priority mapping. Subsequent DataFlavors will represent mappings of decreasing priority.

If the array contains several elements that reference equal DataFlavors, this method will establish new mappings for the first of those elements and ignore the rest of them.

It is recommended that client code not reset mappings established by the data transfer subsystem. This method should only be used for application-level mappings.

**Parameters:**nat - the String native key for the mappingsflavors - the DataFlavor values for the mappings **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if nat or flavors is null or if flavors contains null elements**Since:** 1.4 **See Also:**[setNativesForFlavor(java.awt.datatransfer.DataFlavor, java.lang.String[])](http://docs.google.com/java/awt/datatransfer/SystemFlavorMap.html#setNativesForFlavor(java.awt.datatransfer.DataFlavor,%20java.lang.String%5B%5D))

### encodeJavaMIMEType

public static [String](http://docs.google.com/java/lang/String.html) **encodeJavaMIMEType**([String](http://docs.google.com/java/lang/String.html) mimeType)

Encodes a MIME type for use as a String native. The format of an encoded representation of a MIME type is implementation-dependent. The only restrictions are:

* The encoded representation is null if and only if the MIME type String is null.
* The encoded representations for two non-null MIME type Strings are equal if and only if these Strings are equal according to String.equals(Object).

Sun's reference implementation of this method returns the specified MIME type String prefixed with JAVA\_DATAFLAVOR:.

**Parameters:**mimeType - the MIME type to encode **Returns:**the encoded String, or null if mimeType is null

### encodeDataFlavor

public static [String](http://docs.google.com/java/lang/String.html) **encodeDataFlavor**([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) flav)

Encodes a DataFlavor for use as a String native. The format of an encoded DataFlavor is implementation-dependent. The only restrictions are:

* The encoded representation is null if and only if the specified DataFlavor is null or its MIME type String is null.
* The encoded representations for two non-null DataFlavors with non-null MIME type Strings are equal if and only if the MIME type Strings of these DataFlavors are equal according to String.equals(Object).

Sun's reference implementation of this method returns the MIME type String of the specified DataFlavor prefixed with JAVA\_DATAFLAVOR:.

**Parameters:**flav - the DataFlavor to encode **Returns:**the encoded String, or null if flav is null or has a null MIME type

### isJavaMIMEType

public static boolean **isJavaMIMEType**([String](http://docs.google.com/java/lang/String.html) str)

Returns whether the specified String is an encoded Java MIME type.

**Parameters:**str - the String to test **Returns:**true if the String is encoded; false otherwise

### decodeJavaMIMEType

public static [String](http://docs.google.com/java/lang/String.html) **decodeJavaMIMEType**([String](http://docs.google.com/java/lang/String.html) nat)

Decodes a String native for use as a Java MIME type.

**Parameters:**nat - the String to decode **Returns:**the decoded Java MIME type, or null if nat is not an encoded String native

### decodeDataFlavor

public static [DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) **decodeDataFlavor**([String](http://docs.google.com/java/lang/String.html) nat)  
 throws [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html)

Decodes a String native for use as a DataFlavor.

**Parameters:**nat - the String to decode **Returns:**the decoded DataFlavor, or null if nat is not an encoded String native **Throws:** [ClassNotFoundException](http://docs.google.com/java/lang/ClassNotFoundException.html)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/SystemFlavorMap.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/awt/datatransfer/StringSelection.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/datatransfer/Transferable.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/datatransfer/SystemFlavorMap.html)    [**NO FRAMES**](http://docs.google.com/SystemFlavorMap.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#tyjcwt) |

[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).