| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DropTargetDropEvent.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/dnd/DropTargetDragEvent.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/dnd/DropTargetEvent.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/dnd/DropTargetDropEvent.html)    [**NO FRAMES**](http://docs.google.com/DropTargetDropEvent.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: FIELD | [CONSTR](#3rdcrjn) | [METHOD](#35nkun2) |

## **java.awt.dnd**

Class DropTargetDropEvent

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.util.EventObject](http://docs.google.com/java/util/EventObject.html)  
 [java.awt.dnd.DropTargetEvent](http://docs.google.com/java/awt/dnd/DropTargetEvent.html)  
 **java.awt.dnd.DropTargetDropEvent**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html)

public class **DropTargetDropEvent**extends [DropTargetEvent](http://docs.google.com/java/awt/dnd/DropTargetEvent.html)

The DropTargetDropEvent is delivered via the DropTargetListener drop() method.

The DropTargetDropEvent reports the *source drop actions* and the *user drop action* that reflect the current state of the drag-and-drop operation.

*Source drop actions* is a bitwise mask of DnDConstants that represents the set of drop actions supported by the drag source for this drag-and-drop operation.

*User drop action* depends on the drop actions supported by the drag source and the drop action selected by the user. The user can select a drop action by pressing modifier keys during the drag operation:

Ctrl + Shift -> ACTION\_LINK  
 Ctrl -> ACTION\_COPY  
 Shift -> ACTION\_MOVE

If the user selects a drop action, the *user drop action* is one of DnDConstants that represents the selected drop action if this drop action is supported by the drag source or DnDConstants.ACTION\_NONE if this drop action is not supported by the drag source.

If the user doesn't select a drop action, the set of DnDConstants that represents the set of drop actions supported by the drag source is searched for DnDConstants.ACTION\_MOVE, then for DnDConstants.ACTION\_COPY, then for DnDConstants.ACTION\_LINK and the *user drop action* is the first constant found. If no constant is found the *user drop action* is DnDConstants.ACTION\_NONE.

**Since:** 1.2 **See Also:**[Serialized Form](http://docs.google.com/serialized-form.html#java.awt.dnd.DropTargetDropEvent)

| **Field Summary** | |
| --- | --- |

| **Fields inherited from class java.awt.dnd.**[**DropTargetEvent**](http://docs.google.com/java/awt/dnd/DropTargetEvent.html) |
| --- |
| [context](http://docs.google.com/java/awt/dnd/DropTargetEvent.html#context) |

| **Fields inherited from class java.util.**[**EventObject**](http://docs.google.com/java/util/EventObject.html) |
| --- |
| [source](http://docs.google.com/java/util/EventObject.html#source) |

| **Constructor Summary** | |
| --- | --- |
| [**DropTargetDropEvent**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#DropTargetDropEvent(java.awt.dnd.DropTargetContext,%20java.awt.Point,%20int,%20int))([DropTargetContext](http://docs.google.com/java/awt/dnd/DropTargetContext.html) dtc, [Point](http://docs.google.com/java/awt/Point.html) cursorLocn, int dropAction, int srcActions)            Construct a DropTargetDropEvent given the DropTargetContext for this operation, the location of the drag Cursor's hotspot in the Component's coordinates, the currently selected user drop action, and the current set of actions supported by the source. |
| [**DropTargetDropEvent**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#DropTargetDropEvent(java.awt.dnd.DropTargetContext,%20java.awt.Point,%20int,%20int,%20boolean))([DropTargetContext](http://docs.google.com/java/awt/dnd/DropTargetContext.html) dtc, [Point](http://docs.google.com/java/awt/Point.html) cursorLocn, int dropAction, int srcActions, boolean isLocal)            Construct a DropTargetEvent given the DropTargetContext for this operation, the location of the drag Cursor's hotspot in the Component's coordinates, the currently selected user drop action, the current set of actions supported by the source, and a boolean indicating if the source is in the same JVM as the target. |

| **Method Summary** | |
| --- | --- |
| void | [**acceptDrop**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#acceptDrop(int))(int dropAction)            accept the drop, using the specified action. |
| void | [**dropComplete**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#dropComplete(boolean))(boolean success)            This method notifies the DragSource that the drop transfer(s) are completed. |
| [DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)[] | [**getCurrentDataFlavors**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#getCurrentDataFlavors())()            This method returns the current DataFlavors. |
| [List](http://docs.google.com/java/util/List.html)<[DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)> | [**getCurrentDataFlavorsAsList**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#getCurrentDataFlavorsAsList())()            This method returns the currently available DataFlavors as a java.util.List. |
| int | [**getDropAction**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#getDropAction())()            This method returns the user drop action. |
| [Point](http://docs.google.com/java/awt/Point.html) | [**getLocation**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#getLocation())()            This method returns a Point indicating the Cursor's current location in the Component's coordinates. |
| int | [**getSourceActions**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#getSourceActions())()            This method returns the source drop actions. |
| [Transferable](http://docs.google.com/java/awt/datatransfer/Transferable.html) | [**getTransferable**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#getTransferable())()            This method returns the Transferable object associated with the drop. |
| boolean | [**isDataFlavorSupported**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#isDataFlavorSupported(java.awt.datatransfer.DataFlavor))([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) df)            This method returns a boolean indicating if the specified DataFlavor is available from the source. |
| boolean | [**isLocalTransfer**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#isLocalTransfer())()            This method returns an int indicating if the source is in the same JVM as the target. |
| void | [**rejectDrop**](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#rejectDrop())()            reject the Drop. |

| **Methods inherited from class java.awt.dnd.**[**DropTargetEvent**](http://docs.google.com/java/awt/dnd/DropTargetEvent.html) |
| --- |
| [getDropTargetContext](http://docs.google.com/java/awt/dnd/DropTargetEvent.html#getDropTargetContext()) |

| **Methods inherited from class java.util.**[**EventObject**](http://docs.google.com/java/util/EventObject.html) |
| --- |
| [getSource](http://docs.google.com/java/util/EventObject.html#getSource()), [toString](http://docs.google.com/java/util/EventObject.html#toString()) |

| **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()), [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** |
| --- |

### DropTargetDropEvent

public **DropTargetDropEvent**([DropTargetContext](http://docs.google.com/java/awt/dnd/DropTargetContext.html) dtc,  
 [Point](http://docs.google.com/java/awt/Point.html) cursorLocn,  
 int dropAction,  
 int srcActions)

Construct a DropTargetDropEvent given the DropTargetContext for this operation, the location of the drag Cursor's hotspot in the Component's coordinates, the currently selected user drop action, and the current set of actions supported by the source. By default, this constructor assumes that the target is not in the same virtual machine as the source; that is, [isLocalTransfer()](http://docs.google.com/java/awt/dnd/DropTargetDropEvent.html#isLocalTransfer()) will return false.

**Parameters:**dtc - The DropTargetContext for this operationcursorLocn - The location of the "Drag" Cursor's hotspot in Component coordinatesdropAction - the user drop action.srcActions - the source drop actions. **Throws:** NullPointerException - if cursorLocn is null IllegalArgumentException - if dropAction is not one of DnDConstants. IllegalArgumentException - if srcActions is not a bitwise mask of DnDConstants. IllegalArgumentException - if dtc is null.

### DropTargetDropEvent

public **DropTargetDropEvent**([DropTargetContext](http://docs.google.com/java/awt/dnd/DropTargetContext.html) dtc,  
 [Point](http://docs.google.com/java/awt/Point.html) cursorLocn,  
 int dropAction,  
 int srcActions,  
 boolean isLocal)

Construct a DropTargetEvent given the DropTargetContext for this operation, the location of the drag Cursor's hotspot in the Component's coordinates, the currently selected user drop action, the current set of actions supported by the source, and a boolean indicating if the source is in the same JVM as the target.

**Parameters:**dtc - The DropTargetContext for this operationcursorLocn - The location of the "Drag" Cursor's hotspot in Component's coordinatesdropAction - the user drop action.srcActions - the source drop actions.isLocal - True if the source is in the same JVM as the target **Throws:** NullPointerException - if cursorLocn is null IllegalArgumentException - if dropAction is not one of DnDConstants. IllegalArgumentException - if srcActions is not a bitwise mask of DnDConstants. IllegalArgumentException - if dtc is null.

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

### getLocation

public [Point](http://docs.google.com/java/awt/Point.html) **getLocation**()

This method returns a Point indicating the Cursor's current location in the Component's coordinates.

**Returns:**the current Cursor location in Component's coords.

### getCurrentDataFlavors

public [DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html)[] **getCurrentDataFlavors**()

This method returns the current DataFlavors.

**Returns:**current DataFlavors

### getCurrentDataFlavorsAsList

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

This method returns the currently available DataFlavors as a java.util.List.

**Returns:**the currently available DataFlavors as a java.util.List

### isDataFlavorSupported

public boolean **isDataFlavorSupported**([DataFlavor](http://docs.google.com/java/awt/datatransfer/DataFlavor.html) df)

This method returns a boolean indicating if the specified DataFlavor is available from the source.

**Parameters:**df - the DataFlavor to test

**Returns:**if the DataFlavor specified is available from the source

### getSourceActions

public int **getSourceActions**()

This method returns the source drop actions.

**Returns:**the source drop actions.

### getDropAction

public int **getDropAction**()

This method returns the user drop action.

**Returns:**the user drop actions.

### getTransferable

public [Transferable](http://docs.google.com/java/awt/datatransfer/Transferable.html) **getTransferable**()

This method returns the Transferable object associated with the drop.

**Returns:**the Transferable associated with the drop

### acceptDrop

public void **acceptDrop**(int dropAction)

accept the drop, using the specified action.

**Parameters:**dropAction - the specified action

### rejectDrop

public void **rejectDrop**()

reject the Drop.

### dropComplete

public void **dropComplete**(boolean success)

This method notifies the DragSource that the drop transfer(s) are completed.

**Parameters:**success - a boolean indicating that the drop transfer(s) are completed.

### isLocalTransfer

public boolean **isLocalTransfer**()

This method returns an int indicating if the source is in the same JVM as the target.

**Returns:**if the Source is in the same JVM

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DropTargetDropEvent.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/dnd/DropTargetDragEvent.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/dnd/DropTargetEvent.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/dnd/DropTargetDropEvent.html)    [**NO FRAMES**](http://docs.google.com/DropTargetDropEvent.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: FIELD | [CONSTR](#3rdcrjn) | [METHOD](#35nkun2) |

[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).