| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/EventTarget.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/org/w3c/dom/events/EventListener.html)   [**NEXT CLASS**](http://docs.google.com/org/w3c/dom/events/MouseEvent.html) | [**FRAMES**](http://docs.google.com/index.html?org/w3c/dom/events/EventTarget.html)    [**NO FRAMES**](http://docs.google.com/EventTarget.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **org.w3c.dom.events**

Interface EventTarget

public interface **EventTarget**

The EventTarget interface is implemented by all Nodes in an implementation which supports the DOM Event Model. Therefore, this interface can be obtained by using binding-specific casting methods on an instance of the Node interface. The interface allows registration and removal of EventListeners on an EventTarget and dispatch of events to that EventTarget.

See also the [Document Object Model (DOM) Level 2 Events Specification](http://www.w3.org/TR/2000/REC-DOM-Level-2-Events-20001113).

**Since:** DOM Level 2

| **Method Summary** | |
| --- | --- |
| void | [**addEventListener**](http://docs.google.com/org/w3c/dom/events/EventTarget.html#addEventListener(java.lang.String,%20org.w3c.dom.events.EventListener,%20boolean))([String](http://docs.google.com/java/lang/String.html) type, [EventListener](http://docs.google.com/org/w3c/dom/events/EventListener.html) listener, boolean useCapture)            This method allows the registration of event listeners on the event target. |
| boolean | [**dispatchEvent**](http://docs.google.com/org/w3c/dom/events/EventTarget.html#dispatchEvent(org.w3c.dom.events.Event))([Event](http://docs.google.com/org/w3c/dom/events/Event.html) evt)            This method allows the dispatch of events into the implementations event model. |
| void | [**removeEventListener**](http://docs.google.com/org/w3c/dom/events/EventTarget.html#removeEventListener(java.lang.String,%20org.w3c.dom.events.EventListener,%20boolean))([String](http://docs.google.com/java/lang/String.html) type, [EventListener](http://docs.google.com/org/w3c/dom/events/EventListener.html) listener, boolean useCapture)            This method allows the removal of event listeners from the event target. |

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

### addEventListener

void **addEventListener**([String](http://docs.google.com/java/lang/String.html) type,  
 [EventListener](http://docs.google.com/org/w3c/dom/events/EventListener.html) listener,  
 boolean useCapture)

This method allows the registration of event listeners on the event target. If an EventListener is added to an EventTarget while it is processing an event, it will not be triggered by the current actions but may be triggered during a later stage of event flow, such as the bubbling phase.

If multiple identical EventListeners are registered on the same EventTarget with the same parameters the duplicate instances are discarded. They do not cause the EventListener to be called twice and since they are discarded they do not need to be removed with the removeEventListener method.

**Parameters:**type - The event type for which the user is registeringlistener - The listener parameter takes an interface implemented by the user which contains the methods to be called when the event occurs.useCapture - If true, useCapture indicates that the user wishes to initiate capture. After initiating capture, all events of the specified type will be dispatched to the registered EventListener before being dispatched to any EventTargets beneath them in the tree. Events which are bubbling upward through the tree will not trigger an EventListener designated to use capture.

### removeEventListener

void **removeEventListener**([String](http://docs.google.com/java/lang/String.html) type,  
 [EventListener](http://docs.google.com/org/w3c/dom/events/EventListener.html) listener,  
 boolean useCapture)

This method allows the removal of event listeners from the event target. If an EventListener is removed from an EventTarget while it is processing an event, it will not be triggered by the current actions. EventListeners can never be invoked after being removed.

Calling removeEventListener with arguments which do not identify any currently registered EventListener on the EventTarget has no effect.

**Parameters:**type - Specifies the event type of the EventListener being removed.listener - The EventListener parameter indicates the EventListener to be removed.useCapture - Specifies whether the EventListener being removed was registered as a capturing listener or not. If a listener was registered twice, one with capture and one without, each must be removed separately. Removal of a capturing listener does not affect a non-capturing version of the same listener, and vice versa.

### dispatchEvent

boolean **dispatchEvent**([Event](http://docs.google.com/org/w3c/dom/events/Event.html) evt)  
 throws [EventException](http://docs.google.com/org/w3c/dom/events/EventException.html)

This method allows the dispatch of events into the implementations event model. Events dispatched in this manner will have the same capturing and bubbling behavior as events dispatched directly by the implementation. The target of the event is the EventTarget on which dispatchEvent is called.

**Parameters:**evt - Specifies the event type, behavior, and contextual information to be used in processing the event. **Returns:**The return value of dispatchEvent indicates whether any of the listeners which handled the event called preventDefault. If preventDefault was called the value is false, else the value is true. **Throws:** [EventException](http://docs.google.com/org/w3c/dom/events/EventException.html) - UNSPECIFIED\_EVENT\_TYPE\_ERR: Raised if the Event's type was not specified by initializing the event before dispatchEvent was called. Specification of the Event's type as null or an empty string will also trigger this exception.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/EventTarget.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/org/w3c/dom/events/EventListener.html)   [**NEXT CLASS**](http://docs.google.com/org/w3c/dom/events/MouseEvent.html) | [**FRAMES**](http://docs.google.com/index.html?org/w3c/dom/events/EventTarget.html)    [**NO FRAMES**](http://docs.google.com/EventTarget.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[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).