| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Event.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/DocumentEvent.html)   [**NEXT CLASS**](http://docs.google.com/org/w3c/dom/events/EventException.html) | [**FRAMES**](http://docs.google.com/index.html?org/w3c/dom/events/Event.html)    [**NO FRAMES**](http://docs.google.com/Event.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | CONSTR | [METHOD](#2et92p0) | DETAIL: [FIELD](#tyjcwt) | CONSTR | [METHOD](#2s8eyo1) |

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

Interface Event

**All Known Subinterfaces:** [LSLoadEvent](http://docs.google.com/org/w3c/dom/ls/LSLoadEvent.html), [LSProgressEvent](http://docs.google.com/org/w3c/dom/ls/LSProgressEvent.html), [MouseEvent](http://docs.google.com/org/w3c/dom/events/MouseEvent.html), [MutationEvent](http://docs.google.com/org/w3c/dom/events/MutationEvent.html), [UIEvent](http://docs.google.com/org/w3c/dom/events/UIEvent.html)

public interface **Event**

The Event interface is used to provide contextual information about an event to the handler processing the event. An object which implements the Event interface is generally passed as the first parameter to an event handler. More specific context information is passed to event handlers by deriving additional interfaces from Event which contain information directly relating to the type of event they accompany. These derived interfaces are also implemented by the object passed to the event listener.

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

| **Field Summary** | |
| --- | --- |
| static short | [**AT\_TARGET**](http://docs.google.com/org/w3c/dom/events/Event.html#AT_TARGET)            The event is currently being evaluated at the target EventTarget. |
| static short | [**BUBBLING\_PHASE**](http://docs.google.com/org/w3c/dom/events/Event.html#BUBBLING_PHASE)            The current event phase is the bubbling phase. |
| static short | [**CAPTURING\_PHASE**](http://docs.google.com/org/w3c/dom/events/Event.html#CAPTURING_PHASE)            The current event phase is the capturing phase. |

| **Method Summary** | |
| --- | --- |
| boolean | [**getBubbles**](http://docs.google.com/org/w3c/dom/events/Event.html#getBubbles())()            Used to indicate whether or not an event is a bubbling event. |
| boolean | [**getCancelable**](http://docs.google.com/org/w3c/dom/events/Event.html#getCancelable())()            Used to indicate whether or not an event can have its default action prevented. |
| [EventTarget](http://docs.google.com/org/w3c/dom/events/EventTarget.html) | [**getCurrentTarget**](http://docs.google.com/org/w3c/dom/events/Event.html#getCurrentTarget())()            Used to indicate the EventTarget whose EventListeners are currently being processed. |
| short | [**getEventPhase**](http://docs.google.com/org/w3c/dom/events/Event.html#getEventPhase())()            Used to indicate which phase of event flow is currently being evaluated. |
| [EventTarget](http://docs.google.com/org/w3c/dom/events/EventTarget.html) | [**getTarget**](http://docs.google.com/org/w3c/dom/events/Event.html#getTarget())()            Used to indicate the EventTarget to which the event was originally dispatched. |
| long | [**getTimeStamp**](http://docs.google.com/org/w3c/dom/events/Event.html#getTimeStamp())()            Used to specify the time (in milliseconds relative to the epoch) at which the event was created. |
| [String](http://docs.google.com/java/lang/String.html) | [**getType**](http://docs.google.com/org/w3c/dom/events/Event.html#getType())()            The name of the event (case-insensitive). |
| void | [**initEvent**](http://docs.google.com/org/w3c/dom/events/Event.html#initEvent(java.lang.String,%20boolean,%20boolean))([String](http://docs.google.com/java/lang/String.html) eventTypeArg, boolean canBubbleArg, boolean cancelableArg)            The initEvent method is used to initialize the value of an Event created through the DocumentEvent interface. |
| void | [**preventDefault**](http://docs.google.com/org/w3c/dom/events/Event.html#preventDefault())()            If an event is cancelable, the preventDefault method is used to signify that the event is to be canceled, meaning any default action normally taken by the implementation as a result of the event will not occur. |
| void | [**stopPropagation**](http://docs.google.com/org/w3c/dom/events/Event.html#stopPropagation())()            The stopPropagation method is used prevent further propagation of an event during event flow. |

| **Field Detail** |
| --- |

### CAPTURING\_PHASE

static final short **CAPTURING\_PHASE**

The current event phase is the capturing phase.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#org.w3c.dom.events.Event.CAPTURING_PHASE)

### AT\_TARGET

static final short **AT\_TARGET**

The event is currently being evaluated at the target EventTarget.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#org.w3c.dom.events.Event.AT_TARGET)

### BUBBLING\_PHASE

static final short **BUBBLING\_PHASE**

The current event phase is the bubbling phase.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#org.w3c.dom.events.Event.BUBBLING_PHASE)

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

### getType

[String](http://docs.google.com/java/lang/String.html) **getType**()

The name of the event (case-insensitive). The name must be an XML name.

### getTarget

[EventTarget](http://docs.google.com/org/w3c/dom/events/EventTarget.html) **getTarget**()

Used to indicate the EventTarget to which the event was originally dispatched.

### getCurrentTarget

[EventTarget](http://docs.google.com/org/w3c/dom/events/EventTarget.html) **getCurrentTarget**()

Used to indicate the EventTarget whose EventListeners are currently being processed. This is particularly useful during capturing and bubbling.

### getEventPhase

short **getEventPhase**()

Used to indicate which phase of event flow is currently being evaluated.

### getBubbles

boolean **getBubbles**()

Used to indicate whether or not an event is a bubbling event. If the event can bubble the value is true, else the value is false.

### getCancelable

boolean **getCancelable**()

Used to indicate whether or not an event can have its default action prevented. If the default action can be prevented the value is true, else the value is false.

### getTimeStamp

long **getTimeStamp**()

Used to specify the time (in milliseconds relative to the epoch) at which the event was created. Due to the fact that some systems may not provide this information the value of timeStamp may be not available for all events. When not available, a value of 0 will be returned. Examples of epoch time are the time of the system start or 0:0:0 UTC 1st January 1970.

### stopPropagation

void **stopPropagation**()

The stopPropagation method is used prevent further propagation of an event during event flow. If this method is called by any EventListener the event will cease propagating through the tree. The event will complete dispatch to all listeners on the current EventTarget before event flow stops. This method may be used during any stage of event flow.

### preventDefault

void **preventDefault**()

If an event is cancelable, the preventDefault method is used to signify that the event is to be canceled, meaning any default action normally taken by the implementation as a result of the event will not occur. If, during any stage of event flow, the preventDefault method is called the event is canceled. Any default action associated with the event will not occur. Calling this method for a non-cancelable event has no effect. Once preventDefault has been called it will remain in effect throughout the remainder of the event's propagation. This method may be used during any stage of event flow.

### initEvent

void **initEvent**([String](http://docs.google.com/java/lang/String.html) eventTypeArg,  
 boolean canBubbleArg,  
 boolean cancelableArg)

The initEvent method is used to initialize the value of an Event created through the DocumentEvent interface. This method may only be called before the Event has been dispatched via the dispatchEvent method, though it may be called multiple times during that phase if necessary. If called multiple times the final invocation takes precedence. If called from a subclass of Event interface only the values specified in the initEvent method are modified, all other attributes are left unchanged.

**Parameters:**eventTypeArg - Specifies the event type. This type may be any event type currently defined in this specification or a new event type.. The string must be an XML name. Any new event type must not begin with any upper, lower, or mixed case version of the string "DOM". This prefix is reserved for future DOM event sets. It is also strongly recommended that third parties adding their own events use their own prefix to avoid confusion and lessen the probability of conflicts with other new events.canBubbleArg - Specifies whether or not the event can bubble.cancelableArg - Specifies whether or not the event's default action can be prevented.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Event.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/DocumentEvent.html)   [**NEXT CLASS**](http://docs.google.com/org/w3c/dom/events/EventException.html) | [**FRAMES**](http://docs.google.com/index.html?org/w3c/dom/events/Event.html)    [**NO FRAMES**](http://docs.google.com/Event.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | CONSTR | [METHOD](#2et92p0) | DETAIL: [FIELD](#tyjcwt) | CONSTR | [METHOD](#2s8eyo1) |

[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).