| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/PropertyChangeSupport.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/beans/PropertyChangeListenerProxy.html)   [**NEXT CLASS**](http://docs.google.com/java/beans/PropertyDescriptor.html) | [**FRAMES**](http://docs.google.com/index.html?java/beans/PropertyChangeSupport.html)    [**NO FRAMES**](http://docs.google.com/PropertyChangeSupport.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

## **java.beans**

Class PropertyChangeSupport

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **java.beans.PropertyChangeSupport**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html) **Direct Known Subclasses:** [SwingPropertyChangeSupport](http://docs.google.com/javax/swing/event/SwingPropertyChangeSupport.html)

public class **PropertyChangeSupport**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Serializable](http://docs.google.com/java/io/Serializable.html)

This is a utility class that can be used by beans that support bound properties. You can use an instance of this class as a member field of your bean and delegate various work to it. This class is serializable. When it is serialized it will save (and restore) any listeners that are themselves serializable. Any non-serializable listeners will be skipped during serialization.

**See Also:**[Serialized Form](http://docs.google.com/serialized-form.html#java.beans.PropertyChangeSupport)

| **Constructor Summary** | |
| --- | --- |
| [**PropertyChangeSupport**](http://docs.google.com/java/beans/PropertyChangeSupport.html#PropertyChangeSupport(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) sourceBean)            Constructs a PropertyChangeSupport object. |

| **Method Summary** | |
| --- | --- |
| void | [**addPropertyChangeListener**](http://docs.google.com/java/beans/PropertyChangeSupport.html#addPropertyChangeListener(java.beans.PropertyChangeListener))([PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)            Add a PropertyChangeListener to the listener list. |
| void | [**addPropertyChangeListener**](http://docs.google.com/java/beans/PropertyChangeSupport.html#addPropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener))([String](http://docs.google.com/java/lang/String.html) propertyName, [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)            Add a PropertyChangeListener for a specific property. |
| void | [**fireIndexedPropertyChange**](http://docs.google.com/java/beans/PropertyChangeSupport.html#fireIndexedPropertyChange(java.lang.String,%20int,%20boolean,%20boolean))([String](http://docs.google.com/java/lang/String.html) propertyName, int index, boolean oldValue, boolean newValue)            Report a boolean bound indexed property update to any registered listeners. |
| void | [**fireIndexedPropertyChange**](http://docs.google.com/java/beans/PropertyChangeSupport.html#fireIndexedPropertyChange(java.lang.String,%20int,%20int,%20int))([String](http://docs.google.com/java/lang/String.html) propertyName, int index, int oldValue, int newValue)            Report an int bound indexed property update to any registered listeners. |
| void | [**fireIndexedPropertyChange**](http://docs.google.com/java/beans/PropertyChangeSupport.html#fireIndexedPropertyChange(java.lang.String,%20int,%20java.lang.Object,%20java.lang.Object))([String](http://docs.google.com/java/lang/String.html) propertyName, int index, [Object](http://docs.google.com/java/lang/Object.html) oldValue, [Object](http://docs.google.com/java/lang/Object.html) newValue)            Report a bound indexed property update to any registered listeners. |
| void | [**firePropertyChange**](http://docs.google.com/java/beans/PropertyChangeSupport.html#firePropertyChange(java.beans.PropertyChangeEvent))([PropertyChangeEvent](http://docs.google.com/java/beans/PropertyChangeEvent.html) evt)            Fire an existing PropertyChangeEvent to any registered listeners. |
| void | [**firePropertyChange**](http://docs.google.com/java/beans/PropertyChangeSupport.html#firePropertyChange(java.lang.String,%20boolean,%20boolean))([String](http://docs.google.com/java/lang/String.html) propertyName, boolean oldValue, boolean newValue)            Report a boolean bound property update to any registered listeners. |
| void | [**firePropertyChange**](http://docs.google.com/java/beans/PropertyChangeSupport.html#firePropertyChange(java.lang.String,%20int,%20int))([String](http://docs.google.com/java/lang/String.html) propertyName, int oldValue, int newValue)            Report an int bound property update to any registered listeners. |
| void | [**firePropertyChange**](http://docs.google.com/java/beans/PropertyChangeSupport.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object))([String](http://docs.google.com/java/lang/String.html) propertyName, [Object](http://docs.google.com/java/lang/Object.html) oldValue, [Object](http://docs.google.com/java/lang/Object.html) newValue)            Report a bound property update to any registered listeners. |
| [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html)[] | [**getPropertyChangeListeners**](http://docs.google.com/java/beans/PropertyChangeSupport.html#getPropertyChangeListeners())()            Returns an array of all the listeners that were added to the PropertyChangeSupport object with addPropertyChangeListener(). |
| [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html)[] | [**getPropertyChangeListeners**](http://docs.google.com/java/beans/PropertyChangeSupport.html#getPropertyChangeListeners(java.lang.String))([String](http://docs.google.com/java/lang/String.html) propertyName)            Returns an array of all the listeners which have been associated with the named property. |
| boolean | [**hasListeners**](http://docs.google.com/java/beans/PropertyChangeSupport.html#hasListeners(java.lang.String))([String](http://docs.google.com/java/lang/String.html) propertyName)            Check if there are any listeners for a specific property, including those registered on all properties. |
| void | [**removePropertyChangeListener**](http://docs.google.com/java/beans/PropertyChangeSupport.html#removePropertyChangeListener(java.beans.PropertyChangeListener))([PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)            Remove a PropertyChangeListener from the listener list. |
| void | [**removePropertyChangeListener**](http://docs.google.com/java/beans/PropertyChangeSupport.html#removePropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener))([String](http://docs.google.com/java/lang/String.html) propertyName, [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)            Remove a PropertyChangeListener for a specific property. |

| **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)) |

| **Constructor Detail** |
| --- |

### PropertyChangeSupport

public **PropertyChangeSupport**([Object](http://docs.google.com/java/lang/Object.html) sourceBean)

Constructs a PropertyChangeSupport object.

**Parameters:**sourceBean - The bean to be given as the source for any events.

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

### addPropertyChangeListener

public void **addPropertyChangeListener**([PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)

Add a PropertyChangeListener to the listener list. The listener is registered for all properties. The same listener object may be added more than once, and will be called as many times as it is added. If listener is null, no exception is thrown and no action is taken.

**Parameters:**listener - The PropertyChangeListener to be added

### removePropertyChangeListener

public void **removePropertyChangeListener**([PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)

Remove a PropertyChangeListener from the listener list. This removes a PropertyChangeListener that was registered for all properties. If listener was added more than once to the same event source, it will be notified one less time after being removed. If listener is null, or was never added, no exception is thrown and no action is taken.

**Parameters:**listener - The PropertyChangeListener to be removed

### getPropertyChangeListeners

public [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html)[] **getPropertyChangeListeners**()

Returns an array of all the listeners that were added to the PropertyChangeSupport object with addPropertyChangeListener().

If some listeners have been added with a named property, then the returned array will be a mixture of PropertyChangeListeners and PropertyChangeListenerProxys. If the calling method is interested in distinguishing the listeners then it must test each element to see if it's a PropertyChangeListenerProxy, perform the cast, and examine the parameter.

PropertyChangeListener[] listeners = bean.getPropertyChangeListeners();  
 for (int i = 0; i < listeners.length; i++) {  
 if (listeners[i] instanceof PropertyChangeListenerProxy) {  
 PropertyChangeListenerProxy proxy =   
 (PropertyChangeListenerProxy)listeners[i];  
 if (proxy.getPropertyName().equals("foo")) {  
 // proxy is a PropertyChangeListener which was associated  
 // with the property named "foo"  
 }  
 }  
 }

**Returns:**all of the PropertyChangeListeners added or an empty array if no listeners have been added**Since:** 1.4 **See Also:**[PropertyChangeListenerProxy](http://docs.google.com/java/beans/PropertyChangeListenerProxy.html)

### addPropertyChangeListener

public void **addPropertyChangeListener**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)

Add a PropertyChangeListener for a specific property. The listener will be invoked only when a call on firePropertyChange names that specific property. The same listener object may be added more than once. For each property, the listener will be invoked the number of times it was added for that property. If propertyName or listener is null, no exception is thrown and no action is taken.

**Parameters:**propertyName - The name of the property to listen on.listener - The PropertyChangeListener to be added

### removePropertyChangeListener

public void **removePropertyChangeListener**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) listener)

Remove a PropertyChangeListener for a specific property. If listener was added more than once to the same event source for the specified property, it will be notified one less time after being removed. If propertyName is null, no exception is thrown and no action is taken. If listener is null, or was never added for the specified property, no exception is thrown and no action is taken.

**Parameters:**propertyName - The name of the property that was listened on.listener - The PropertyChangeListener to be removed

### getPropertyChangeListeners

public [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html)[] **getPropertyChangeListeners**([String](http://docs.google.com/java/lang/String.html) propertyName)

Returns an array of all the listeners which have been associated with the named property.

**Parameters:**propertyName - The name of the property being listened to **Returns:**all of the PropertyChangeListeners associated with the named property. If no such listeners have been added, or if propertyName is null, an empty array is returned.**Since:** 1.4

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 [Object](http://docs.google.com/java/lang/Object.html) oldValue,  
 [Object](http://docs.google.com/java/lang/Object.html) newValue)

Report a bound property update to any registered listeners. No event is fired if old and new are equal and non-null.

This is merely a convenience wrapper around the more general firePropertyChange method that takes PropertyChangeEvent value.

**Parameters:**propertyName - The programmatic name of the property that was changed.oldValue - The old value of the property.newValue - The new value of the property.

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 int oldValue,  
 int newValue)

Report an int bound property update to any registered listeners. No event is fired if old and new are equal.

This is merely a convenience wrapper around the more general firePropertyChange method that takes Object values.

**Parameters:**propertyName - The programmatic name of the property that was changed.oldValue - The old value of the property.newValue - The new value of the property.

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 boolean oldValue,  
 boolean newValue)

Report a boolean bound property update to any registered listeners. No event is fired if old and new are equal.

This is merely a convenience wrapper around the more general firePropertyChange method that takes Object values.

**Parameters:**propertyName - The programmatic name of the property that was changed.oldValue - The old value of the property.newValue - The new value of the property.

### firePropertyChange

public void **firePropertyChange**([PropertyChangeEvent](http://docs.google.com/java/beans/PropertyChangeEvent.html) evt)

Fire an existing PropertyChangeEvent to any registered listeners. No event is fired if the given event's old and new values are equal and non-null.

**Parameters:**evt - The PropertyChangeEvent object.

### fireIndexedPropertyChange

public void **fireIndexedPropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 int index,  
 [Object](http://docs.google.com/java/lang/Object.html) oldValue,  
 [Object](http://docs.google.com/java/lang/Object.html) newValue)

Report a bound indexed property update to any registered listeners.

No event is fired if old and new values are equal and non-null.

This is merely a convenience wrapper around the more general firePropertyChange method that takes PropertyChangeEvent value.

**Parameters:**propertyName - The programmatic name of the property that was changed.index - index of the property element that was changed.oldValue - The old value of the property.newValue - The new value of the property.**Since:** 1.5

### fireIndexedPropertyChange

public void **fireIndexedPropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 int index,  
 int oldValue,  
 int newValue)

Report an int bound indexed property update to any registered listeners.

No event is fired if old and new values are equal.

This is merely a convenience wrapper around the more general fireIndexedPropertyChange method which takes Object values.

**Parameters:**propertyName - The programmatic name of the property that was changed.index - index of the property element that was changed.oldValue - The old value of the property.newValue - The new value of the property.**Since:** 1.5

### fireIndexedPropertyChange

public void **fireIndexedPropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 int index,  
 boolean oldValue,  
 boolean newValue)

Report a boolean bound indexed property update to any registered listeners.

No event is fired if old and new values are equal.

This is merely a convenience wrapper around the more general fireIndexedPropertyChange method which takes Object values.

**Parameters:**propertyName - The programmatic name of the property that was changed.index - index of the property element that was changed.oldValue - The old value of the property.newValue - The new value of the property.**Since:** 1.5

### hasListeners

public boolean **hasListeners**([String](http://docs.google.com/java/lang/String.html) propertyName)

Check if there are any listeners for a specific property, including those registered on all properties. If propertyName is null, only check for listeners registered on all properties.

**Parameters:**propertyName - the property name. **Returns:**true if there are one or more listeners for the given property

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/PropertyChangeSupport.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/beans/PropertyChangeListenerProxy.html)   [**NEXT CLASS**](http://docs.google.com/java/beans/PropertyDescriptor.html) | [**FRAMES**](http://docs.google.com/index.html?java/beans/PropertyChangeSupport.html)    [**NO FRAMES**](http://docs.google.com/PropertyChangeSupport.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

[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).