| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ContainerOrderFocusTraversalPolicy.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/Container.AccessibleAWTContainer.AccessibleContainerHandler.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/Cursor.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/ContainerOrderFocusTraversalPolicy.html)    [**NO FRAMES**](http://docs.google.com/ContainerOrderFocusTraversalPolicy.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#2s8eyo1) |

## **java.awt**

Class ContainerOrderFocusTraversalPolicy

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

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html) **Direct Known Subclasses:** [DefaultFocusTraversalPolicy](http://docs.google.com/java/awt/DefaultFocusTraversalPolicy.html)

public class **ContainerOrderFocusTraversalPolicy**extends [FocusTraversalPolicy](http://docs.google.com/java/awt/FocusTraversalPolicy.html)implements [Serializable](http://docs.google.com/java/io/Serializable.html)

A FocusTraversalPolicy that determines traversal order based on the order of child Components in a Container. From a particular focus cycle root, the policy makes a pre-order traversal of the Component hierarchy, and traverses a Container's children according to the ordering of the array returned by Container.getComponents(). Portions of the hierarchy that are not visible and displayable will not be searched.

By default, ContainerOrderFocusTraversalPolicy implicitly transfers focus down-cycle. That is, during normal forward focus traversal, the Component traversed after a focus cycle root will be the focus-cycle-root's default Component to focus. This behavior can be disabled using the setImplicitDownCycleTraversal method.

By default, methods of this class with return a Component only if it is visible, displayable, enabled, and focusable. Subclasses can modify this behavior by overriding the accept method.

This policy takes into account [focus traversal policy providers](http://docs.google.com/doc-files/FocusSpec.html#FocusTraversalPolicyProviders). When searching for first/last/next/previous Component, if a focus traversal policy provider is encountered, its focus traversal policy is used to perform the search operation.

**Since:** 1.4 **See Also:**[Container.getComponents()](http://docs.google.com/java/awt/Container.html#getComponents()), [Serialized Form](http://docs.google.com/serialized-form.html#java.awt.ContainerOrderFocusTraversalPolicy)

| **Constructor Summary** | |
| --- | --- |
| [**ContainerOrderFocusTraversalPolicy**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#ContainerOrderFocusTraversalPolicy())() |

| **Method Summary** | |
| --- | --- |
| protected  boolean | [**accept**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#accept(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) aComponent)            Determines whether a Component is an acceptable choice as the new focus owner. |
| [Component](http://docs.google.com/java/awt/Component.html) | [**getComponentAfter**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getComponentAfter(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent)            Returns the Component that should receive the focus after aComponent. |
| [Component](http://docs.google.com/java/awt/Component.html) | [**getComponentBefore**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getComponentBefore(java.awt.Container,%20java.awt.Component))([Container](http://docs.google.com/java/awt/Container.html) aContainer, [Component](http://docs.google.com/java/awt/Component.html) aComponent)            Returns the Component that should receive the focus before aComponent. |
| [Component](http://docs.google.com/java/awt/Component.html) | [**getDefaultComponent**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getDefaultComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the default Component to focus. |
| [Component](http://docs.google.com/java/awt/Component.html) | [**getFirstComponent**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getFirstComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the first Component in the traversal cycle. |
| boolean | [**getImplicitDownCycleTraversal**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getImplicitDownCycleTraversal())()            Returns whether this ContainerOrderFocusTraversalPolicy transfers focus down-cycle implicitly. |
| [Component](http://docs.google.com/java/awt/Component.html) | [**getLastComponent**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getLastComponent(java.awt.Container))([Container](http://docs.google.com/java/awt/Container.html) aContainer)            Returns the last Component in the traversal cycle. |
| void | [**setImplicitDownCycleTraversal**](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#setImplicitDownCycleTraversal(boolean))(boolean implicitDownCycleTraversal)            Sets whether this ContainerOrderFocusTraversalPolicy transfers focus down-cycle implicitly. |

| **Methods inherited from class java.awt.**[**FocusTraversalPolicy**](http://docs.google.com/java/awt/FocusTraversalPolicy.html) |
| --- |
| [getInitialComponent](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getInitialComponent(java.awt.Window)) |

| **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** |
| --- |

### ContainerOrderFocusTraversalPolicy

public **ContainerOrderFocusTraversalPolicy**()

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

### getComponentAfter

public [Component](http://docs.google.com/java/awt/Component.html) **getComponentAfter**([Container](http://docs.google.com/java/awt/Container.html) aContainer,  
 [Component](http://docs.google.com/java/awt/Component.html) aComponent)

Returns the Component that should receive the focus after aComponent. aContainer must be a focus cycle root of aComponent or a focus traversal policy provider.

By default, ContainerOrderFocusTraversalPolicy implicitly transfers focus down-cycle. That is, during normal forward focus traversal, the Component traversed after a focus cycle root will be the focus-cycle- root's default Component to focus. This behavior can be disabled using the setImplicitDownCycleTraversal method.

If aContainer is [focus traversal policy provider](http://docs.google.com/doc-files/FocusSpec.html#FocusTraversalPolicyProviders), the focus is always transferred down-cycle.

**Specified by:**[getComponentAfter](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getComponentAfter(java.awt.Container,%20java.awt.Component)) in class [FocusTraversalPolicy](http://docs.google.com/java/awt/FocusTraversalPolicy.html) **Parameters:**aContainer - a focus cycle root of aComponent or a focus traversal policy provideraComponent - a (possibly indirect) child of aContainer, or aContainer itself **Returns:**the Component that should receive the focus after aComponent, or null if no suitable Component can be found **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if aContainer is not a focus cycle root of aComponent or focus traversal policy provider, or if either aContainer or aComponent is null

### getComponentBefore

public [Component](http://docs.google.com/java/awt/Component.html) **getComponentBefore**([Container](http://docs.google.com/java/awt/Container.html) aContainer,  
 [Component](http://docs.google.com/java/awt/Component.html) aComponent)

Returns the Component that should receive the focus before aComponent. aContainer must be a focus cycle root of aComponent or a [focus traversal policy provider](http://docs.google.com/doc-files/FocusSpec.html#FocusTraversalPolicyProviders).

**Specified by:**[getComponentBefore](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getComponentBefore(java.awt.Container,%20java.awt.Component)) in class [FocusTraversalPolicy](http://docs.google.com/java/awt/FocusTraversalPolicy.html) **Parameters:**aContainer - a focus cycle root of aComponent or focus traversal policy provideraComponent - a (possibly indirect) child of aContainer, or aContainer itself **Returns:**the Component that should receive the focus before aComponent, or null if no suitable Component can be found **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if aContainer is not a focus cycle root of aComponent or focus traversal policy provider, or if either aContainer or aComponent is null

### getFirstComponent

public [Component](http://docs.google.com/java/awt/Component.html) **getFirstComponent**([Container](http://docs.google.com/java/awt/Container.html) aContainer)

Returns the first Component in the traversal cycle. This method is used to determine the next Component to focus when traversal wraps in the forward direction.

**Specified by:**[getFirstComponent](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getFirstComponent(java.awt.Container)) in class [FocusTraversalPolicy](http://docs.google.com/java/awt/FocusTraversalPolicy.html) **Parameters:**aContainer - the focus cycle root or focus traversal policy provider whose first Component is to be returned **Returns:**the first Component in the traversal cycle of aContainer, or null if no suitable Component can be found **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if aContainer is null

### getLastComponent

public [Component](http://docs.google.com/java/awt/Component.html) **getLastComponent**([Container](http://docs.google.com/java/awt/Container.html) aContainer)

Returns the last Component in the traversal cycle. This method is used to determine the next Component to focus when traversal wraps in the reverse direction.

**Specified by:**[getLastComponent](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getLastComponent(java.awt.Container)) in class [FocusTraversalPolicy](http://docs.google.com/java/awt/FocusTraversalPolicy.html) **Parameters:**aContainer - the focus cycle root or focus traversal policy provider whose last Component is to be returned **Returns:**the last Component in the traversal cycle of aContainer, or null if no suitable Component can be found **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if aContainer is null

### getDefaultComponent

public [Component](http://docs.google.com/java/awt/Component.html) **getDefaultComponent**([Container](http://docs.google.com/java/awt/Container.html) aContainer)

Returns the default Component to focus. This Component will be the first to receive focus when traversing down into a new focus traversal cycle rooted at aContainer. The default implementation of this method returns the same Component as getFirstComponent.

**Specified by:**[getDefaultComponent](http://docs.google.com/java/awt/FocusTraversalPolicy.html#getDefaultComponent(java.awt.Container)) in class [FocusTraversalPolicy](http://docs.google.com/java/awt/FocusTraversalPolicy.html) **Parameters:**aContainer - the focus cycle root or focus traversal policy provider whose default Component is to be returned **Returns:**the default Component in the traversal cycle of aContainer, or null if no suitable Component can be found **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if aContainer is null**See Also:**[getFirstComponent(java.awt.Container)](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getFirstComponent(java.awt.Container))

### setImplicitDownCycleTraversal

public void **setImplicitDownCycleTraversal**(boolean implicitDownCycleTraversal)

Sets whether this ContainerOrderFocusTraversalPolicy transfers focus down-cycle implicitly. If true, during normal forward focus traversal, the Component traversed after a focus cycle root will be the focus-cycle-root's default Component to focus. If false, the next Component in the focus traversal cycle rooted at the specified focus cycle root will be traversed instead. The default value for this property is true.

**Parameters:**implicitDownCycleTraversal - whether this ContainerOrderFocusTraversalPolicy transfers focus down-cycle implicitly**See Also:**[getImplicitDownCycleTraversal()](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getImplicitDownCycleTraversal()), [getFirstComponent(java.awt.Container)](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getFirstComponent(java.awt.Container))

### getImplicitDownCycleTraversal

public boolean **getImplicitDownCycleTraversal**()

Returns whether this ContainerOrderFocusTraversalPolicy transfers focus down-cycle implicitly. If true, during normal forward focus traversal, the Component traversed after a focus cycle root will be the focus-cycle-root's default Component to focus. If false, the next Component in the focus traversal cycle rooted at the specified focus cycle root will be traversed instead.

**Returns:**whether this ContainerOrderFocusTraversalPolicy transfers focus down-cycle implicitly**See Also:**[setImplicitDownCycleTraversal(boolean)](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#setImplicitDownCycleTraversal(boolean)), [getFirstComponent(java.awt.Container)](http://docs.google.com/java/awt/ContainerOrderFocusTraversalPolicy.html#getFirstComponent(java.awt.Container))

### accept

protected boolean **accept**([Component](http://docs.google.com/java/awt/Component.html) aComponent)

Determines whether a Component is an acceptable choice as the new focus owner. By default, this method will accept a Component if and only if it is visible, displayable, enabled, and focusable.

**Parameters:**aComponent - the Component whose fitness as a focus owner is to be tested **Returns:**true if aComponent is visible, displayable, enabled, and focusable; false otherwise

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ContainerOrderFocusTraversalPolicy.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/Container.AccessibleAWTContainer.AccessibleContainerHandler.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/Cursor.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/ContainerOrderFocusTraversalPolicy.html)    [**NO FRAMES**](http://docs.google.com/ContainerOrderFocusTraversalPolicy.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [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).