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

## **java.awt**

Class GraphicsConfiguration

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

public abstract class **GraphicsConfiguration**extends [Object](http://docs.google.com/java/lang/Object.html)

The GraphicsConfiguration class describes the characteristics of a graphics destination such as a printer or monitor. There can be many GraphicsConfiguration objects associated with a single graphics device, representing different drawing modes or capabilities. The corresponding native structure will vary from platform to platform. For example, on X11 windowing systems, each visual is a different GraphicsConfiguration. On Microsoft Windows, GraphicsConfigurations represent PixelFormats available in the current resolution and color depth.

In a virtual device multi-screen environment in which the desktop area could span multiple physical screen devices, the bounds of the GraphicsConfiguration objects are relative to the virtual coordinate system. When setting the location of a component, use [getBounds](http://docs.google.com/java/awt/GraphicsConfiguration.html#getBounds()) to get the bounds of the desired GraphicsConfiguration and offset the location with the coordinates of the GraphicsConfiguration, as the following code sample illustrates:

Frame f = new Frame(gc); // where gc is a GraphicsConfiguration  
 Rectangle bounds = gc.getBounds();  
 f.setLocation(10 + bounds.x, 10 + bounds.y);

To determine if your environment is a virtual device environment, call getBounds on all of the GraphicsConfiguration objects in your system. If any of the origins of the returned bounds is not (0, 0), your environment is a virtual device environment.

You can also use getBounds to determine the bounds of the virtual device. To do this, first call getBounds on all of the GraphicsConfiguration objects in your system. Then calculate the union of all of the bounds returned from the calls to getBounds. The union is the bounds of the virtual device. The following code sample calculates the bounds of the virtual device.

Rectangle virtualBounds = new Rectangle();  
 GraphicsEnvironment ge = GraphicsEnvironment.  
 getLocalGraphicsEnvironment();  
 GraphicsDevice[] gs =  
 ge.getScreenDevices();  
 for (int j = 0; j < gs.length; j++) {   
 GraphicsDevice gd = gs[j];  
 GraphicsConfiguration[] gc =  
 gd.getConfigurations();  
 for (int i=0; i < gc.length; i++) {  
 virtualBounds =  
 virtualBounds.union(gc[i].getBounds());  
 }  
 }

**See Also:**[Window](http://docs.google.com/java/awt/Window.html), [Frame](http://docs.google.com/java/awt/Frame.html), [GraphicsEnvironment](http://docs.google.com/java/awt/GraphicsEnvironment.html), [GraphicsDevice](http://docs.google.com/java/awt/GraphicsDevice.html)

| **Constructor Summary** | |
| --- | --- |
| protected | [**GraphicsConfiguration**](http://docs.google.com/java/awt/GraphicsConfiguration.html#GraphicsConfiguration())()            This is an abstract class that cannot be instantiated directly. |

| **Method Summary** | |
| --- | --- |
| abstract  [BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) | [**createCompatibleImage**](http://docs.google.com/java/awt/GraphicsConfiguration.html#createCompatibleImage(int,%20int))(int width, int height)            Returns a [BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) with a data layout and color model compatible with this GraphicsConfiguration. |
| [BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) | [**createCompatibleImage**](http://docs.google.com/java/awt/GraphicsConfiguration.html#createCompatibleImage(int,%20int,%20int))(int width, int height, int transparency)            Returns a BufferedImage that supports the specified transparency and has a data layout and color model compatible with this GraphicsConfiguration. |
| [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) | [**createCompatibleVolatileImage**](http://docs.google.com/java/awt/GraphicsConfiguration.html#createCompatibleVolatileImage(int,%20int))(int width, int height)            Returns a [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) with a data layout and color model compatible with this GraphicsConfiguration. |
| [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) | [**createCompatibleVolatileImage**](http://docs.google.com/java/awt/GraphicsConfiguration.html#createCompatibleVolatileImage(int,%20int,%20java.awt.ImageCapabilities))(int width, int height, [ImageCapabilities](http://docs.google.com/java/awt/ImageCapabilities.html) caps)            Returns a [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) with a data layout and color model compatible with this GraphicsConfiguration, using the specified image capabilities. |
| [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) | [**createCompatibleVolatileImage**](http://docs.google.com/java/awt/GraphicsConfiguration.html#createCompatibleVolatileImage(int,%20int,%20java.awt.ImageCapabilities,%20int))(int width, int height, [ImageCapabilities](http://docs.google.com/java/awt/ImageCapabilities.html) caps, int transparency)            Returns a [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) with a data layout and color model compatible with this GraphicsConfiguration, using the specified image capabilities and transparency value. |
| [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) | [**createCompatibleVolatileImage**](http://docs.google.com/java/awt/GraphicsConfiguration.html#createCompatibleVolatileImage(int,%20int,%20int))(int width, int height, int transparency)            Returns a [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) with a data layout and color model compatible with this GraphicsConfiguration. |
| abstract  [Rectangle](http://docs.google.com/java/awt/Rectangle.html) | [**getBounds**](http://docs.google.com/java/awt/GraphicsConfiguration.html#getBounds())()            Returns the bounds of the GraphicsConfiguration in the device coordinates. |
| [BufferCapabilities](http://docs.google.com/java/awt/BufferCapabilities.html) | [**getBufferCapabilities**](http://docs.google.com/java/awt/GraphicsConfiguration.html#getBufferCapabilities())()            Returns the buffering capabilities of this GraphicsConfiguration. |
| abstract  [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) | [**getColorModel**](http://docs.google.com/java/awt/GraphicsConfiguration.html#getColorModel())()            Returns the [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) associated with this GraphicsConfiguration. |
| abstract  [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) | [**getColorModel**](http://docs.google.com/java/awt/GraphicsConfiguration.html#getColorModel(int))(int transparency)            Returns the ColorModel associated with this GraphicsConfiguration that supports the specified transparency. |
| abstract  [AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) | [**getDefaultTransform**](http://docs.google.com/java/awt/GraphicsConfiguration.html#getDefaultTransform())()            Returns the default [AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) for this GraphicsConfiguration. |
| abstract  [GraphicsDevice](http://docs.google.com/java/awt/GraphicsDevice.html) | [**getDevice**](http://docs.google.com/java/awt/GraphicsConfiguration.html#getDevice())()            Returns the [GraphicsDevice](http://docs.google.com/java/awt/GraphicsDevice.html) associated with this GraphicsConfiguration. |
| [ImageCapabilities](http://docs.google.com/java/awt/ImageCapabilities.html) | [**getImageCapabilities**](http://docs.google.com/java/awt/GraphicsConfiguration.html#getImageCapabilities())()            Returns the image capabilities of this GraphicsConfiguration. |
| abstract  [AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) | [**getNormalizingTransform**](http://docs.google.com/java/awt/GraphicsConfiguration.html#getNormalizingTransform())()            Returns a AffineTransform that can be concatenated with the default AffineTransform of a GraphicsConfiguration so that 72 units in user space equals 1 inch in device space. |

| **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** |
| --- |

### GraphicsConfiguration

protected **GraphicsConfiguration**()

This is an abstract class that cannot be instantiated directly. Instances must be obtained from a suitable factory or query method.

**See Also:**[GraphicsDevice.getConfigurations()](http://docs.google.com/java/awt/GraphicsDevice.html#getConfigurations()), [GraphicsDevice.getDefaultConfiguration()](http://docs.google.com/java/awt/GraphicsDevice.html#getDefaultConfiguration()), [GraphicsDevice.getBestConfiguration(java.awt.GraphicsConfigTemplate)](http://docs.google.com/java/awt/GraphicsDevice.html#getBestConfiguration(java.awt.GraphicsConfigTemplate)), [Graphics2D.getDeviceConfiguration()](http://docs.google.com/java/awt/Graphics2D.html#getDeviceConfiguration())

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

### getDevice

public abstract [GraphicsDevice](http://docs.google.com/java/awt/GraphicsDevice.html) **getDevice**()

Returns the [GraphicsDevice](http://docs.google.com/java/awt/GraphicsDevice.html) associated with this GraphicsConfiguration.

**Returns:**a GraphicsDevice object that is associated with this GraphicsConfiguration.

### createCompatibleImage

public abstract [BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) **createCompatibleImage**(int width,  
 int height)

Returns a [BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) with a data layout and color model compatible with this GraphicsConfiguration. This method has nothing to do with memory-mapping a device. The returned BufferedImage has a layout and color model that is closest to this native device configuration and can therefore be optimally blitted to this device.

**Parameters:**width - the width of the returned BufferedImageheight - the height of the returned BufferedImage **Returns:**a BufferedImage whose data layout and color model is compatible with this GraphicsConfiguration.

### createCompatibleImage

public [BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) **createCompatibleImage**(int width,  
 int height,  
 int transparency)

Returns a BufferedImage that supports the specified transparency and has a data layout and color model compatible with this GraphicsConfiguration. This method has nothing to do with memory-mapping a device. The returned BufferedImage has a layout and color model that can be optimally blitted to a device with this GraphicsConfiguration.

**Parameters:**width - the width of the returned BufferedImageheight - the height of the returned BufferedImagetransparency - the specified transparency mode **Returns:**a BufferedImage whose data layout and color model is compatible with this GraphicsConfiguration and also supports the specified transparency. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the transparency is not a valid value**See Also:**[Transparency.OPAQUE](http://docs.google.com/java/awt/Transparency.html#OPAQUE), [Transparency.BITMASK](http://docs.google.com/java/awt/Transparency.html#BITMASK), [Transparency.TRANSLUCENT](http://docs.google.com/java/awt/Transparency.html#TRANSLUCENT)

### createCompatibleVolatileImage

public [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) **createCompatibleVolatileImage**(int width,  
 int height)

Returns a [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) with a data layout and color model compatible with this GraphicsConfiguration. The returned VolatileImage may have data that is stored optimally for the underlying graphics device and may therefore benefit from platform-specific rendering acceleration.

**Parameters:**width - the width of the returned VolatileImageheight - the height of the returned VolatileImage **Returns:**a VolatileImage whose data layout and color model is compatible with this GraphicsConfiguration.**Since:** 1.4 **See Also:**[Component.createVolatileImage(int, int)](http://docs.google.com/java/awt/Component.html#createVolatileImage(int,%20int))

### createCompatibleVolatileImage

public [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) **createCompatibleVolatileImage**(int width,  
 int height,  
 int transparency)

Returns a [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) with a data layout and color model compatible with this GraphicsConfiguration. The returned VolatileImage may have data that is stored optimally for the underlying graphics device and may therefore benefit from platform-specific rendering acceleration.

**Parameters:**width - the width of the returned VolatileImageheight - the height of the returned VolatileImagetransparency - the specified transparency mode **Returns:**a VolatileImage whose data layout and color model is compatible with this GraphicsConfiguration. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the transparency is not a valid value**Since:** 1.5 **See Also:**[Transparency.OPAQUE](http://docs.google.com/java/awt/Transparency.html#OPAQUE), [Transparency.BITMASK](http://docs.google.com/java/awt/Transparency.html#BITMASK), [Transparency.TRANSLUCENT](http://docs.google.com/java/awt/Transparency.html#TRANSLUCENT), [Component.createVolatileImage(int, int)](http://docs.google.com/java/awt/Component.html#createVolatileImage(int,%20int))

### createCompatibleVolatileImage

public [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) **createCompatibleVolatileImage**(int width,  
 int height,  
 [ImageCapabilities](http://docs.google.com/java/awt/ImageCapabilities.html) caps)  
 throws [AWTException](http://docs.google.com/java/awt/AWTException.html)

Returns a [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) with a data layout and color model compatible with this GraphicsConfiguration, using the specified image capabilities. If the caps parameter is null, it is effectively ignored and this method will create a VolatileImage without regard to ImageCapabilities constraints. The returned VolatileImage has a layout and color model that is closest to this native device configuration and can therefore be optimally blitted to this device.

**Parameters:**width - the width of the returned VolatileImageheight - the height of the returned VolatileImagecaps - the image capabilities **Returns:**a VolatileImage whose data layout and color model is compatible with this GraphicsConfiguration. **Throws:** [AWTException](http://docs.google.com/java/awt/AWTException.html) - if the supplied image capabilities could not be met by this graphics configuration**Since:** 1.4

### createCompatibleVolatileImage

public [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) **createCompatibleVolatileImage**(int width,  
 int height,  
 [ImageCapabilities](http://docs.google.com/java/awt/ImageCapabilities.html) caps,  
 int transparency)  
 throws [AWTException](http://docs.google.com/java/awt/AWTException.html)

Returns a [VolatileImage](http://docs.google.com/java/awt/image/VolatileImage.html) with a data layout and color model compatible with this GraphicsConfiguration, using the specified image capabilities and transparency value. If the caps parameter is null, it is effectively ignored and this method will create a VolatileImage without regard to ImageCapabilities constraints. The returned VolatileImage has a layout and color model that is closest to this native device configuration and can therefore be optimally blitted to this device.

**Parameters:**width - the width of the returned VolatileImageheight - the height of the returned VolatileImagecaps - the image capabilitiestransparency - the specified transparency mode **Returns:**a VolatileImage whose data layout and color model is compatible with this GraphicsConfiguration. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the transparency is not a valid value [AWTException](http://docs.google.com/java/awt/AWTException.html) - if the supplied image capabilities could not be met by this graphics configuration**Since:** 1.5 **See Also:**[Transparency.OPAQUE](http://docs.google.com/java/awt/Transparency.html#OPAQUE), [Transparency.BITMASK](http://docs.google.com/java/awt/Transparency.html#BITMASK), [Transparency.TRANSLUCENT](http://docs.google.com/java/awt/Transparency.html#TRANSLUCENT), [Component.createVolatileImage(int, int)](http://docs.google.com/java/awt/Component.html#createVolatileImage(int,%20int))

### getColorModel

public abstract [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) **getColorModel**()

Returns the [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) associated with this GraphicsConfiguration.

**Returns:**a ColorModel object that is associated with this GraphicsConfiguration.

### getColorModel

public abstract [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) **getColorModel**(int transparency)

Returns the ColorModel associated with this GraphicsConfiguration that supports the specified transparency.

**Parameters:**transparency - the specified transparency mode **Returns:**a ColorModel object that is associated with this GraphicsConfiguration and supports the specified transparency or null if the transparency is not a valid value.**See Also:**[Transparency.OPAQUE](http://docs.google.com/java/awt/Transparency.html#OPAQUE), [Transparency.BITMASK](http://docs.google.com/java/awt/Transparency.html#BITMASK), [Transparency.TRANSLUCENT](http://docs.google.com/java/awt/Transparency.html#TRANSLUCENT)

### getDefaultTransform

public abstract [AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) **getDefaultTransform**()

Returns the default [AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) for this GraphicsConfiguration. This AffineTransform is typically the Identity transform for most normal screens. The default AffineTransform maps coordinates onto the device such that 72 user space coordinate units measure approximately 1 inch in device space. The normalizing transform can be used to make this mapping more exact. Coordinates in the coordinate space defined by the default AffineTransform for screen and printer devices have the origin in the upper left-hand corner of the target region of the device, with X coordinates increasing to the right and Y coordinates increasing downwards. For image buffers not associated with a device, such as those not created by createCompatibleImage, this AffineTransform is the Identity transform.

**Returns:**the default AffineTransform for this GraphicsConfiguration.

### getNormalizingTransform

public abstract [AffineTransform](http://docs.google.com/java/awt/geom/AffineTransform.html) **getNormalizingTransform**()

Returns a AffineTransform that can be concatenated with the default AffineTransform of a GraphicsConfiguration so that 72 units in user space equals 1 inch in device space.

For a particular [Graphics2D](http://docs.google.com/java/awt/Graphics2D.html), g, one can reset the transformation to create such a mapping by using the following pseudocode:

GraphicsConfiguration gc = g.getDeviceConfiguration();  
  
 g.setTransform(gc.getDefaultTransform());  
 g.transform(gc.getNormalizingTransform());

Note that sometimes this AffineTransform is identity, such as for printers or metafile output, and that this AffineTransform is only as accurate as the information supplied by the underlying system. For image buffers not associated with a device, such as those not created by createCompatibleImage, this AffineTransform is the Identity transform since there is no valid distance measurement.

**Returns:**an AffineTransform to concatenate to the default AffineTransform so that 72 units in user space is mapped to 1 inch in device space.

### getBounds

public abstract [Rectangle](http://docs.google.com/java/awt/Rectangle.html) **getBounds**()

Returns the bounds of the GraphicsConfiguration in the device coordinates. In a multi-screen environment with a virtual device, the bounds can have negative X or Y origins.

**Returns:**the bounds of the area covered by this GraphicsConfiguration.**Since:** 1.3

### getBufferCapabilities

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

Returns the buffering capabilities of this GraphicsConfiguration.

**Returns:**the buffering capabilities of this graphics configuration object**Since:** 1.4

### getImageCapabilities

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

Returns the image capabilities of this GraphicsConfiguration.

**Returns:**the image capabilities of this graphics configuration object**Since:** 1.4

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/GraphicsConfiguration.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/GraphicsConfigTemplate.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/GraphicsDevice.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/GraphicsConfiguration.html)    [**NO FRAMES**](http://docs.google.com/GraphicsConfiguration.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).