| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ImageFilter.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/image/ImageConsumer.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/image/ImageObserver.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/ImageFilter.html)    [**NO FRAMES**](http://docs.google.com/ImageFilter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: [FIELD](#4d34og8) | [CONSTR](#17dp8vu) | [METHOD](#26in1rg) |

## **java.awt.image**

Class ImageFilter

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

**All Implemented Interfaces:** [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html) **Direct Known Subclasses:** [BufferedImageFilter](http://docs.google.com/java/awt/image/BufferedImageFilter.html), [CropImageFilter](http://docs.google.com/java/awt/image/CropImageFilter.html), [ReplicateScaleFilter](http://docs.google.com/java/awt/image/ReplicateScaleFilter.html), [RGBImageFilter](http://docs.google.com/java/awt/image/RGBImageFilter.html)

public class **ImageFilter**extends [Object](http://docs.google.com/java/lang/Object.html)implements [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html)

This class implements a filter for the set of interface methods that are used to deliver data from an ImageProducer to an ImageConsumer. It is meant to be used in conjunction with a FilteredImageSource object to produce filtered versions of existing images. It is a base class that provides the calls needed to implement a "Null filter" which has no effect on the data being passed through. Filters should subclass this class and override the methods which deal with the data that needs to be filtered and modify it as necessary.

**See Also:**[FilteredImageSource](http://docs.google.com/java/awt/image/FilteredImageSource.html), [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html)

| **Field Summary** | |
| --- | --- |
| protected  [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) | [**consumer**](http://docs.google.com/java/awt/image/ImageFilter.html#consumer)            The consumer of the particular image data stream for which this instance of the ImageFilter is filtering data. |

| **Fields inherited from interface java.awt.image.**[**ImageConsumer**](http://docs.google.com/java/awt/image/ImageConsumer.html) |
| --- |
| [COMPLETESCANLINES](http://docs.google.com/java/awt/image/ImageConsumer.html#COMPLETESCANLINES), [IMAGEABORTED](http://docs.google.com/java/awt/image/ImageConsumer.html#IMAGEABORTED), [IMAGEERROR](http://docs.google.com/java/awt/image/ImageConsumer.html#IMAGEERROR), [RANDOMPIXELORDER](http://docs.google.com/java/awt/image/ImageConsumer.html#RANDOMPIXELORDER), [SINGLEFRAME](http://docs.google.com/java/awt/image/ImageConsumer.html#SINGLEFRAME), [SINGLEFRAMEDONE](http://docs.google.com/java/awt/image/ImageConsumer.html#SINGLEFRAMEDONE), [SINGLEPASS](http://docs.google.com/java/awt/image/ImageConsumer.html#SINGLEPASS), [STATICIMAGEDONE](http://docs.google.com/java/awt/image/ImageConsumer.html#STATICIMAGEDONE), [TOPDOWNLEFTRIGHT](http://docs.google.com/java/awt/image/ImageConsumer.html#TOPDOWNLEFTRIGHT) |

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

| **Method Summary** | |
| --- | --- |
| [Object](http://docs.google.com/java/lang/Object.html) | [**clone**](http://docs.google.com/java/awt/image/ImageFilter.html#clone())()            Clones this object. |
| [ImageFilter](http://docs.google.com/java/awt/image/ImageFilter.html) | [**getFilterInstance**](http://docs.google.com/java/awt/image/ImageFilter.html#getFilterInstance(java.awt.image.ImageConsumer))([ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) ic)            Returns a unique instance of an ImageFilter object which will actually perform the filtering for the specified ImageConsumer. |
| void | [**imageComplete**](http://docs.google.com/java/awt/image/ImageFilter.html#imageComplete(int))(int status)            Filters the information provided in the imageComplete method of the ImageConsumer interface. |
| void | [**resendTopDownLeftRight**](http://docs.google.com/java/awt/image/ImageFilter.html#resendTopDownLeftRight(java.awt.image.ImageProducer))([ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html) ip)            Responds to a request for a TopDownLeftRight (TDLR) ordered resend of the pixel data from an ImageConsumer. |
| void | [**setColorModel**](http://docs.google.com/java/awt/image/ImageFilter.html#setColorModel(java.awt.image.ColorModel))([ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model)            Filter the information provided in the setColorModel method of the ImageConsumer interface. |
| void | [**setDimensions**](http://docs.google.com/java/awt/image/ImageFilter.html#setDimensions(int,%20int))(int width, int height)            Filters the information provided in the setDimensions method of the ImageConsumer interface. |
| void | [**setHints**](http://docs.google.com/java/awt/image/ImageFilter.html#setHints(int))(int hints)            Filters the information provided in the setHints method of the ImageConsumer interface. |
| void | [**setPixels**](http://docs.google.com/java/awt/image/ImageFilter.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int))(int x, int y, int w, int h, [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model, byte[] pixels, int off, int scansize)            Filters the information provided in the setPixels method of the ImageConsumer interface which takes an array of bytes. |
| void | [**setPixels**](http://docs.google.com/java/awt/image/ImageFilter.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20int%5B%5D,%20int,%20int))(int x, int y, int w, int h, [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model, int[] pixels, int off, int scansize)            Filters the information provided in the setPixels method of the ImageConsumer interface which takes an array of integers. |
| void | [**setProperties**](http://docs.google.com/java/awt/image/ImageFilter.html#setProperties(java.util.Hashtable))([Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> props)            Passes the properties from the source object along after adding a property indicating the stream of filters it has been run through. |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [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)) |

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

### consumer

protected [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **consumer**

The consumer of the particular image data stream for which this instance of the ImageFilter is filtering data. It is not initialized during the constructor, but rather during the getFilterInstance() method call when the FilteredImageSource is creating a unique instance of this object for a particular image data stream.

**See Also:**[getFilterInstance(java.awt.image.ImageConsumer)](http://docs.google.com/java/awt/image/ImageFilter.html#getFilterInstance(java.awt.image.ImageConsumer)), [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html)

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

### ImageFilter

public **ImageFilter**()

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

### getFilterInstance

public [ImageFilter](http://docs.google.com/java/awt/image/ImageFilter.html) **getFilterInstance**([ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) ic)

Returns a unique instance of an ImageFilter object which will actually perform the filtering for the specified ImageConsumer. The default implementation just clones this object.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Parameters:**ic - the specified ImageConsumer **Returns:**an ImageFilter used to perform the filtering for the specified ImageConsumer.

### setDimensions

public void **setDimensions**(int width,  
 int height)

Filters the information provided in the setDimensions method of the ImageConsumer interface.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[setDimensions](http://docs.google.com/java/awt/image/ImageConsumer.html#setDimensions(int,%20int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**width - the width of the source imageheight - the height of the source image**See Also:**[ImageConsumer.setDimensions(int, int)](http://docs.google.com/java/awt/image/ImageConsumer.html#setDimensions(int,%20int))

### setProperties

public void **setProperties**([Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> props)

Passes the properties from the source object along after adding a property indicating the stream of filters it has been run through.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[setProperties](http://docs.google.com/java/awt/image/ImageConsumer.html#setProperties(java.util.Hashtable)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**props - the properties from the source object **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if props is null

### setColorModel

public void **setColorModel**([ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model)

Filter the information provided in the setColorModel method of the ImageConsumer interface.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[setColorModel](http://docs.google.com/java/awt/image/ImageConsumer.html#setColorModel(java.awt.image.ColorModel)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**model - the specified ColorModel**See Also:**[ImageConsumer.setColorModel(java.awt.image.ColorModel)](http://docs.google.com/java/awt/image/ImageConsumer.html#setColorModel(java.awt.image.ColorModel))

### setHints

public void **setHints**(int hints)

Filters the information provided in the setHints method of the ImageConsumer interface.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[setHints](http://docs.google.com/java/awt/image/ImageConsumer.html#setHints(int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**hints - a set of hints that the ImageConsumer uses to process the pixels**See Also:**[ImageConsumer.setHints(int)](http://docs.google.com/java/awt/image/ImageConsumer.html#setHints(int))

### setPixels

public void **setPixels**(int x,  
 int y,  
 int w,  
 int h,  
 [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model,  
 byte[] pixels,  
 int off,  
 int scansize)

Filters the information provided in the setPixels method of the ImageConsumer interface which takes an array of bytes.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[setPixels](http://docs.google.com/java/awt/image/ImageConsumer.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**x - the X coordinate of the upper-left corner of the area of pixels to be sety - the Y coordinate of the upper-left corner of the area of pixels to be setw - the width of the area of pixelsh - the height of the area of pixelsmodel - the specified ColorModelpixels - the array of pixelsoff - the offset into the pixels arrayscansize - the distance from one row of pixels to the next in the pixels array**See Also:**[ImageConsumer.setPixels(int, int, int, int, java.awt.image.ColorModel, byte[], int, int)](http://docs.google.com/java/awt/image/ImageConsumer.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int))

### setPixels

public void **setPixels**(int x,  
 int y,  
 int w,  
 int h,  
 [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model,  
 int[] pixels,  
 int off,  
 int scansize)

Filters the information provided in the setPixels method of the ImageConsumer interface which takes an array of integers.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[setPixels](http://docs.google.com/java/awt/image/ImageConsumer.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20int%5B%5D,%20int,%20int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**x - the X coordinate of the upper-left corner of the area of pixels to be sety - the Y coordinate of the upper-left corner of the area of pixels to be setw - the width of the area of pixelsh - the height of the area of pixelsmodel - the specified ColorModelpixels - the array of pixelsoff - the offset into the pixels arrayscansize - the distance from one row of pixels to the next in the pixels array**See Also:**[ImageConsumer.setPixels(int, int, int, int, java.awt.image.ColorModel, byte[], int, int)](http://docs.google.com/java/awt/image/ImageConsumer.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int))

### imageComplete

public void **imageComplete**(int status)

Filters the information provided in the imageComplete method of the ImageConsumer interface.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[imageComplete](http://docs.google.com/java/awt/image/ImageConsumer.html#imageComplete(int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) **Parameters:**status - the status of image loading**See Also:**[ImageConsumer.imageComplete(int)](http://docs.google.com/java/awt/image/ImageConsumer.html#imageComplete(int))

### resendTopDownLeftRight

public void **resendTopDownLeftRight**([ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html) ip)

Responds to a request for a TopDownLeftRight (TDLR) ordered resend of the pixel data from an ImageConsumer. When an ImageConsumer being fed by an instance of this ImageFilter requests a resend of the data in TDLR order, the FilteredImageSource invokes this method of the ImageFilter.

An ImageFilter subclass might override this method or not, depending on if and how it can send data in TDLR order. Three possibilities exist:

* Do not override this method. This makes the subclass use the default implementation, which is to forward the request to the indicated ImageProducer using this filter as the requesting ImageConsumer. This behavior is appropriate if the filter can determine that it will forward the pixels in TDLR order if its upstream producer object sends them in TDLR order.
* Override the method to simply send the data. This is appropriate if the filter can handle the request itself — for example, if the generated pixels have been saved in some sort of buffer.
* Override the method to do nothing. This is appropriate if the filter cannot produce filtered data in TDLR order.

**Parameters:**ip - the ImageProducer that is feeding this instance of the filter - also the ImageProducer that the request should be forwarded to if necessary **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if ip is null**See Also:**[ImageProducer.requestTopDownLeftRightResend(java.awt.image.ImageConsumer)](http://docs.google.com/java/awt/image/ImageProducer.html#requestTopDownLeftRightResend(java.awt.image.ImageConsumer))

### clone

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

Clones this object.

**Overrides:**[clone](http://docs.google.com/java/lang/Object.html#clone()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a clone of this instance.**See Also:**[Cloneable](http://docs.google.com/java/lang/Cloneable.html)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ImageFilter.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/image/ImageConsumer.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/image/ImageObserver.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/ImageFilter.html)    [**NO FRAMES**](http://docs.google.com/ImageFilter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: [FIELD](#4d34og8) | [CONSTR](#17dp8vu) | [METHOD](#26in1rg) |

[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).