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

Class RenderableImageProducer

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

**All Implemented Interfaces:** [ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html), [Runnable](http://docs.google.com/java/lang/Runnable.html)

public class **RenderableImageProducer**extends [Object](http://docs.google.com/java/lang/Object.html)implements [ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html), [Runnable](http://docs.google.com/java/lang/Runnable.html)

An adapter class that implements ImageProducer to allow the asynchronous production of a RenderableImage. The size of the ImageConsumer is determined by the scale factor of the usr2dev transform in the RenderContext. If the RenderContext is null, the default rendering of the RenderableImage is used. This class implements an asynchronous production that produces the image in one thread at one resolution. This class may be subclassed to implement versions that will render the image using several threads. These threads could render either the same image at progressively better quality, or different sections of the image at a single resolution.

| **Constructor Summary** | |
| --- | --- |
| [**RenderableImageProducer**](http://docs.google.com/java/awt/image/renderable/RenderableImageProducer.html#RenderableImageProducer(java.awt.image.renderable.RenderableImage,%20java.awt.image.renderable.RenderContext))([RenderableImage](http://docs.google.com/java/awt/image/renderable/RenderableImage.html) rdblImage, [RenderContext](http://docs.google.com/java/awt/image/renderable/RenderContext.html) rc)            Constructs a new RenderableImageProducer from a RenderableImage and a RenderContext. |

| **Method Summary** | |
| --- | --- |
| void | [**addConsumer**](http://docs.google.com/java/awt/image/renderable/RenderableImageProducer.html#addConsumer(java.awt.image.ImageConsumer))([ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) ic)            Adds an ImageConsumer to the list of consumers interested in data for this image. |
| boolean | [**isConsumer**](http://docs.google.com/java/awt/image/renderable/RenderableImageProducer.html#isConsumer(java.awt.image.ImageConsumer))([ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) ic)            Determine if an ImageConsumer is on the list of consumers currently interested in data for this image. |
| void | [**removeConsumer**](http://docs.google.com/java/awt/image/renderable/RenderableImageProducer.html#removeConsumer(java.awt.image.ImageConsumer))([ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) ic)            Remove an ImageConsumer from the list of consumers interested in data for this image. |
| void | [**requestTopDownLeftRightResend**](http://docs.google.com/java/awt/image/renderable/RenderableImageProducer.html#requestTopDownLeftRightResend(java.awt.image.ImageConsumer))([ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) ic)            Requests that a given ImageConsumer have the image data delivered one more time in top-down, left-right order. |
| void | [**run**](http://docs.google.com/java/awt/image/renderable/RenderableImageProducer.html#run())()            The runnable method for this class. |
| void | [**setRenderContext**](http://docs.google.com/java/awt/image/renderable/RenderableImageProducer.html#setRenderContext(java.awt.image.renderable.RenderContext))([RenderContext](http://docs.google.com/java/awt/image/renderable/RenderContext.html) rc)            Sets a new RenderContext to use for the next startProduction() call. |
| void | [**startProduction**](http://docs.google.com/java/awt/image/renderable/RenderableImageProducer.html#startProduction(java.awt.image.ImageConsumer))([ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html) ic)            Adds an ImageConsumer to the list of consumers interested in data for this image, and immediately starts delivery of the image data through the ImageConsumer interface. |

| **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** |
| --- |

### RenderableImageProducer

public **RenderableImageProducer**([RenderableImage](http://docs.google.com/java/awt/image/renderable/RenderableImage.html) rdblImage,  
 [RenderContext](http://docs.google.com/java/awt/image/renderable/RenderContext.html) rc)

Constructs a new RenderableImageProducer from a RenderableImage and a RenderContext.

**Parameters:**rdblImage - the RenderableImage to be rendered.rc - the RenderContext to use for producing the pixels.

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

### setRenderContext

public void **setRenderContext**([RenderContext](http://docs.google.com/java/awt/image/renderable/RenderContext.html) rc)

Sets a new RenderContext to use for the next startProduction() call.

**Parameters:**rc - the new RenderContext.

### addConsumer

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

Adds an ImageConsumer to the list of consumers interested in data for this image.

**Specified by:**[addConsumer](http://docs.google.com/java/awt/image/ImageProducer.html#addConsumer(java.awt.image.ImageConsumer)) in interface [ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html) **Parameters:**ic - an ImageConsumer to be added to the interest list.**See Also:**[ImageProducer.startProduction(java.awt.image.ImageConsumer)](http://docs.google.com/java/awt/image/ImageProducer.html#startProduction(java.awt.image.ImageConsumer))

### isConsumer

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

Determine if an ImageConsumer is on the list of consumers currently interested in data for this image.

**Specified by:**[isConsumer](http://docs.google.com/java/awt/image/ImageProducer.html#isConsumer(java.awt.image.ImageConsumer)) in interface [ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html) **Parameters:**ic - the ImageConsumer to be checked. **Returns:**true if the ImageConsumer is on the list; false otherwise.

### removeConsumer

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

Remove an ImageConsumer from the list of consumers interested in data for this image.

**Specified by:**[removeConsumer](http://docs.google.com/java/awt/image/ImageProducer.html#removeConsumer(java.awt.image.ImageConsumer)) in interface [ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html) **Parameters:**ic - the ImageConsumer to be removed.

### startProduction

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

Adds an ImageConsumer to the list of consumers interested in data for this image, and immediately starts delivery of the image data through the ImageConsumer interface.

**Specified by:**[startProduction](http://docs.google.com/java/awt/image/ImageProducer.html#startProduction(java.awt.image.ImageConsumer)) in interface [ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html) **Parameters:**ic - the ImageConsumer to be added to the list of consumers.**See Also:**[ImageProducer.addConsumer(java.awt.image.ImageConsumer)](http://docs.google.com/java/awt/image/ImageProducer.html#addConsumer(java.awt.image.ImageConsumer))

### requestTopDownLeftRightResend

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

Requests that a given ImageConsumer have the image data delivered one more time in top-down, left-right order.

**Specified by:**[requestTopDownLeftRightResend](http://docs.google.com/java/awt/image/ImageProducer.html#requestTopDownLeftRightResend(java.awt.image.ImageConsumer)) in interface [ImageProducer](http://docs.google.com/java/awt/image/ImageProducer.html) **Parameters:**ic - the ImageConsumer requesting the resend.**See Also:**[ImageConsumer.setHints(int)](http://docs.google.com/java/awt/image/ImageConsumer.html#setHints(int))

### run

public void **run**()

The runnable method for this class. This will produce an image using the current RenderableImage and RenderContext and send it to all the ImageConsumer currently registered with this class.

**Specified by:**[run](http://docs.google.com/java/lang/Runnable.html#run()) in interface [Runnable](http://docs.google.com/java/lang/Runnable.html) **See Also:**[Thread.run()](http://docs.google.com/java/lang/Thread.html#run())

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