| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/WritableRaster.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/VolatileImage.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/image/WritableRenderedImage.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/WritableRaster.html)    [**NO FRAMES**](http://docs.google.com/WritableRaster.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#lnxbz9) |

## **java.awt.image**

Class WritableRaster

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

public class **WritableRaster**extends [Raster](http://docs.google.com/java/awt/image/Raster.html)

This class extends Raster to provide pixel writing capabilities. Refer to the class comment for Raster for descriptions of how a Raster stores pixels.

The constructors of this class are protected. To instantiate a WritableRaster, use one of the createWritableRaster factory methods in the Raster class.

| **Field Summary** | |
| --- | --- |

| **Fields inherited from class java.awt.image.**[**Raster**](http://docs.google.com/java/awt/image/Raster.html) |
| --- |
| [dataBuffer](http://docs.google.com/java/awt/image/Raster.html#dataBuffer), [height](http://docs.google.com/java/awt/image/Raster.html#height), [minX](http://docs.google.com/java/awt/image/Raster.html#minX), [minY](http://docs.google.com/java/awt/image/Raster.html#minY), [numBands](http://docs.google.com/java/awt/image/Raster.html#numBands), [numDataElements](http://docs.google.com/java/awt/image/Raster.html#numDataElements), [parent](http://docs.google.com/java/awt/image/Raster.html#parent), [sampleModel](http://docs.google.com/java/awt/image/Raster.html#sampleModel), [sampleModelTranslateX](http://docs.google.com/java/awt/image/Raster.html#sampleModelTranslateX), [sampleModelTranslateY](http://docs.google.com/java/awt/image/Raster.html#sampleModelTranslateY), [width](http://docs.google.com/java/awt/image/Raster.html#width) |

| **Constructor Summary** | |
| --- | --- |
| protected | [**WritableRaster**](http://docs.google.com/java/awt/image/WritableRaster.html#WritableRaster(java.awt.image.SampleModel,%20java.awt.image.DataBuffer,%20java.awt.Point))([SampleModel](http://docs.google.com/java/awt/image/SampleModel.html) sampleModel, [DataBuffer](http://docs.google.com/java/awt/image/DataBuffer.html) dataBuffer, [Point](http://docs.google.com/java/awt/Point.html) origin)            Constructs a WritableRaster with the given SampleModel and DataBuffer. |
| protected | [**WritableRaster**](http://docs.google.com/java/awt/image/WritableRaster.html#WritableRaster(java.awt.image.SampleModel,%20java.awt.image.DataBuffer,%20java.awt.Rectangle,%20java.awt.Point,%20java.awt.image.WritableRaster))([SampleModel](http://docs.google.com/java/awt/image/SampleModel.html) sampleModel, [DataBuffer](http://docs.google.com/java/awt/image/DataBuffer.html) dataBuffer, [Rectangle](http://docs.google.com/java/awt/Rectangle.html) aRegion, [Point](http://docs.google.com/java/awt/Point.html) sampleModelTranslate, [WritableRaster](http://docs.google.com/java/awt/image/WritableRaster.html) parent)            Constructs a WritableRaster with the given SampleModel, DataBuffer, and parent. |
| protected | [**WritableRaster**](http://docs.google.com/java/awt/image/WritableRaster.html#WritableRaster(java.awt.image.SampleModel,%20java.awt.Point))([SampleModel](http://docs.google.com/java/awt/image/SampleModel.html) sampleModel, [Point](http://docs.google.com/java/awt/Point.html) origin)            Constructs a WritableRaster with the given SampleModel. |

| **Method Summary** | |
| --- | --- |
| [WritableRaster](http://docs.google.com/java/awt/image/WritableRaster.html) | [**createWritableChild**](http://docs.google.com/java/awt/image/WritableRaster.html#createWritableChild(int,%20int,%20int,%20int,%20int,%20int,%20int%5B%5D))(int parentX, int parentY, int w, int h, int childMinX, int childMinY, int[] bandList)            Returns a new WritableRaster which shares all or part of this WritableRaster's DataBuffer. |
| [WritableRaster](http://docs.google.com/java/awt/image/WritableRaster.html) | [**createWritableTranslatedChild**](http://docs.google.com/java/awt/image/WritableRaster.html#createWritableTranslatedChild(int,%20int))(int childMinX, int childMinY)            Create a WritableRaster with the same size, SampleModel and DataBuffer as this one, but with a different location. |
| [WritableRaster](http://docs.google.com/java/awt/image/WritableRaster.html) | [**getWritableParent**](http://docs.google.com/java/awt/image/WritableRaster.html#getWritableParent())()            Returns the parent WritableRaster (if any) of this WritableRaster, or else null. |
| void | [**setDataElements**](http://docs.google.com/java/awt/image/WritableRaster.html#setDataElements(int,%20int,%20int,%20int,%20java.lang.Object))(int x, int y, int w, int h, [Object](http://docs.google.com/java/lang/Object.html) inData)            Sets the data for a rectangle of pixels from a primitive array of type TransferType. |
| void | [**setDataElements**](http://docs.google.com/java/awt/image/WritableRaster.html#setDataElements(int,%20int,%20java.lang.Object))(int x, int y, [Object](http://docs.google.com/java/lang/Object.html) inData)            Sets the data for a single pixel from a primitive array of type TransferType. |
| void | [**setDataElements**](http://docs.google.com/java/awt/image/WritableRaster.html#setDataElements(int,%20int,%20java.awt.image.Raster))(int x, int y, [Raster](http://docs.google.com/java/awt/image/Raster.html) inRaster)            Sets the data for a rectangle of pixels from an input Raster. |
| void | [**setPixel**](http://docs.google.com/java/awt/image/WritableRaster.html#setPixel(int,%20int,%20double%5B%5D))(int x, int y, double[] dArray)            Sets a pixel in the DataBuffer using a double array of samples for input. |
| void | [**setPixel**](http://docs.google.com/java/awt/image/WritableRaster.html#setPixel(int,%20int,%20float%5B%5D))(int x, int y, float[] fArray)            Sets a pixel in the DataBuffer using a float array of samples for input. |
| void | [**setPixel**](http://docs.google.com/java/awt/image/WritableRaster.html#setPixel(int,%20int,%20int%5B%5D))(int x, int y, int[] iArray)            Sets a pixel in the DataBuffer using an int array of samples for input. |
| void | [**setPixels**](http://docs.google.com/java/awt/image/WritableRaster.html#setPixels(int,%20int,%20int,%20int,%20double%5B%5D))(int x, int y, int w, int h, double[] dArray)            Sets all samples for a rectangle of pixels from a double array containing one sample per array element. |
| void | [**setPixels**](http://docs.google.com/java/awt/image/WritableRaster.html#setPixels(int,%20int,%20int,%20int,%20float%5B%5D))(int x, int y, int w, int h, float[] fArray)            Sets all samples for a rectangle of pixels from a float array containing one sample per array element. |
| void | [**setPixels**](http://docs.google.com/java/awt/image/WritableRaster.html#setPixels(int,%20int,%20int,%20int,%20int%5B%5D))(int x, int y, int w, int h, int[] iArray)            Sets all samples for a rectangle of pixels from an int array containing one sample per array element. |
| void | [**setRect**](http://docs.google.com/java/awt/image/WritableRaster.html#setRect(int,%20int,%20java.awt.image.Raster))(int dx, int dy, [Raster](http://docs.google.com/java/awt/image/Raster.html) srcRaster)            Copies pixels from Raster srcRaster to this WritableRaster. |
| void | [**setRect**](http://docs.google.com/java/awt/image/WritableRaster.html#setRect(java.awt.image.Raster))([Raster](http://docs.google.com/java/awt/image/Raster.html) srcRaster)            Copies pixels from Raster srcRaster to this WritableRaster. |
| void | [**setSample**](http://docs.google.com/java/awt/image/WritableRaster.html#setSample(int,%20int,%20int,%20double))(int x, int y, int b, double s)            Sets a sample in the specified band for the pixel located at (x,y) in the DataBuffer using a double for input. |
| void | [**setSample**](http://docs.google.com/java/awt/image/WritableRaster.html#setSample(int,%20int,%20int,%20float))(int x, int y, int b, float s)            Sets a sample in the specified band for the pixel located at (x,y) in the DataBuffer using a float for input. |
| void | [**setSample**](http://docs.google.com/java/awt/image/WritableRaster.html#setSample(int,%20int,%20int,%20int))(int x, int y, int b, int s)            Sets a sample in the specified band for the pixel located at (x,y) in the DataBuffer using an int for input. |
| void | [**setSamples**](http://docs.google.com/java/awt/image/WritableRaster.html#setSamples(int,%20int,%20int,%20int,%20int,%20double%5B%5D))(int x, int y, int w, int h, int b, double[] dArray)            Sets the samples in the specified band for the specified rectangle of pixels from a double array containing one sample per array element. |
| void | [**setSamples**](http://docs.google.com/java/awt/image/WritableRaster.html#setSamples(int,%20int,%20int,%20int,%20int,%20float%5B%5D))(int x, int y, int w, int h, int b, float[] fArray)            Sets the samples in the specified band for the specified rectangle of pixels from a float array containing one sample per array element. |
| void | [**setSamples**](http://docs.google.com/java/awt/image/WritableRaster.html#setSamples(int,%20int,%20int,%20int,%20int,%20int%5B%5D))(int x, int y, int w, int h, int b, int[] iArray)            Sets the samples in the specified band for the specified rectangle of pixels from an int array containing one sample per array element. |

| **Methods inherited from class java.awt.image.**[**Raster**](http://docs.google.com/java/awt/image/Raster.html) |
| --- |
| [createBandedRaster](http://docs.google.com/java/awt/image/Raster.html#createBandedRaster(java.awt.image.DataBuffer,%20int,%20int,%20int,%20int%5B%5D,%20int%5B%5D,%20java.awt.Point)), [createBandedRaster](http://docs.google.com/java/awt/image/Raster.html#createBandedRaster(int,%20int,%20int,%20int,%20int%5B%5D,%20int%5B%5D,%20java.awt.Point)), [createBandedRaster](http://docs.google.com/java/awt/image/Raster.html#createBandedRaster(int,%20int,%20int,%20int,%20java.awt.Point)), [createChild](http://docs.google.com/java/awt/image/Raster.html#createChild(int,%20int,%20int,%20int,%20int,%20int,%20int%5B%5D)), [createCompatibleWritableRaster](http://docs.google.com/java/awt/image/Raster.html#createCompatibleWritableRaster()), [createCompatibleWritableRaster](http://docs.google.com/java/awt/image/Raster.html#createCompatibleWritableRaster(int,%20int)), [createCompatibleWritableRaster](http://docs.google.com/java/awt/image/Raster.html#createCompatibleWritableRaster(int,%20int,%20int,%20int)), [createCompatibleWritableRaster](http://docs.google.com/java/awt/image/Raster.html#createCompatibleWritableRaster(java.awt.Rectangle)), [createInterleavedRaster](http://docs.google.com/java/awt/image/Raster.html#createInterleavedRaster(java.awt.image.DataBuffer,%20int,%20int,%20int,%20int,%20int%5B%5D,%20java.awt.Point)), [createInterleavedRaster](http://docs.google.com/java/awt/image/Raster.html#createInterleavedRaster(int,%20int,%20int,%20int,%20int,%20int%5B%5D,%20java.awt.Point)), [createInterleavedRaster](http://docs.google.com/java/awt/image/Raster.html#createInterleavedRaster(int,%20int,%20int,%20int,%20java.awt.Point)), [createPackedRaster](http://docs.google.com/java/awt/image/Raster.html#createPackedRaster(java.awt.image.DataBuffer,%20int,%20int,%20int,%20int%5B%5D,%20java.awt.Point)), [createPackedRaster](http://docs.google.com/java/awt/image/Raster.html#createPackedRaster(java.awt.image.DataBuffer,%20int,%20int,%20int,%20java.awt.Point)), [createPackedRaster](http://docs.google.com/java/awt/image/Raster.html#createPackedRaster(int,%20int,%20int,%20int%5B%5D,%20java.awt.Point)), [createPackedRaster](http://docs.google.com/java/awt/image/Raster.html#createPackedRaster(int,%20int,%20int,%20int,%20int,%20java.awt.Point)), [createRaster](http://docs.google.com/java/awt/image/Raster.html#createRaster(java.awt.image.SampleModel,%20java.awt.image.DataBuffer,%20java.awt.Point)), [createTranslatedChild](http://docs.google.com/java/awt/image/Raster.html#createTranslatedChild(int,%20int)), [createWritableRaster](http://docs.google.com/java/awt/image/Raster.html#createWritableRaster(java.awt.image.SampleModel,%20java.awt.image.DataBuffer,%20java.awt.Point)), [createWritableRaster](http://docs.google.com/java/awt/image/Raster.html#createWritableRaster(java.awt.image.SampleModel,%20java.awt.Point)), [getBounds](http://docs.google.com/java/awt/image/Raster.html#getBounds()), [getDataBuffer](http://docs.google.com/java/awt/image/Raster.html#getDataBuffer()), [getDataElements](http://docs.google.com/java/awt/image/Raster.html#getDataElements(int,%20int,%20int,%20int,%20java.lang.Object)), [getDataElements](http://docs.google.com/java/awt/image/Raster.html#getDataElements(int,%20int,%20java.lang.Object)), [getHeight](http://docs.google.com/java/awt/image/Raster.html#getHeight()), [getMinX](http://docs.google.com/java/awt/image/Raster.html#getMinX()), [getMinY](http://docs.google.com/java/awt/image/Raster.html#getMinY()), [getNumBands](http://docs.google.com/java/awt/image/Raster.html#getNumBands()), [getNumDataElements](http://docs.google.com/java/awt/image/Raster.html#getNumDataElements()), [getParent](http://docs.google.com/java/awt/image/Raster.html#getParent()), [getPixel](http://docs.google.com/java/awt/image/Raster.html#getPixel(int,%20int,%20double%5B%5D)), [getPixel](http://docs.google.com/java/awt/image/Raster.html#getPixel(int,%20int,%20float%5B%5D)), [getPixel](http://docs.google.com/java/awt/image/Raster.html#getPixel(int,%20int,%20int%5B%5D)), [getPixels](http://docs.google.com/java/awt/image/Raster.html#getPixels(int,%20int,%20int,%20int,%20double%5B%5D)), [getPixels](http://docs.google.com/java/awt/image/Raster.html#getPixels(int,%20int,%20int,%20int,%20float%5B%5D)), [getPixels](http://docs.google.com/java/awt/image/Raster.html#getPixels(int,%20int,%20int,%20int,%20int%5B%5D)), [getSample](http://docs.google.com/java/awt/image/Raster.html#getSample(int,%20int,%20int)), [getSampleDouble](http://docs.google.com/java/awt/image/Raster.html#getSampleDouble(int,%20int,%20int)), [getSampleFloat](http://docs.google.com/java/awt/image/Raster.html#getSampleFloat(int,%20int,%20int)), [getSampleModel](http://docs.google.com/java/awt/image/Raster.html#getSampleModel()), [getSampleModelTranslateX](http://docs.google.com/java/awt/image/Raster.html#getSampleModelTranslateX()), [getSampleModelTranslateY](http://docs.google.com/java/awt/image/Raster.html#getSampleModelTranslateY()), [getSamples](http://docs.google.com/java/awt/image/Raster.html#getSamples(int,%20int,%20int,%20int,%20int,%20double%5B%5D)), [getSamples](http://docs.google.com/java/awt/image/Raster.html#getSamples(int,%20int,%20int,%20int,%20int,%20float%5B%5D)), [getSamples](http://docs.google.com/java/awt/image/Raster.html#getSamples(int,%20int,%20int,%20int,%20int,%20int%5B%5D)), [getTransferType](http://docs.google.com/java/awt/image/Raster.html#getTransferType()), [getWidth](http://docs.google.com/java/awt/image/Raster.html#getWidth()) |

| **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** |
| --- |

### WritableRaster

protected **WritableRaster**([SampleModel](http://docs.google.com/java/awt/image/SampleModel.html) sampleModel,  
 [Point](http://docs.google.com/java/awt/Point.html) origin)

Constructs a WritableRaster with the given SampleModel. The WritableRaster's upper left corner is origin and it is the same size as the SampleModel. A DataBuffer large enough to describe the WritableRaster is automatically created.

**Parameters:**sampleModel - The SampleModel that specifies the layout.origin - The Point that specifies the origin. **Throws:** [RasterFormatException](http://docs.google.com/java/awt/image/RasterFormatException.html) - if computing either origin.x + sampleModel.getWidth() or origin.y + sampleModel.getHeight() results in integer overflow

### WritableRaster

protected **WritableRaster**([SampleModel](http://docs.google.com/java/awt/image/SampleModel.html) sampleModel,  
 [DataBuffer](http://docs.google.com/java/awt/image/DataBuffer.html) dataBuffer,  
 [Point](http://docs.google.com/java/awt/Point.html) origin)

Constructs a WritableRaster with the given SampleModel and DataBuffer. The WritableRaster's upper left corner is origin and it is the same size as the SampleModel. The DataBuffer is not initialized and must be compatible with SampleModel.

**Parameters:**sampleModel - The SampleModel that specifies the layout.dataBuffer - The DataBuffer that contains the image data.origin - The Point that specifies the origin. **Throws:** [RasterFormatException](http://docs.google.com/java/awt/image/RasterFormatException.html) - if computing either origin.x + sampleModel.getWidth() or origin.y + sampleModel.getHeight() results in integer overflow

### WritableRaster

protected **WritableRaster**([SampleModel](http://docs.google.com/java/awt/image/SampleModel.html) sampleModel,  
 [DataBuffer](http://docs.google.com/java/awt/image/DataBuffer.html) dataBuffer,  
 [Rectangle](http://docs.google.com/java/awt/Rectangle.html) aRegion,  
 [Point](http://docs.google.com/java/awt/Point.html) sampleModelTranslate,  
 [WritableRaster](http://docs.google.com/java/awt/image/WritableRaster.html) parent)

Constructs a WritableRaster with the given SampleModel, DataBuffer, and parent. aRegion specifies the bounding rectangle of the new Raster. When translated into the base Raster's coordinate system, aRegion must be contained by the base Raster. (The base Raster is the Raster's ancestor which has no parent.) sampleModelTranslate specifies the sampleModelTranslateX and sampleModelTranslateY values of the new Raster. Note that this constructor should generally be called by other constructors or create methods, it should not be used directly.

**Parameters:**sampleModel - The SampleModel that specifies the layout.dataBuffer - The DataBuffer that contains the image data.aRegion - The Rectangle that specifies the image area.sampleModelTranslate - The Point that specifies the translation from SampleModel to Raster coordinates.parent - The parent (if any) of this raster. **Throws:** [RasterFormatException](http://docs.google.com/java/awt/image/RasterFormatException.html) - if aRegion has width or height less than or equal to zero, or computing either aRegion.x + aRegion.width or aRegion.y + aRegion.height results in integer overflow

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

### getWritableParent

public [WritableRaster](http://docs.google.com/java/awt/image/WritableRaster.html) **getWritableParent**()

Returns the parent WritableRaster (if any) of this WritableRaster, or else null.

**Returns:**the parent of this WritableRaster, or null.

### createWritableTranslatedChild

public [WritableRaster](http://docs.google.com/java/awt/image/WritableRaster.html) **createWritableTranslatedChild**(int childMinX,  
 int childMinY)

Create a WritableRaster with the same size, SampleModel and DataBuffer as this one, but with a different location. The new WritableRaster will possess a reference to the current WritableRaster, accessible through its getParent() and getWritableParent() methods.

**Parameters:**childMinX - X coord of the upper left corner of the new Raster.childMinY - Y coord of the upper left corner of the new Raster. **Returns:**a WritableRaster the same as this one except for the specified location. **Throws:** [RasterFormatException](http://docs.google.com/java/awt/image/RasterFormatException.html) - if computing either childMinX + this.getWidth() or childMinY + this.getHeight() results in integer overflow

### createWritableChild

public [WritableRaster](http://docs.google.com/java/awt/image/WritableRaster.html) **createWritableChild**(int parentX,  
 int parentY,  
 int w,  
 int h,  
 int childMinX,  
 int childMinY,  
 int[] bandList)

Returns a new WritableRaster which shares all or part of this WritableRaster's DataBuffer. The new WritableRaster will possess a reference to the current WritableRaster, accessible through its getParent() and getWritableParent() methods.

The parentX, parentY, width and height parameters form a Rectangle in this WritableRaster's coordinate space, indicating the area of pixels to be shared. An error will be thrown if this Rectangle is not contained with the bounds of the current WritableRaster.

The new WritableRaster may additionally be translated to a different coordinate system for the plane than that used by the current WritableRaster. The childMinX and childMinY parameters give the new (x, y) coordinate of the upper-left pixel of the returned WritableRaster; the coordinate (childMinX, childMinY) in the new WritableRaster will map to the same pixel as the coordinate (parentX, parentY) in the current WritableRaster.

The new WritableRaster may be defined to contain only a subset of the bands of the current WritableRaster, possibly reordered, by means of the bandList parameter. If bandList is null, it is taken to include all of the bands of the current WritableRaster in their current order.

To create a new WritableRaster that contains a subregion of the current WritableRaster, but shares its coordinate system and bands, this method should be called with childMinX equal to parentX, childMinY equal to parentY, and bandList equal to null.

**Parameters:**parentX - X coordinate of the upper left corner in this WritableRaster's coordinates.parentY - Y coordinate of the upper left corner in this WritableRaster's coordinates.w - Width of the region starting at (parentX, parentY).h - Height of the region starting at (parentX, parentY).childMinX - X coordinate of the upper left corner of the returned WritableRaster.childMinY - Y coordinate of the upper left corner of the returned WritableRaster.bandList - Array of band indices, or null to use all bands. **Returns:**a WritableRaster sharing all or part of the DataBuffer of this WritableRaster. **Throws:** [RasterFormatException](http://docs.google.com/java/awt/image/RasterFormatException.html) - if the subregion is outside of the raster bounds. [RasterFormatException](http://docs.google.com/java/awt/image/RasterFormatException.html) - if w or h is less than or equal to zero, or computing any of parentX + w, parentY + h, childMinX + w, or childMinY + h results in integer overflow

### setDataElements

public void **setDataElements**(int x,  
 int y,  
 [Object](http://docs.google.com/java/lang/Object.html) inData)

Sets the data for a single pixel from a primitive array of type TransferType. For image data supported by the Java 2D(tm) API, this will be one of DataBuffer.TYPE\_BYTE, DataBuffer.TYPE\_USHORT, DataBuffer.TYPE\_INT, DataBuffer.TYPE\_SHORT, DataBuffer.TYPE\_FLOAT, or DataBuffer.TYPE\_DOUBLE. Data in the array may be in a packed format, thus increasing efficiency for data transfers. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds, or if inData is not large enough to hold the pixel data. However, explicit bounds checking is not guaranteed. A ClassCastException will be thrown if the input object is not null and references anything other than an array of TransferType.

**Parameters:**x - The X coordinate of the pixel location.y - The Y coordinate of the pixel location.inData - An object reference to an array of type defined by getTransferType() and length getNumDataElements() containing the pixel data to place at x,y. **Throws:** [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates are not in bounds, or if inData is too small to hold the input.**See Also:**[SampleModel.setDataElements(int, int, Object, DataBuffer)](http://docs.google.com/java/awt/image/SampleModel.html#setDataElements(int,%20int,%20java.lang.Object,%20java.awt.image.DataBuffer))

### setDataElements

public void **setDataElements**(int x,  
 int y,  
 [Raster](http://docs.google.com/java/awt/image/Raster.html) inRaster)

Sets the data for a rectangle of pixels from an input Raster. The input Raster must be compatible with this WritableRaster in that they must have the same number of bands, corresponding bands must have the same number of bits per sample, the TransferTypes and NumDataElements must be the same, and the packing used by the getDataElements/setDataElements must be identical. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the pixel location.y - The Y coordinate of the pixel location.inRaster - Raster containing data to place at x,y. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if inRaster is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates are not in bounds.

### setDataElements

public void **setDataElements**(int x,  
 int y,  
 int w,  
 int h,  
 [Object](http://docs.google.com/java/lang/Object.html) inData)

Sets the data for a rectangle of pixels from a primitive array of type TransferType. For image data supported by the Java 2D API, this will be one of DataBuffer.TYPE\_BYTE, DataBuffer.TYPE\_USHORT, DataBuffer.TYPE\_INT, DataBuffer.TYPE\_SHORT, DataBuffer.TYPE\_FLOAT, or DataBuffer.TYPE\_DOUBLE. Data in the array may be in a packed format, thus increasing efficiency for data transfers. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds, or if inData is not large enough to hold the pixel data. However, explicit bounds checking is not guaranteed. A ClassCastException will be thrown if the input object is not null and references anything other than an array of TransferType.

**Parameters:**x - The X coordinate of the upper left pixel location.y - The Y coordinate of the upper left pixel location.w - Width of the pixel rectangle.h - Height of the pixel rectangle.inData - An object reference to an array of type defined by getTransferType() and length w\*h\*getNumDataElements() containing the pixel data to place between x,y and x+w-1, y+h-1. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if inData is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates are not in bounds, or if inData is too small to hold the input.**See Also:**[SampleModel.setDataElements(int, int, int, int, Object, DataBuffer)](http://docs.google.com/java/awt/image/SampleModel.html#setDataElements(int,%20int,%20int,%20int,%20java.lang.Object,%20java.awt.image.DataBuffer))

### setRect

public void **setRect**([Raster](http://docs.google.com/java/awt/image/Raster.html) srcRaster)

Copies pixels from Raster srcRaster to this WritableRaster. Each pixel in srcRaster is copied to the same x,y address in this raster, unless the address falls outside the bounds of this raster. srcRaster must have the same number of bands as this WritableRaster. The copy is a simple copy of source samples to the corresponding destination samples.

If all samples of both source and destination Rasters are of integral type and less than or equal to 32 bits in size, then calling this method is equivalent to executing the following code for all x,y addresses valid in both Rasters.

Raster srcRaster;  
 WritableRaster dstRaster;  
 for (int b = 0; b < srcRaster.getNumBands(); b++) {  
 dstRaster.setSample(x, y, b, srcRaster.getSample(x, y, b));  
 }

Thus, when copying an integral type source to an integral type destination, if the source sample size is greater than the destination sample size for a particular band, the high order bits of the source sample are truncated. If the source sample size is less than the destination size for a particular band, the high order bits of the destination are zero-extended or sign-extended depending on whether srcRaster's SampleModel treats the sample as a signed or unsigned quantity.

When copying a float or double source to an integral type destination, each source sample is cast to the destination type. When copying an integral type source to a float or double destination, the source is first converted to a 32-bit int (if necessary), using the above rules for integral types, and then the int is cast to float or double.

**Parameters:**srcRaster - The Raster from which to copy pixels. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if srcRaster is null.

### setRect

public void **setRect**(int dx,  
 int dy,  
 [Raster](http://docs.google.com/java/awt/image/Raster.html) srcRaster)

Copies pixels from Raster srcRaster to this WritableRaster. For each (x, y) address in srcRaster, the corresponding pixel is copied to address (x+dx, y+dy) in this WritableRaster, unless (x+dx, y+dy) falls outside the bounds of this raster. srcRaster must have the same number of bands as this WritableRaster. The copy is a simple copy of source samples to the corresponding destination samples. For details, see [setRect(Raster)](http://docs.google.com/java/awt/image/WritableRaster.html#setRect(java.awt.image.Raster)).

**Parameters:**dx - The X translation factor from src space to dst space of the copy.dy - The Y translation factor from src space to dst space of the copy.srcRaster - The Raster from which to copy pixels. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if srcRaster is null.

### setPixel

public void **setPixel**(int x,  
 int y,  
 int[] iArray)

Sets a pixel in the DataBuffer using an int array of samples for input. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the pixel location.y - The Y coordinate of the pixel location.iArray - The input samples in a int array. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if iArray is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates are not in bounds, or if iArray is too small to hold the input.

### setPixel

public void **setPixel**(int x,  
 int y,  
 float[] fArray)

Sets a pixel in the DataBuffer using a float array of samples for input. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the pixel location.y - The Y coordinate of the pixel location.fArray - The input samples in a float array. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if fArray is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates are not in bounds, or if fArray is too small to hold the input.

### setPixel

public void **setPixel**(int x,  
 int y,  
 double[] dArray)

Sets a pixel in the DataBuffer using a double array of samples for input. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the pixel location.y - The Y coordinate of the pixel location.dArray - The input samples in a double array. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if dArray is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates are not in bounds, or if dArray is too small to hold the input.

### setPixels

public void **setPixels**(int x,  
 int y,  
 int w,  
 int h,  
 int[] iArray)

Sets all samples for a rectangle of pixels from an int array containing one sample per array element. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the upper left pixel location.y - The Y coordinate of the upper left pixel location.w - Width of the pixel rectangle.h - Height of the pixel rectangle.iArray - The input int pixel array. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if iArray is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates are not in bounds, or if iArray is too small to hold the input.

### setPixels

public void **setPixels**(int x,  
 int y,  
 int w,  
 int h,  
 float[] fArray)

Sets all samples for a rectangle of pixels from a float array containing one sample per array element. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the upper left pixel location.y - The Y coordinate of the upper left pixel location.w - Width of the pixel rectangle.h - Height of the pixel rectangle.fArray - The input float pixel array. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if fArray is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates are not in bounds, or if fArray is too small to hold the input.

### setPixels

public void **setPixels**(int x,  
 int y,  
 int w,  
 int h,  
 double[] dArray)

Sets all samples for a rectangle of pixels from a double array containing one sample per array element. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the upper left pixel location.y - The Y coordinate of the upper left pixel location.w - Width of the pixel rectangle.h - Height of the pixel rectangle.dArray - The input double pixel array. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if dArray is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates are not in bounds, or if dArray is too small to hold the input.

### setSample

public void **setSample**(int x,  
 int y,  
 int b,  
 int s)

Sets a sample in the specified band for the pixel located at (x,y) in the DataBuffer using an int for input. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the pixel location.y - The Y coordinate of the pixel location.b - The band to set.s - The input sample. **Throws:** [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates or the band index are not in bounds.

### setSample

public void **setSample**(int x,  
 int y,  
 int b,  
 float s)

Sets a sample in the specified band for the pixel located at (x,y) in the DataBuffer using a float for input. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the pixel location.y - The Y coordinate of the pixel location.b - The band to set.s - The input sample as a float. **Throws:** [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates or the band index are not in bounds.

### setSample

public void **setSample**(int x,  
 int y,  
 int b,  
 double s)

Sets a sample in the specified band for the pixel located at (x,y) in the DataBuffer using a double for input. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the pixel location.y - The Y coordinate of the pixel location.b - The band to set.s - The input sample as a double. **Throws:** [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates or the band index are not in bounds.

### setSamples

public void **setSamples**(int x,  
 int y,  
 int w,  
 int h,  
 int b,  
 int[] iArray)

Sets the samples in the specified band for the specified rectangle of pixels from an int array containing one sample per array element. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the upper left pixel location.y - The Y coordinate of the upper left pixel location.w - Width of the pixel rectangle.h - Height of the pixel rectangle.b - The band to set.iArray - The input int sample array. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if iArray is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates or the band index are not in bounds, or if iArray is too small to hold the input.

### setSamples

public void **setSamples**(int x,  
 int y,  
 int w,  
 int h,  
 int b,  
 float[] fArray)

Sets the samples in the specified band for the specified rectangle of pixels from a float array containing one sample per array element. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the upper left pixel location.y - The Y coordinate of the upper left pixel location.w - Width of the pixel rectangle.h - Height of the pixel rectangle.b - The band to set.fArray - The input float sample array. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if fArray is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates or the band index are not in bounds, or if fArray is too small to hold the input.

### setSamples

public void **setSamples**(int x,  
 int y,  
 int w,  
 int h,  
 int b,  
 double[] dArray)

Sets the samples in the specified band for the specified rectangle of pixels from a double array containing one sample per array element. An ArrayIndexOutOfBoundsException may be thrown if the coordinates are not in bounds. However, explicit bounds checking is not guaranteed.

**Parameters:**x - The X coordinate of the upper left pixel location.y - The Y coordinate of the upper left pixel location.w - Width of the pixel rectangle.h - Height of the pixel rectangle.b - The band to set.dArray - The input double sample array. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if dArray is null. [ArrayIndexOutOfBoundsException](http://docs.google.com/java/lang/ArrayIndexOutOfBoundsException.html) - if the coordinates or the band index are not in bounds, or if dArray is too small to hold the input.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/WritableRaster.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/VolatileImage.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/image/WritableRenderedImage.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/WritableRaster.html)    [**NO FRAMES**](http://docs.google.com/WritableRaster.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#lnxbz9) |

[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).