Interface java.awt.image.ImageObserver

public interface **ImageObserver**
extends Object

An asynchronous update interface for receiving notifications about Image information as the Image is constructed.

Version:
1.14 09/08/95
Author:
Jim Graham

Variable Index

o **ALLBITS**
A static image which was previously drawn is now complete and can be drawn again in its final form.
o **ERROR**
An image which was being tracked asynchronously has encountered an error.
o **FRAMEBITS**
Another complete frame of a multi–frame image which was previously drawn is now available to be drawn again.
o **HEIGHT**
The height of the base image is now available and can be taken from the height argument to the imageUpdate callback method.
o **PROPERTIES**
The properties of the image are now available.
o **SOMEBITS**
More pixels needed for drawing a scaled variation of the image are available.
o **WIDTH**
The width of the base image is now available and can be taken from the width argument to the imageUpdate callback method.

Method Index

o **imageUpdate**(Image, int, int, int, int, int)
This method is called when information about an image which was previously requested using an asynchronous interface becomes available.

Variables

o **WIDTH**

```
public final static int WIDTH
```

The width of the base image is now available and can be taken from the width argument to the imageUpdate callback method.
See Also:
getWidth, imageUpdate

o **HEIGHT**

```
public final static int HEIGHT
```

The height of the base image is now available and can be taken from the height argument to the imageUpdate callback method.
See Also:
getHeight, imageUpdate

o **PROPERTIES**

```
public final static int PROPERTIES
```

The properties of the image are now available.
See Also:
getProperty, imageUpdate

o **SOMEBITS**

```
public final static int SOMEBITS
```

More pixels needed for drawing a scaled variation of the image are available. The bounding box of the new pixels can be taken from the x, y, width, and height arguments to the imageUpdate callback method.
See Also:
drawImage, imageUpdate

o **FRAMEBITS**

```
public final static int FRAMEBITS
```

Another complete frame of a multi–frame image which was previously drawn is now available to be drawn again. The x, y, width, and height arguments to the imageUpdate callback method should be ignored.
See Also:
drawImage, imageUpdate

o **ALLBITS**

```
public final static int ALLBITS
```

A static image which was previously drawn is now complete and can be drawn again in its final form. The x, y, width, and height arguments to the imageUpdate callback method should be ignored.

See Also:

drawImage, imageUpdate

o ERROR

```
public final static int ERROR
```

An image which was being tracked asynchronously has encountered an error. No further information will become available and drawing the image will fail.

See Also:

imageUpdate

Methods

o imageUpdate

```
public abstract boolean imageUpdate(Image img,
                                    int infoflags,
                                    int x,
                                    int y,
                                    int width,
                                    int height)
```

This method is called when information about an image which was previously requested using an asynchronous interface becomes available. Asynchronous interfaces are method calls such as getWidth(ImageObserver) and drawImage(img, x, y, ImageObserver) which take an ImageObserver object as an argument. Those methods register the caller as interested either in information about the overall image itself (in the case of getWidth(ImageObserver)) or about an output version of an image (in the case of the drawImage(img, x, y, [w, h,] ImageObserver) call). This method should return true if further updates are needed or false if the required information has been acquired. The image which was being tracked is passed in using the img argument. Various constants are combined to form the infoflags argument which indicates what information about the image is now available. The interpretation of the x, y, width, and height arguments depends on the contents of the infoflags argument.

See Also:

getWidth, getHeight, drawImage
