

comp.lang.java.gui FAQ

Source: <http://coding.derkeiler.com/Archive/Java/comp.lang.java.gui/2004-10/0076.html>

From: Thomas Weidenfeller (*nobody_at_eed.ericsson.invalid*)

Date: 10/04/04

Date: Mon, 4 Oct 2004 10:25:31 +0000 (UTC)

Archive-name: computer-lang/java/gui/faq

Version: \$Revision: 1.13 \$

Posting-Frequency: monthly

Copyright: Copyright (c) 2003, 2004 Thomas Weidenfeller

Maintainer: Thomas Weidenfeller. See below for mailing instructions.

Last-modified: \$Date: 2004/10/04 10:08:15 \$

comp.lang.java.gui FAQ

Table of Contents

1 Introduction

- Q1.1 What is this, and what does it contain?
- Q1.2 Where do I find a copy of the FAQ?
- Q1.3 There are so many Java FAQs.
Which is the right, official one?
- Q1.4 Does Sun support or endorse this FAQ?
- Q1.5 I noticed broken links in the FAQ.
Don't you verify them before publishing?
- Q1.6 Is there an HTML, Word, <whatever format> version of the FAQ?
- Q1.7 What is AWT?
- Q1.8 What is Swing?
- Q1.9 What is SWT?

2 The comp.lang.java.gui Newsgroup

- Q2.1 What is the newsgroup's charter? What are acceptable topics?
- Q2.2 Which topics are not welcome in the newsgroup?
- Q2.3 Where can I find an archive of the newsgroup?
- Q2.4 What is an SSCCE?
- Q2.5 Why don't people like top-posting? What is top-posting?
- Q2.6 Is there more about posting to newsgroups and asking questions?
- Q2.7 Does Sun support or endorse the newsgroup?

3 The Top 5 Questions

- Q3.1 My GUI freezes or doesn't update. What to do?
- Q3.2 How do I update the GUI from another thread?
- Q3.3 I have arranged all my widgets nicely on a window. Then I changed

the OS / Java version / font / PLAF. Now everything is broken.

What's going on?

Q3.4 My graphics on a Canvas/JPanel/JComponent, etc. gets corrupted, or I get a null pointer exception when trying to draw.

How can I avoid this?

Q3.5 How to create a transparent or non-rectangular window?

4 Architecture

Q4.1 What is this Model-View-Controller (MVC) stuff?

Q4.2 What is the Swing single-threading issue?

Q4.3 What is the right way to start a Swing GUI?

5 Window / [J]Frame / [J]Dialog (Top-Level Containers)

Q5.1 How can I ensure a window is always on top of all other windows using AWT or Swing?

Q5.2 How can I (de)iconify a window?

Q5.3 How can I replace/remove the icon in the title bar (window decoration) of a [J]Frame?

Q5.4 How to replace the icon in the title bar (window decoration) of a [J]Dialog?

Q5.5 My modal dialog goes behind the main window. How can I ensure it is in front instead?

Q5.6 How to bind the escape key to the JDialog cancel operation?

Q5.7 How can I implement my own JFrame/JDialog close handling?

Q5.8 How Do I center a window on the screen? How do I get the screen size?

Q5.9 How to ensure a minimum or maximum window size?

Q5.10 How to ensure a particular aspect ration of a window?

Q5.11 How can I delegate the window placement to the window system or manager?

Q5.12 I need to take some toolbar (dock, panel) size into account when calculating a window position an/or size. How?

6 [J]Component (Widgets)

6.1 General Questions

Q6.1.1 How do I position components (widgets) on a window?

Q6.1.2 How to create a transparent widget?

Q6.1.3 How to create a non-rectangular widget?

Q6.1.4 What are Insets?

Q6.1.5 How do I find a component's top-level container (e.g. the window)?

6.2 JTree

Q6.2.1 I changed the data / structure for my JTree, but the display doesn't get updated. What's going on?

Q6.2.2 How do I set a custom icon for a node?

Q6.2.3 How do I remove all my nodes from a JTree at once?

6.3 Styled Text / JEditorPane / JTextPane

Q6.3.1 Can I use RTFEditorKit to read RTF documents created by Word?

Q6.3.2 I have problems using the Swing HTML parser to parse all

kinds of HTML. Is this normal?

Q6.3.3 Some of my CSS styles don't work out. Is this normal?

Q6.3.4 Can I use Swing's HTML support to write a web browser?

Q6.3.5 Can I use Swing's HTML support to build an on-line help system or e-book?

Q6.3.6 If HTML support is really so broken in Java, what is it good for?

6.4 [J]TextArea

Q6.4.1 I append text to a JTextArea. How to ensure the text area is always scrolled down to the end of the text?

Q6.4.2 How to use several different fonts (styles, sizes) in one [J]TextArea?

6.5 [J]Label / [J]Button

Q6.5.1 How can I have multiple Lines in a [J]Label?

Q6.5.2 I want to have a hyperlink in a [J]Label. How can I do this?

Q6.5.3 How do I make a JButton the default button in a JDialog?

7 Graphics & Painting

Q7.1 What is the equivalent of AWT's Canvas in Swing?

Q7.2 When drawing on a JPanel, the background is garbled.

Q7.3 How do I generate some charts / plots in Java?

Q7.4 How to draw some graphs in Java?

Q7.5 I want to write a diagram editor. Where to start?

Q7.6 How do I draw lines between JLabels on a JPanel?

Q7.7 How to debug graph painting?

Q7.8 I need to draw a tree. How?

Q7.9 I need an algorithm for drawing ...

Q7.10 When I subclass JPanel/JComponent, I need to override paint(), right?

Q7.11 Why does drawImage() fail when I try to display a loaded image? Why are the width and height of my loaded image both zero?

Q7.12 How do I resize (zoom in/out) my Graphics?

8 Other Common Questions

Q8.1 My GUI has rendering problems when the JMenu opens over my top Panel ...

Q8.2 How can I do this JavaScript thing on my web site?

Q8.3 I want to make ...

9 Resources

9.1 Sun's Java Web Site

9.2 Other Sun Sites

9.3 Icons

9.4 Misc. Examples, Tips and Tricks

9.5 Style Guides

9.6 SDK Documentation

9.7 More Swing

9.8 Online Magazines

9.9 Java 2D API

9.10 Java 3D API

9.11 General Java

9.12 More?

Q9.12.1 But I need more!

10 Improvement Suggestions

11 Acknowledgments

1 Introduction

~~~~~

Q1.1 What is this, and what does it contain?

This is the FAQ for the comp.lang.java.gui newsgroup. It mostly deals with Swing issues, and contains some AWT information, too.

Q1.2 Where do I find a copy of the FAQ?

1) The FAQ is regularly posted to

news:comp.lang.java.gui

2) Just search an archive like

<http://groups.google.com/groups?group=comp.lang.java.gui>

3) People have placed (sometimes older) versions of the FAQ on web sites, too, e.g.

<http://mindprod.com/jgloss/guifaq.html>

<http://www.physci.org/guifaq.jsp>

4) There is even a Japanese translation around:

<http://homepage1.nifty.com/algafield/JavaGUIFaq19j.html>

Q1.3 There are so many Java FAQs.

Which is the right, official one?

There is probably not THE FAQ. Everyone can start an FAQ, and many have done so. See it as some benefit. There is a lot of information out there. Note, however, that there is at least one so-called "Java FAQ" which just recycles postings from the comp.lang.java.\* newsgroups and distributes them as a newsletter and as a book. You might consider searching an archive of the groups instead of paying for this material.

For a list of other FAQs and FAQ lists :- ) see the "Resources" chapter of this FAQ and

<http://mindprod.com/jgloss/faqs.html>

Q1.4 Does Sun support or endorse this FAQ?

No, it is just a newsgroup FAQ. Sun probably doesn't know about it.

NOTE: The author of this FAQ does not have any inside information or contacts to Sun's Java or GUI development team and has never been contacted by Sun. Therefore, the FAQ's author is not in a position to forward suggestions to Sun or help with expedited answers from Sun. Please refrain from such requests.

See also: "Q2.7 Does Sun support or endorse the newsgroup?"

Q1.5 I noticed broken links in the FAQ.

Don't you verify them before publishing?

No, I don't. I rely on feedback from readers. Also, links might break at any time, e.g. just seconds after a link has been verified.

Q1.6 Is there an HTML, Word, <whatever format> version of the FAQ?

There is no such official version. Some people went through the effort to convert the FAQ to HTML. I suggest to use the

<http://txt2html.sourceforge.net>

software to create an HTML version for personal usage.

See also: "Q1.2 Where do I find a copy of the FAQ?"

Q1.7 What is AWT?

AWT (The Abstract Window Toolkit) is Sun's first Java GUI toolkit. It is rather limited and uses the native GUI components of the operating system.

Unless you have to support an old VM, Swing is usually the better choice for a Java GUI toolkit.

Q1.8 What is Swing?

Swing is Sun's second attempt at a Java toolkit. It is rich in functions and widgets, and is considered the standard Java GUI toolkit. Nowadays it is bundled with the Java 2 Standard Edition.

Most parts of Swing are written in Java, especially most of the GUI components. Swing uses some parts of AWT in order to gain access to the native GUI system for event handling and top-level containers. It is built on AWT's lightweight component framework.

## Q1.9 What is SWT?

SWT is an alternative GUI toolkit from IBM. Unlike AWT and Swing, it is not part of the Java 2 Standard Edition. You have to obtain it separately for the platforms you want to support (it uses a native library).

## 2 The comp.lang.java.gui Newsgroup

~~~~~

Q2.1 What is the newsgroup's charter? What are acceptable topics?

The voting for the group with the group's charter passed on 1997-04-10:

<http://groups.google.com/groups?selm=860665862.3170%40isc.org>

A longer history of comp.lang.java.* reorganizations can be found in

<ftp://ftp.isc.org/usenet/news.announce.newgroups/comp/comp.lang.java-reorg>

Here is an excerpt from the '97 reorg charter (Note, "all groups" refers to all the Java groups from that voting, including comp.lang.java.gui):

CHARTER: all groups

The normal practice should be that most articles are posted to one single, correct group ONLY. Cross-posting is only appropriate when the problem is hard to categorize or when it legitimately concerns more than one group. Answers should be posted to a single group only once the nature of the problem has been ascertained. Many articles of this sort should go to comp.lang.java.help (only).

It is not appropriate to post binary class files or long (longer than one or two screenfuls) source listings on any of these groups. Instead, the post should reference a WWW or FTP site (short source snippets to demonstrate a particular point or problem are fine).

END CHARTER.

[...]

CHARTER: comp.lang.java.gui

This unmoderated group is for any and all discussion relating to GUI toolkits or window frameworks in Java. Topics include the AWT, Netscape's IFC, Microsoft's planned AFC, Visix's Vibe toolkit, among others. The newsgroup will also be the appropriate place for discussion of the JDK event model, mouse and keyboard issues,

bugs in windowing code, and graphics programming in Java. If it concerns something that can be seen on the screen, it belongs in this group.

END CHARTER.

One will note the list of ancient Java GUI technologies, and the absence of Swing. It is save to say, that nowadays most discussions are about Swing, plus a few about AWT and Java printing.

Q2.2 Which topics are not welcome in the newsgroup?

Of course, this question is never asked, but over the time, it has turned out that certain types of postings are not welcome, even if Java related. This includes:

- * Posting your homework.
- * Advocacy. Goto `comp.lang.java.advocacy` instead.
- * Postings urging the readers to help. Especially in conjunction with whining. No one in the group is paid to help you. No one owns you anything.
- * Public and hidden advertising. You think you are too clever to be caught? Well, read this thread in our sister group `c.l.j.programmer` first, and watch how some business lost all its reputation:

<http://groups.google.com/groups?threadm=58882783.0307010403.67c54a5f%40posting.google.com>
- * Postings which just contain a statement like "Help! It does not work!", without any additional information, like the exact error message, source code, or even a hint what "it" is supposed to mean.
- * Postings demonstrating unwillingness to learn are not welcome, too. And learning starts by reading the API documentation before posting.
- * Test messages are not welcome. Instead, use `alt.test.*`, and learn how newsgroups work.

See also: "Q2.4 What is an SSCCE?"

Q2.3 Where can I find an archive of the newsgroup?

See e.g.

<http://groups.google.com/groups?group=comp.lang.java.gui>

Q2.4 What is an SSCCE?

Short/small, self contained, compilable, example (source code).

It would be best if you provide such short (see the group's charter) example source code in our first request for help. When asked for one, please don't complain that your source code is too large, too tricky, too secret for being cut down to a reasonable size and posted. You have the problem, and you asked in a public forum, so it is in your interest to provide the requested information.

For more information about hacking some example code together, go to

<http://www.physci.org/codes/sscce.jsp>

Q2.5 Why don't people like top-posting? What is top-posting?

See the following Question & Answer (well, Answer & Question) section:

A: Because it messes up the order in which people normally read text.

Q: Why is top-posting such a bad thing?

A: Top-posting.

Q: What is the most annoying thing on Usenet and in e-mail?

--- Common Usenet signature

Q2.6 Is there more about posting to newsgroups and asking questions?

Yes, see e.g. the following for making the most out of a newsgroup (ignore the hacker slang):

<http://www.catb.org/~esr/faqs/smart-questions.html>

[The author of that web page has requested a notice that he is not a general help desk for all your problems].

<http://www.yoda.arachsys.com/java/newsgroups.html>

Also see the newsgroups:

news:news.newusers.questions

news:news.answers

And RFC 1855. E.g. at

<ftp://ftp.rfc-editor.org/in-notes/rfc1855.txt>

<http://www.faqs.org/rfcs/rfc1855.html>

Q2.7 Does Sun support or endorse the newsgroup?

Frankly, no one really knows, but it doesn't appear so. From time to time a poster can be spotted apparently working for Sun's Swing

development team. But there is no one in an official capacity from Sun taking part in the discussions.

It is safe to say that the newsgroup isn't of much interest for Sun, and that suggestions or informed opinions posted to the group will most likely not be seen, noted or followed-up by Sun.

You might want to try

<http://developer.java.sun.com/developer/bugParade/>

or

<http://jcp.org/>

if you want to suggest some changes to Java. Good luck.

See also: "Q1.4 Does Sun support or endorse this FAQ?"

3 The Top 5 Questions

~~~~~

#### Q3.1 My GUI freezes or doesn't update. What to do?

Most likely you are blocking the event dispatching thread (EDT). Offload time-consuming tasks from your event listeners to separate threads.

You can do the necessary implementation by hand, or you can use existing frameworks like the `SwingWorker` class from Sun. See the series of articles in

<http://java.sun.com/products/jfc/tsc/articles/threads/threads1.html>

For some special cases you can give `paintImmediately()` a look. See

<http://java.sun.com/products/jfc/tsc/articles/painting/index.html>

for some information about using `paintImmediately()`.

See also: "Q3.2 How do I update the GUI from another thread?"

#### Q3.2 How do I update the GUI from another thread?

If you have to update the GUI from another thread (e.g. once you offloaded a time consuming task from the EDT to another thread) you should use the `javax.swing.SwingUtilities.invokeLater()` or `javax.swing.SwingUtilities.invokeAndWait()`. Usually you want `invokeLater()`.

The code is rather simple. E.g. when using an anonymous class:

```
SwingUtilities.invokeLater(new Runnable() {  
    public void run() {  
        // Code to be executed on the EDT  
    }  
});  
// Current thread will immediately continue here
```

And

```
SwingUtilities.invokeAndWait(new Runnable() {  
    public void run() {  
        // Code to be executed on the EDT  
    }  
});  
// Current thread will wait until code has been executed on  
// the EDT.
```

Again, see

<http://java.sun.com/products/jfc/tsc/articles/threads/threads1.html>

for more details.

See also: "Q3.1 My GUI freezes or doesn't update. What to do?"

Q3.3 I have arranged all my widgets nicely on a window. Then I changed the OS / Java version / font / PLAF. Now everything is broken. What's going on?

This sounds as if you don't use layout managers, but instead hard-coded component sizes and widgets. If you want to avoid this problem, there is no way around using layout managers, or implementing your own geometry management from scratch.

Q3.4 My graphics on a Canvas/JPanel/JComponent, etc. gets corrupted, or I get a null pointer exception when trying to draw. How can I avoid this?

Do not use `Component.getGraphics()`. Instead, subclass and override the `paint()` (AWT), or `paintComponent()` (Swing) method.

`Component.getGraphics()` simply can't work. Java uses a callback mechanism for drawing graphics. You are not supposed to "push" graphics information into a component using `getGraphics()`. Instead you are supposed to wait until Java calls your `paint()/paintComponent()` method. At that moment you are supposed to provide the Component with the drawings you would like to do.

This mechanism is necessary so Java can support graphics systems which don't remember a window's contents when it is obscured (e.g. overlaid by another window). When the window becomes visible again, such graphics systems have to ask the application to reconstruct the window content. Therefore, `paint()/paintComponent()` is supposed to be the memory of a component. `getGraphics()`, however, doesn't have any recollection of previous drawing operations. So once a drawing done via `getGraphics()` is lost, it can't be reconstructed. There is nothing in there that stores the old drawing data, and there is nothing in AWT/Swing which informs `getGraphics()` to do some re-drawing.

In addition, there are situations where `Component.getGraphics()` simply returns null. This is a defined behavior of the method. And finally, most users of `getGraphics()` forget to dispose the `Graphics` object after usage.

See

<http://java.sun.com/products/jfc/tsc/articles/painting/index.html>

for more information.

Q3.5 How to create a transparent or non-rectangular window?

1) You can't in a good, platform independent way.

Although particular Java components can be 'transparent', without native support they will be contained inside a rectangular root component that is not transparent.

2) One hack is to take a snapshot of the underlying screen region using `java.awt.Robot.createScreenCapture(rectangle)`. And then using that snapshot as a background image for the window. If the background changes, the illusion is gone.

3) The `nativeskin.jar` of the `Skin LnF` at

<https://skinlf.dev.java.net/>

provides a Windows-only region feature for building non-rectangular windows. It comes with a native Win32 library, so applications written with this library are not portable to other platforms.

It should in principle be possible to write a similar library for other GUI systems (e.g. X11 with the very common shape extension).

4) An alternative for applets can be found here:

<http://java.sun.com/docs/books/faq/src/app/MatchBackgroundExample.html>

See also: "Q6.1.2 How to create a transparent widget?"  
"Q6.1.3 How to create a non-rectangular widget?"

#### 4 Architecture

~~~~~

Q4.1 What is this Model-View-Controller (MVC) stuff?

MVC is a way to structure an application. It is based on the idea of separating the presentation of data from the data itself. MVC originated in the Smalltalk world and has since then become a common design pattern.

Swing uses a variant of MVC (not THE original MVC as Smalltalk users will point out).

The following TSC article contains a good description of Swing's architecture and MVC variant:

<http://www.javadesktop.org/tsc/articles/architecture/index.html>

Q4.2 What is the Swing single-threading issue?

There is really no issue. With very few exceptions Swing is not thread safe. But it can be safely used in a multi-threaded environment if the necessary precautions are taken.

There are three simple things to take care of, not more:

1. Call all Swing API methods from the event dispatching thread (EDT), unless the API documentation states that a method can be called from another thread (is thread save).

NOTE: The API documentation has been known to be wrong in the past. Check Sun's bug parade when in doubt.

2. If you are not in the EDT, but need to call a Swing API method, use either `invokeAndWait()` or `invokeLater()` to schedule your code for execution on the EDT.
3. Do not block the EDT. That is, don't run time-consuming tasks on the EDT. Instead, run these tasks in a separate thread and use an `invoke...()` method to update the GUI from that separate thread.

There is really nothing more to it.

See also: "Q3.1 My GUI freezes or doesn't update. What to do?"
"Q3.2 How do I update the GUI from another thread?"
"Q4.3 What is the right way to start a Swing GUI?"

Q4.3 What is the right way to start a Swing GUI?

Sun almost silently changed the recommended GUI startup procedure. At the beginning of 2004 the examples in Sun's GUI tutorial were changed, and some rather short "explanation" was given. The explanation can be summarized as: "There is a threading bug somewhere in Swing. To work around, already build and start the GUI from the EDT". No more useful information is provided, e.g. if Sun knows the root cause of the bug and intends to fix it.

So now the official way to build and start (calling setVisible(true)) the GUI is to use invokeLater(). The shortest version of a GUI application's main() method becomes:

```
public static void main(String[] args) {
    invokeLater(new Runnable() {
        public void run() {
            //
            // Build and start the GUI here.
            //
        }
    });
}
```

See also: "Q4.2 What is the Swing single-threading issue?"

5 Window / [J]Frame / [J]Dialog (Top-Level Containers)

Q5.1 How can I ensure a window is always on top of all other windows using AWT or Swing?

1) Before Java 1.5 you couldn't at all:

AWT and Swing didn't provide this feature. All you could do was to use a (modal) [J]Dialog, and make sure the [J]Dialog is provided with the correct parent/owner in the constructor.

2) Since Java 1.5:

You have Window.setAlwaysOnTop(), which is inherited by the other top-level containers like JFrame. However, the implementation is incomplete, and not supported on all platforms. There are especially problems with a number of Unix version / Window manager version combinations.

Q5.2 How can I (de)iconify a window?

1) Before Java 1.2 you had to revert to native calls.

2) Since Java 1.2 you can use [J]Frame.setState().

3) Since Java 1.4 you can use `[J]Frame.setExtendedState()`, too.
`setExtendedState()` provides more features than `setState()`.

Q5.3 How can I replace/remove the icon in the title bar (window decoration) of a `[J]Frame`?

Use `setIconImage()`.

To revert to the platform's default icon use:

```
frame.setIconImage(null);
```

On some platforms this might remove the icon. Alternatively you can try a transparent `Image` if you don't want to have an icon.

Q5.4 How to replace the icon in the title bar (window decoration) of a `[J]Dialog`?

There is only a partial solution to this problem, and it is not recommended.

A dialog gets its icon from its parent frame. You can create a dummy frame which you don't show, set the icon of that dummy frame, and use it in the constructor of the dialog as the dialog's owner:

```
JFrame dummy = new JFrame();  
Image icon = ...  
dummyFrame.setIconImage(icon);  
JDialog dialog = new JDialog(dummy);
```

However, this is dangerous. Certain GUI behavior depends on a correct `[J]Frame` (parent window) \leftrightarrow `[J]Dialog` (child window) relation. Introducing a dummy parent breaks this relation. Things which can go wrong include (de)iconising of all windows of an application, and ensuring a modal dialog is always placed on-top of the main window.

Q5.5 My modal dialog goes behind the main window. How can I ensure it is in front instead?

Make sure you have properly set up the 'owner' of the dialog in the dialog's constructor. Don't use null, and don't use a dummy frame (to set the dialog's icon).

Q5.6 How to bind the escape key to the `JDialog` cancel operation?

First the bad news. Some `JComponents`, like `JComboBox` or `JTable` use the escape key by themselves. As a result, the following will not work, unless you manage to remove the escape key handling from these components.

To bind the escape key to some operation use code like it follows:

```
final String ESC_ACTION_KEY = "ESC_ACTION_KEY";
theDialog.getRootPane().getActionMap().put(
    ESC_ACTION_KEY,
    new AbstractAction() {
        public void actionPerformed(ActionEvent e) {
            //
            // perform the action associated with the escape key
            //
            // e.g. inform all window listeners, or just call
            // dispose()
            //
        }
    });
```

```
theDialog.getRootPane().getInputMap(
    JComponent.WHEN_ANCESTOR_OF_FOCUSED_COMPONENT
).put(
    KeyStroke.getKeyStroke(KeyEvent.VK_ESCAPE, 0),
    ESC_ACTION_KEY
);
```

Q5.7 How can I implement my own JFrame/JDialog close handling?

Ironically, by setting the JDialog, JFrame, or JInternalFrame default close operation to do nothing:

```
theJDialog.setDefaultCloseOperation(JDialog.DO_NOTHING_ON_CLOSE);
```

And then adding an own window listener:

```
theJDialog.addWindowListener(new WindowAdapter() {
    public void windowClosing(WindowEvent we) {
        //
        // Ask for closing confirmation,
        // perform any necessary cleanup, etc. here.
        // Close window if desired.
        //
        // we.getWindow().dispose(); // close window
    }
});
```

Q5.8 How Do I center a window on the screen? How do I get the screen size?

1) Manually, pre 1.4:

Use `java.awt.Toolkit.getDefaultToolkit().getScreenSize()` to get the screen size, and do the math:

```
import java.awt.*;
Dimension winSize = win.getSize();
```

```
Dimension screenSize =  
    Toolkit.getDefaultToolkit().getScreenSize();  
win.setLocation(  
    (screenSize.width - winSize.width) / 2,  
    (screenSize.height - winSize.height) / 2  
);
```

2) Since 1.4:

```
Window.setLocationRelativeTo(null);
```

Q5.9 How to ensure a minimum or maximum window size?

You can't in a good way. Swing and AWT ignore the `minimumSize` and `maximumSize` attributes. E.g. if your GUI system allows you to interactively resize a window to the size of 1 x 1 pixels Swing/AWT will happily allow this.

A few ugly hacks are possible to correct the size after the the unwanted resizing happened. They are ugly, because the window snaps back, due to the fact that the original resizing is not prevent but just corrected with another resizing.

This correction can e.g. be done with an event listener, listening to window component resizing events. Or it can be done by overriding the `doLayout()` method. In both cases, the actual size can be obtained with `getWidth()/getHeight()` and, if the size is not acceptable, corrected with `setSize()` (which triggers the second resizing and makes the window jump).

See also: "Q5.10 How to ensure a particular aspect ration of a window?"

Q5.10 How to ensure a particular aspect ration of a window?

You can't in a good way. You can use similar hacks as the ones for enforcing a minimum or maximum window size.

See also: "Q5.9 How to ensure a minimum or maximum window size?"

Q5.11 How can I delegate the window placement to the window system or manager?

First of all, you of course need to use a window manager or system which is capable of placing the windows by its own. The rest depends on the Java version:

1) Before Java 1.5:

You have no control over the behavior. If you don't specify an explicit window position this might be interpreted as the position (0, 0) and the window might be placed there, or it might be

interpreted as a request to position the window where it best fits.

The same might happen when you explicitly specify (0, 0) as the position. Some systems interpret this as a request to place the window where it best fits, others just position the window at (0, 0).

2) Since Java 1.5:

Sun has added a hack for specifying the desired behavior. If the `java.awt.Window.locationByPlatform` system property is set to true, a window manager can place a window with the origin (0, 0) or no specified origin at will. If the property is false, a window with the origin (0, 0) is indeed positioned at (0, 0).

The behavior can also be specified on a per window, per `setVisible(true)` basis by using `Window.locationByPlatform(true)` immediately before `setVisible(true)`.

Q5.12 I need to take some toolbar (dock, panel) size into account when calculating a window position an/or size. How?

The information is not available on all platforms. If you are lucky, `Toolkit.getScreenInsets()` delivers information about the screen border space occupied by the toolbar (dock, panel, or whatever your window system uses). You can use this information in your calculations.

```
Insets i = theWindow.getToolkit().getScreenInsets(  
    theWindow.getGraphicsConfiguration()  
);
```

Do not assume that the desired size is e.g. always in `Insets.bottom`. Many graphics systems allow to move the toolbar to the top or a side of the screen, too. Or they allow to have multiple toolbars. Always take all four values into account.

In general, for cross-platform compatibility, it is best to not rely on this information at all. If it is essential for your application then at least provide some configuration means to manually specify some screen border space. This way people on systems where the information is not available can still use your application as desired by you.

See also: "Q6.1.4 What are Insets?"

6 [J]Component (Widgets)

~~~~~

### 6.1 General Questions

#### Q6.1.1 How do I position components (widgets) on a window?

You add them to [J]Panels and use one or more of the many layout managers that come with Java (one for each [J]Panel). And you leave the exact calculation of the position to the layout managers.

Learning the layout managers is essential to AWT and Swing programming. Many people don't buy this and think they can get away without. Later they come to the newsgroup and whine because their GUIs don't work on other platforms, or don't look good when resized. But whining will not change their GUIs or the way AWT/Swing works. You have been warned.

See and work through:

<http://java.sun.com/books/tutorial/uiswing/layout/index.html>

#### Q6.1.2 How to create a transparent widget?

Since the introduction of the lightweight framework in Java 1.1 it is possible for a Component subclasses (if lightweight) to have transparent parts if properly implemented. Swing components are a special case, since here most of the existing components can be switched to have a transparent background.

So there are three different cases:

##### 1) Component (excluding Swing and heavyweight) subclasses

All it takes is a subclass of Component with a paint() implementation that only draws the non-transparent parts.

If an existing lightweight Component is subclassed to build a transparent component, it should be made sure that the superclass doesn't fill the background with the background color in the paint() method.

##### 2) JComponent (Swing) subclasses

The Swing JComponent subclasses can usually be switched between painting all pixels within their boundaries or not, effectively allowing the programmer to control if the JComponent's background is non-transparent or transparent.

This behavior is a side-effect of the opaque contract between JComponents and the Swing repaint manager.

```
setOpaque(true)
```

kindly ask a JComponent not to paint all pixels within its boundaries. For most, but not all, Swing components this results in a transparent background.

Setting `setOpaque(true)` doesn't work for all JComponents. E.g. some contain nested component (e.g. `JScrollPane`) which need to be changed to transparent, too. Or some PLAFs ignore the `opaque` attribute and always paint every pixel within their bounds. This behavior is not illegal, it is still covered by the `opaque` contract. But it means that the desired side-effect (transparency) of setting `opaque` to `true` will not work.

### 3) Heavyweight Components

Transparent heavyweight components are not supported in Java.

See also: "Q6.1.3 How to create a non-rectangular widget?"

"Q3.5 How to create a transparent or non-rectangular window?"

#### Q6.1.3 How to create a non-rectangular widget?

The boundaries of a Component (widget) in Java are always rectangular. Non-rectangular Components can be faked by implementing Components with a transparent background.

Java will still treat these components as rectangular objects. E.g. The standard layout managers always use the rectangular boundaries to layout Components. If another behavior is required for transparent Components (e.g. partly overlapping boundaries) custom layout managers need to be implemented.

See also: "Q6.1.2 How to create a transparent widget?"

"Q3.5 How to create a transparent or non-rectangular window?"

#### Q6.1.4 What are Insets?

##### 1) Simple answer:

`Insets` is a data type, describing additional space around some type of rectangle space, e.g. a component. A better choice of words would have been border space.

##### 2) More detailed answer:

There is no consistent usage of the term or the `Insets` data type in AWT or Swing. Originally `Insets` were used in AWT to describe the size of window decorations, like the size of a `Frame`'s title bar, and borders in a `GridBagLayout`. Over the time `Insets` popped up in all sorts of places in AWT and Swing. Often there is no exact definition of what an `Insets` data type describes, especially in relation to the size of an item.

For a particular usage of `Insets` one has to figure out if the `Insets` describe additional or subtractive space. Roughly, one has to figure out, often by experiment, which of the following holds true for a

particular situation:

usable size = item size – insets  
total size = item size

or

usable size = item size  
total size = item size + inset

The exact relation doesn't matter in the trivial case

Insets = {0, 0, 0, 0}

Q6.1.5 How do I find a component's top-level container (e.g. the window)?

Use

```
SwingUtilities.getRoot()
```

## 6.2 JTree

Q6.2.1 I changed the data / structure for my JTree, but the display doesn't get updated. What's going on?

Most likely you are directly manipulating the `TreeNode`s, instead of updating the data via the `TreeModel`. `TreeNode`s don't have any means to inform the `JTree` about changes. This is the job of the `TreeModel`.

If you use `DefaultTreeModel`, all the event notification mechanisms are already implemented. If you use an own implementation of `TreeModel`, you need to implement the necessary event firing yourself.

Don't call `repaint()` on the `JTree`. The `JTree` painting is not broken. Your event notification is. Get your event notification right.

Q6.2.2 How do I set a custom icon for a node?

If you just want to change the icon for all nodes from the default, get a `DefaultCellRenderer`, use the `set...Icon()` methods, and set the renderer to be used by the `JTree`.

If you, however, need different icons for different nodes, then the following three steps are involved.

1. Create your own `TreeNode` implementation (or subclass an existing `TreeNode` implementation).

- a. Add some means to the `TreeNode` to identify itself, e.g.

```
public int getType() { // return a type id }
```

- b. Or let the `TreeNode` return a corresponding `Icon` (possibly a shared instance):

```
public Icon getLeafIcon() { /* return icon */ }
public Icon getOpenIcon() { /* return icon */ }
public Icon getClosedIcon() { /* return icon */ }
```

- c. Or just rely on the type of your subclass (to be checked with `instanceof`). You will have to have an own subclass for each node which needs a different icon.

## 2. Subclass `DefaultTreeCellRenderer`. Override `getTreeCellRendererComponent()`:

### 2.1. Start with

```
Component getTreeCellRendererComponent(
    JTree tree, Object value,
    boolean sel, boolean expanded, boolean leaf,
    int row,
    boolean hasFocus)
{
```

And continue, depending on your `TreeNode` implementation:

### 2.2a If you use a `getType()` method:

```
YourTreeNode node = (YourTreeNode)value;
// Use a lookup table (faster), some factory, or:
switch(node.getType()) {
case: ...
    setLeafIcon(...);
    setOpenIcon(...);
    setClosedIcon(...);
    break;
}
```

### 2.2b If you use a `getIcon()` method:

```
YourTreeNode node = (YourTreeNode)value;
setLeafIcon(node.getLeafIcon());
setOpenIcon(node.getOpenIcon());
setClosedIcon(node.getClosedIcon());
```

### 2.2c If you rely on the type:

```
if(value instanceof YourTreeNode) {
    setLeafIcon(...);
    setOpenIcon(...);
    setClosedIcon(...);
} else if(value instanceof ...) ...
```

2.3. And finish with:

```
return super.getTreeCellRendererComponent(  
    tree, value,  
    sel, expanded, leaf,  
    row,  
    hasFocus);  
  
}
```

3. Use `JTree.setCellRenderer()` to set your renderer.

Q6.2.3 How do I remove all my nodes from a `JTree` at once?

Just replace the model. Deleting all nodes individually is a waste of time.

`JTree`'s API documentation does not indicate if it is permissible to use null as a model, but it is known to work in Sun's reference implementation:

```
tree.setModel(null);
```

If you don't trust his code, create an empty model and use it instead of null.

6.3 Styled Text / `JEditorPane` / `JTextPane`

Q6.3.1 Can I use `RTFEditorKit` to read RTF documents created by Word?

Well, you can try. The `RTFEditorKit` is, however, very limited. There is a subtle hint in the `RTFEditorKit` API documentation. It points out that the Swing team "hops" to improve the class in the future. This hint is there since the first release of the class years ago.

The newer your Word is, the less likely it is that the `RTFEditorKit` can read the RTF. There have been reports about crashes when using RTF generated by the latest Word version.

Q6.3.2 I have problems using the Swing HTML parser to parse all kinds of HTML. Is this normal?

Unfortunately it is. The Swing HTML parser is the old HotJava parser (Sun's pure Java web browser, once a separate product). It is limited and hasn't been updated for a long time. In principle it can deal with HTML 3.2 and stylesheets. In practice it is picky.

Use another parser if you have problems, or convert your HTML with HTML Tidy or its Java port `JTidy` before trying to read it.

<http://tidy.sourceforge.net/>  
<http://jtidy.sourceforge.net/>

Q6.3.3 Some of my CSS styles don't work out. Is this normal?

Yes, for the same reasons as described in the previous answer. See the class `javax.swing.text.html.CSS` for a list of the officially supported CSS attributes, and brace yourself for some more deviations.

Q6.3.4 Can I use Swing's HTML support to write a web browser?

You can, but the resulting browser will suffer from the limitations of Swing's HTML parser and CSS handling. The parser was in fact originally developed for Sun's HotJava web browser. But this was years ago, and HotJava was never a serious competitor in the browser business.

To get some ideas have a look at

<http://java.sun.com/developer/onlineTraining/GUI/Swing1/shortcourse.html#JFCEditorPane>

Q6.3.5 Can I use Swing's HTML support to build an on-line help system or e-book?

Sure you can. Or, you could take Sun's JavaHelp, which does exactly this. It uses the Swing HTML components to parse and render help text or other text written in HTML. It also provided for navigation and other common help features.

Q6.3.6 If HTML support is really so broken in Java, what is it good for?

As long as you have the generation of the HTML under your control it is quite usable. E.g. the JavaHelp system uses Swing's HTML parser and display capabilities.

If you need to handle real-world HTML from sources not under your control, you better look for some other parser.

## 6.4 [J]TextArea

Q6.4.1 I append text to a JTextArea. How to ensure the text area is always scrolled down to the end of the text?

```
textarea.setCaretPosition(textarea.getDocument().getLength());
```

Q6.4.2 How to use several different fonts (styles, sizes) in one [J]TextArea?

You can't. Use a J[Editor|Text]Pane.

## 6.5 [J]Label / [J]Button

### Q6.5.1 How can I have multiple Lines in a [J]Label?

#### 1) Label

You can't. Use a `TextArea` instead, and turn editing of the `TextArea` off.

#### 2) JLabel

2.a) Use HTML markup in the label's Text. E.g.:

```
theJLabel.setText("<html>Line 1<br>Line 2<br>Line 3</html>");
```

2.b) Or use a `JTextArea`.

### Q6.5.2 I want to have a hyperlink in a [J]Label. How can I do this?

You probably don't want to have a label, but a button.

Just configure a `JButton` so it looks like a link. E.g. use HTML to format the button's text, and remove the button's border:

```
b = new JButton("<html><a>http://java.sun.com</html>");  
b.setBorderPainted(false);
```

Provide the button with an `ActionListener` to handle the `ActionEvent` when the button is clicked. Implement whatever is desired in the event handler. E.g. start an external web browser.

```
b.addActionListener(new ActionListener() {
```

Please mail suggestions, corrections, updates, etc. to the author "cljg\_faq" at the host "gmx.de". Your "Subject:" line must contain the string

```
[cljg]
```

somewhere in the line, including the square brackets. Otherwise your mail will be discarded automatically. In addition, the address is heavily spam-protected. So if you mail from a spam-invested network, there is little chance to reach the author. Your alternative is to post to c.l.j.g.

If you suggest a new entry, please also provide the answer, not only the question.

## 11 Acknowledgments

~~~~~

comp.lang.java.gui: comp.lang.java.gui FAQ

This FAQ contains contributions and help from:

Andrew Thompson, David Postill, Manish Hatwalne, Hiwa.

###