Start Here!

500 Java Tips

JavaFAQ.nu

?!! javaFAQ.nu
JavaQ:A

Hello dear friend!

I am very glad you are here!

"500 Java Tips" is my collection of good questions and answers from numerous Java fora and newsletters.

Please send me your feedback, critical opinions and wishes! Please visit my site at: http://JavaFAQ.nu !

Receive our newsletter with new tips! More than 9,000 subscribers (by 10 January 2002) can not be wrong! They read our tips every week!

To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to: javafaqtips-request@javafaq.nu

or on the web:

http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

Contents at a Glance

```````````````````````````````````````````````

# Contents at a Glance

### Our Media Kit
Excuse me for possible mistakes! English is not native language for me.
I will be glad if you send me your corrections of my mistakes!

**JOIN to**
**Java IT NET!**

# Instant Job Search

Member Login

E-mail:

Password:

(New User)

**Member Login**

**New User**

**For Employers**

(boolean help)

All words      Any words

Return To...
Career Center
Absolute Java FAQ

Job Search Tools
Employer Profiles
Detailed Search

Career Guide
Career News
Resume Writing
Interviewing
Negotiations
Letter Writing

Resource Pages
UserGroups/SIGs
Training Resources
Career Coaches

Expert's Corner
On Resume
On Retirement
On Immigration
On Incorporation

Employer Tools
Client's Log-In

QuickAgent: Jobs via Email*

*You don't need to be a registered member to sign up for QuickAgent.

### Job Seeker Tools

#### New Users
Not a member yet? What are you waiting for? Register and get instant access to our quality pool of IT openings. Your dream IT job is just a few keystrokes away.

#### Candidates Login
Login and access our IT specific database of openings. Thousands of IT jobs and hundreds of the hottest employers are all looking for you.

#### Resume Builder
Build your resume and get the ball rolling on your search process.

#### Tech Agent
Tech-Agent is your automated job search function. Select criteria for your ideal job, and we will email matches directly to you every week.

### Employer Tools

#### Employers
Pinpoint the hottest IT talent around with ease and efficiency. This is the one-stop-shop for all your IT recruiting needs.

Positions | Profile

The Farmers Insurance Group of Companies was founded in 1928 when Farmers Insurance Exchange, an automobile insurer, was formed. As customer demand for additional insurance services increased, the Fire Insurance Exchange and Truck Insurance Exchange were established for home and commercial insurance needs.

Positions | Profile

Founded November 1980 Thomson Financial is a US$1.44 billion provider of information services and work solutions to the financial and corporate communities worldwide. Through the widest range of products and services in the industry, Thomson Financial helps clients in more than 70 countries make better decisions, be more productive and achieve superior results.

[Positions](#) | [Profile](#)

Axiom Group is a leading supplier of technical market data outsourcing services for financial trading floors. Our staff of service professionals brings to our clients many years of expertise in the implementation and technical support of trading floor market data operations.

# Applets

Q: What are restrictions for an applet? What are applets prevented from doing?
Answer: In general, applets loaded over the net are prevented from reading and writing files on the client file system, and from making network connections except to the originating host.

In addition, applets loaded over the net are prevented from starting other programs on the client. Applets loaded over the net are also not allowed to load libraries, or to define native method calls. If an applet could define native method calls, that would give the applet direct access to the underlying computer.

Q: Do I need special server software to use applets?
Answer: No. Java applets may be served by any HTTP server. On the server side they are handled the same as any other file, such as a text, image, or sound file. All the special action happens when the applet class files are interpreted on the client side by a Java technology-enabled browser, such as HotJava browser or 1.x or Netscape 3.x/4.x.

source: http://java.sun.com/products/jdk/faq.html#A8

Q: I know that applets have limited possibility to do many things. It is about network connections, file reading/writhing and more.
Can applet read all system properties and if not how many of them are restricted?

Answer: Applets can read quite many of system properties by using:
String ss = System.getProperty(String key):

java.version
java.vendor
java.vendor.url
java.class.version
os.name
os.arch
os.version

file.separator
path.separator
line.separator

Applets are prevented from reading these system properties:

java.home
java.class.path
user.name
user.home
user.dir


source: http://java.sun.com/sfaq/
--
AP. (J.A.)

📋  Q: I write my first applet and it become very huge! It is an applet but looks like huge Java Application.
Could you point me what is most is important for having a small applet?

Answer:
1. Use compiler optimization: javac -O But check it the size anyway. Sometime it makes the code bigger..
2. Use jar files instead of class files
3. Try to use inheritance as much as possible: than more code you can reuse than less new lines you have to add.
4. Try to use standard APIs. Often they are better optimized in size than some private exotic packages. Of course often they have better methods and so on but try to use efficiently what we have already!
5. Use short names.
6. Do not initialize big arrays because. They will be initialized and put directly into bytecode. You can do it later on the fly

Please if you know more methods to make an applet smaller mail us and we will add your comments here!
--
AP. (J.A.)

📋  Q: Why do I get message like "wrong magic number" when I am trying to run applet? What is a magic number?

Answer: The first thing a JVM does when it loads a class is check that the first four bytes are (in hex) CA FE BA BE. This is the "magic number" and thats why you are getting that error, you are trying to load a file that isnt a class and so the class loader in the JVM is throwing out that exception.

Make sure you transfer the class files to site in binary mode, rather than text or ASCII mode.
An error from the browser saying "cannot start applet ... bad magic number" usually means that one of the class files on the server is corrupted. '
Replace your class binary files on the web server; clean up the cache of your browser, and reload your applet.

📋 Q: I've got problems with the Socket class (network)

I've got problems with the Socket class. I use it inside an applet (I've written a small chatbox). I have code like this:

Socket s = new Socket("192.168.0.4", 13780);

When the server I'm connecting to is on the same machine as the client, it works.
When the server is an other machine, both NS and IE give an error message like:

Security:Can't connect to 192.168.0.4 with origin ''

Does anyone know how I can fix this??

Answer: The standard security concept for an applet is the 'sandbox'. An applet can't talk outside it's memory space, can't talk to any files at all, and cannot talk to anything on the internet except the same machine that it's 'parent' HTML page originated from. So your applet can never talk to 192.168.0.4 unless the HTML came from 192.168.0.4

Q: How do I view the error output from my Java applets in IE?

Answer: The file windows\Java\Javalog.txt contains info about the last Applet loaded in IE. All the System.out messages and exception information is stored here when Java Logging is enabled in IE. To enable Java Logging start IE and select View/Options/Advanced. Select "Enable Java Logging" check box click OK. Restart IE.
In NT4 the file  in C:\WINNT\Java

Q: Is there a way to reduce the amount of time that it takes to download an applet?

Answer: There is a way to reduce the amount of time an applet takes to download. What ever classes the Java applet is refering, you cluster them in a JAR file with the help of JAR utility that comes with the JDK version. Check out the help for the options of that utility and make a ".jar" file out of the applets refered classes and images and other relevent data which you want to load.

Use the archive option of the applet tag and assign the .jar file:

```
<applet code="xyz.class" archieve="pqr.jar" width=100 height=100>
</applet>
```

Q: When I reload my applet my hidden canvas is shown directly! Why?

Answer: Put mycanvas.setVisible (false); in Start() rather than init()...

Q: I want to be able to print debugging text messages during the whole applet's lifetime. Is there an easy way to do that???

I'm a beginner in java. Right now i am doing an applet and i want to write messages to the browser window for debugging purposes i.e. to follow how the applet executes. Like when i'm developing an C++ application i usually use lots of "couts" to check values and the programs behavior. Is there an easy way to do things like that when making a Java applet? For me it seems like everything happens in a function called "paint(graphics g)" and that function is only called at the beginning of the applet start. I want to be able to print text messages during the whole applet's lifetime. Is there an easy way to do that???

Answer: you'd be better off doing a
System.out.println("the value is " + whateverValue);

This will show up in the java console. to see it in ie5, do View->Java Console, and in netscape4.7, do Communicator->Tools->Java Console and it will pop up the java console window.

If you are doing it in appletviewer from dos, it will show up in the dos window you used to call appletviewer.

Q: I am writing an applet that will use images. I would like to ship out the images using a jar file that contains all the images that the applet is going to use. I have seen a piece of code that does that in the past, but I don't remember where.

Answer: by David Risner  The following is from:
http://developer.netscape.com/docs/technote/java/getresource/getresource.html

```
import java.applet.*;
import java.awt.*;
import java.io.*;

public class ResourceDemoApplet extends Applet {
  Image m_image;

  public void init() {
    try {
      InputStream in = getClass().getResourceAsStream("my.gif");
      if (in == null) {
        System.err.println("Image not found.");
        return;
      }
      byte[] buffer = new byte[in.available()];
      in.read(buffer);
      m_image = Toolkit.getDefaultToolkit().createImage(buffer);
    } catch (java.io.IOException e) {
        System.err.println("Unable to read image.");
        e.printStackTrace();
    }
  }

  public void paint(Graphics g) {
    if (m_image == null)
    return;

    Dimension d = getSize();
    g.drawImage(m_image, 0, 0, d.width, d.height, Color.white, this);
  }
}
```

Q: I have made an applet in VJ++ which I have to sign. Is there any tool to do it (both signing and cabbing)..?

Answer: Signing and archive files are two of the biggest bothers in Java. Everyone uses a different system. A good place to start is:
http://www.suitable.com/Doc_CodeSigning.shtml
One of the other bothers is that the unsigned window warning can't be removed by

signing an applet for Internet Explorer for Macintosh. And while I am on the subject, the Windows Netscape 4.x system has a bunch of privilege calls:
http://developer.netscape.com/docs/manuals/signedobj/capsapi.html
and you need under most circumstances to make Microsoft specific calls too, detailed in links from:
http://www.microsoft.com/java/security/

Going through all this will make you want to curse. Unfortunately it is hard to pick a convincing scapegoat. It is true that Microsoft chose an entirely nonstandard CAB system, but it produces archives that are about 40% smaller than JAR files. Signing archive files is a perfect microcosm of the "freedom to innovate" controversy. Microsoft has done a better job but taken away predictability and uniformity. If the Java standards were not controlled entirely by Sun, a Microsoft competitor, perhaps everyone would be using smaller archive files by now.
--
Mickey Segal

Q: I want to use more fonts in my applet... say for example Arial... which is not avilable in the present jdk package...

 How can i deal with it?

Answer: import java.awt.Toolkit;
....
Toolkit tools = new Toolkit();
String[] fontList = tools.getFontList();

Q: How can I slow down my applet?

I have a game applet that is running too fast on newer systems that have high-end video cards. Its easy enough to slow down the game by having it sleep between thread cycles, but I need to be able to
determine how fast a users machine is before I determine how long to sleep for.

I have been muddling through the documentation but cannot find any calls that will tell my applet what the users configuration is as regards to CPU speed and other components they may have on their system.

Answer: Simple create a new Date (), then perform a standard lengthy operation on the order of something that takes about one second on your machine, like a long loop, then create another new Date() and compare it to the first. If it takes 1/2 of the time compared to your machine, then the CPU is probably about 2 times faster. if it takes 3 times the duration compared to your machine, the CPU is probably 1/3 as fast as yours.

Do this dynamically, and it might help with speed changes when there's lots of action happening as well - unless this issue is already being dealt with using threads, that is.
--
by Max Polk

Q: Why do I see applet in applet viewer and do not in a browser?

When I try to view my applet on a web page i get the error

java.lang.NoSuchMethodError: java/lang/Double: method
parseDouble(Ljava/lang/String;)D not found

Which is weird as it compiles fine on Borland and with the JDK using applet viewer
Anyone have any ideas what is going wrong?

Answer: The parseDouble method was only added to Java in JDK 1.2
Browsers typically only support Java 1.1
If you have the JRE installed, you can run Java 1.2 applets. But you must also change the HTML code that embeds the applet. Check javasoft.com. I believe they have a program which will automatically change the <APPLET> tag to <EMBED> and add whatever else is needed. It's been a while since I've done applets but I do remember running across a similar problem.

Q: In my applet I have a bunch of gif's in my JAR file. When I try to access a gif using:

Image img = getImage(getCodeBase(), "image.gif");

everything works fine under Microsoft Internet Explorer but it does not under Netscape and appletviewer. Of course I do not have any gifs in my CodeBase directory on server.

Any idea why?????
Answer:  Because this is not how you access resources in a Jar file. You need to use getResourceAsStream if you want to access GIFs from Netscape. Look at:
http://developer.iplanet.com/docs/technote/java/getresource/getresource.html
for example code. This same code will work in Sun's Appletviewer.
--
David Risner
http://david.risner.org/

Q: How do I get JVM version in Internet Explorer?

When you open the Java Console through internet explorer, it prints the following useful line at the top:

Microsoft (R) VM for Java, 5.0 Release 5.0.0.3318

We would like to be able to obtain the above String (or atleast the 5.0.0.3318 part of it) through a Java Applet / Javascript at runtime.
Does anyone know of any handy methods that allow access to this String ? I've looked in all the System.properties, but it wasn't there. Is it stored in the user's registry anywhere ?

Answer: just for Microsoft't VM!
try :

```
class test{
  public static void main(String[] args){
    String build;
    build=com.ms.util.SystemVersionManager.getVMVersion().getProperty ("BuildIncrement");
    System.out.println("Using build "+build);
  }
}
```

Real Gagnon from Quebec, Canada
* Looking for code code snippets ? Visit Real's How-to
* http://www.rgagnon.com/howto.html

Q: I wonder if there is a way to find out if a button in an applet has been clicked, no matter which

of the buttons in an applet it might be.

Of course I can write, with a particular button (if event.target==button1) but maybe there is a syntax that looks more or less like this (it is an imaginary code just to show what I would like to do)
(if.event.target.ComponentType==Button) etc.
I tried a lot of things with getClass but none of them worked

Answer: Have your applet implement the ActionListener interface, and have every button that's instantiated add the applet as an ActionListener. Then, inside of your applet, have the following method:

```
public void actionPerformed(ActionEvent event) {
   // check to see if the source of the event was a button
   if(event.getSource() instanceof Button) {
      // do whatever it is you want to do with buttons...
   }
}
```

Darryl L. Pierce Visit <http://welcome.to/mcpierce>

Q: Could you suggest how to draw one centimeter grid in applet, please? One cm on the screen must be equal to real cm.

Answer: If you're not all that picky about it, you can always use java.awt.Toolkit's getScreenResolution() to see how far between the lines should be in the grid....that's assuming the applet security allows it.

But have it _exactly_ one cm, you can't do, since the user can always adjust the display with the monitor controls (making the picture wider/taller/whatever), and no computer that I know of can know those settings.
--
Fredrik Lännergren

Not only that, the OS (and thus Java) does not know if I am using a 21" or a 14" monitor and thus can't know the actual physical size of a given number of pixels. By convention, on Windows monitors are assumed to be either 96dpi or 120dpi (depending on the selection of large or small fonts). Java usually assumes 72dpi. None of these values is likely to be
accurate.
--
Mark Thornton

📑 Q: Does anyone know how to or where I can find information about determining if cookies are disabled on a client browser making a request to a servlet or JSP (or any server side request handler, for that matter)? Also, is there a way to determine whether or not a client's browser has style sheets enabled?

Answer: To test if the client has cookies enabled, create a cookie, send it, and read it back. If you can't read it back, then the client does not accept them. It's not a clean way of doing it, but it's the only way (that I know if).

As for CSS, there is no way to know if they allow CSS. Different versions of the browsers support varying levels of CSS. You can get the browser type from the request object and then make decisions based on that.

📑 Q: How can two applets communicate with each other? Have you some examples?

Answer: You will occasionally need to allow two or more applets on a Web page to communicate with each other. Because the applets all run within the same Java context-that is, they are all in the same virtual machine together-applets can invoke each other's methods. The AppletContext class has methods for locating another applet by name, or retrieving all the applets in the current runtime environment

---------------------------------------------------------------

```java
import java.applet.*;
import java.awt.*;
import java.util.*;

// This applet demonstrates the use of the getApplets method to
// get an enumeration of the current applets.

public class ListApplets extends Applet {
  public void init() {
    // Get an enumeration all the applets in the runtime environment
    Enumeration e = getAppletContext().getApplets();
    // Create a scrolling list for the applet names
    List appList = new List();
    while (e.hasMoreElements()) {
      // Get the next applet
      Applet app = (Applet) e.nextElement();
      // Store the name of the applet's class in the scrolling list
      appList.addItem(app.getClass().getName());
    }
    add(appList);
  }
}
```

I hope that did it!
by 11037803

Here are some useful links on applet to applet communication. I don't know if they will solve your problem but these are a variety of good approaches for this type of issue.
http://www.javaworld.com/javaworld/javatips/jw-javatip101.html
http://www.twf.ro/calculatoare/TricksJavaProgramGurus/ch1.htm
http://www.galasoft-lb.ch/myjava/CommTest/backup00/

http://www.rgagnon.com/javadetails/java-0181.html
http://www.2nu.com/Doug/FAQs/InterframeIAC.html
by Mickey Segal

📋 Q: I would like to ask if there 's anyway that I can use the same program run as an applet or application?

Answer: You would have to provide at least a main() for the application part, and init(), start(), stop(), destroy() for the applet part of your program. Your class could simply display the applet within a Frame.

Example:

```
class Foo extends Frame {
  public Foo(String title){
     //...
     Foo applet = new Foo();
     applet.start();
     add(applet, "Center");
     //...
  }
```

main()is function of course, not constructor
--
Alex

📋 Q: Is it possible to run a java applet in a dos window (win98 se)?

Answer: No. A dos window is a character device. You can use the applet viewer program that comes with the JDK though.
--
Mike

📋 Q: Is there a simple way to tell if a PC online or not from within an applet?

Answer: Not without either server-side support or signing the applet, since applets are not allowed to connect to other hosts than the one they are downloaded from. Best approach, I suppose, would be to ping the target from the server.
However, this is not quite full proof because of firewalling: my pc, for example, will not answer to pings.
--
Michiel

📋 Q: Is it possible to close browser from applet?

Answer: Yes, use this (tested):

```
//////////////////////////////////////////////////
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;
import netscape.javascript.JSObject;

class CloseApplet extends Applet
implements ActionListener{
   protected Button closeButton = null;
```

```
    protected JSObject win = null;

    public void init(){
        this.win = JSObject.getWindow(this);
        this.closeButton = new Button("Close Browser Window");
        this.add(this.closeButton);
        this.closeButton.addActionListener(this);
    } // ends init(void)

    public void actionPerformed(ActionEvent ae){
        this.win.eval("self.close();");
    }

} // ends class CloseApplet
/////////////////////////////////////////////
```

and the HTML needs to have MAYSCRIPT enabled.

```
/////////////////////////////////////////////
<HTML>

<HEAD>
<TITLE>Integre Technical Publishing</TITLE>
</HEAD>

<BODY BGCOLOR="#FFFFFF">
<DIV ALIGN="CENTER">
<APPLET WIDTH="150" HEIGHT="30" CODE="CloseApplet.class"
    CODEBASE="java/" MAYSCRIPT>
</APPLET>
</DIV>
</BODY>

</HTML>
/////////////////////////////////////////////
```

Here's the API:
<http://home.netscape.com/eng/mozilla/3.0/handbook/plugins/doc/Package-netscape.javascript.html>

It's small enough that you could include it in your JAR if you'd like. But most users will even have it on their systems.
It says "Netscape," but I know that IE understands it fine.
--
Greg Faron
Integre Technical Publishing

Q: Is it possible to run an Applet inside a JAVA application?

Answer: An applet is just another class that can be instantiated:
Applet myApplet = new MyApplet();

where MyApplet is the name of the applet class that you have written and then added to a container of

some kind
myFrame.add(myApplet);

..but you need explicitly call the init() method that a browser would normally call "behind the scenes":
myApplet.init();
--
artntek

Q: I want to change the size(Width x Height) of applet by mouse click on the Button. I used resize(), but it does not get bigger than the area defined by html file. Is there any way to do this?

Answer: I don't think so. You *could* (perhaps) open your own frame.
-- Tim Tyler

Q: I want to bypass the security sandbox so that my Applet loaded from IE can read and write files on the client side:

 (1)configure the server as trust side, (2)install my own security manager to override those checks. Will that work? Do I still need to have my Applet signed?

Answer: AFAIK this will not work. "Core Java II" says on page 716:

Once the program installs a security manager, any attempt to install a second security manager only succeeds if the first security manager agrees to be replaced. This is clearly essential; otherwise, a bad applet could install its own security manager.
Sign the applet and make the Security calls that request the access you need. If the user consents, your code will be allowed. If they don't, it'll fail.

Be aware that the Netscape security model is not that of the Java specification, and if you want to run there, too, you'll have to find a way of dealing with the discrepancy. Netscape's developer Web site includes the documentation for their model, as well as stub libraries you can compile against.

For information on the Java security model (Sun sanctioned version), check out these Web pages (chosen from the results of the Google search "Java Security Model"):

http://java.sun.com/security/

http://java.sun.com/security/SRM.html

http://www.sans.org/infosecFAQ/code/java_sec.htm
--
Randall Schulz, Michael Pellaton

Q: I just started to do Java programming. How can I bring the system standard output (what you see on your telnet window) to display in an applet?

I want to do:

system(myprogram);
then I want the screen output to be in my applet.

Answer: There is no system() method in the standard API. This functionality in Java is provided by the java.lang.Runtime class. Runtime allows you to capture standard output from an external process you started.

Q: How does one remove or replace the message "Java Applet Window" at the bottom of a frame?

Do not confuse this with the status message at bottom of a browser.

Answer: This can be done.

Use the .java.policy file in the user|home directory with following contents:

```
grant {
    permission java.awt.AWTPermission "showWindowWithoutWarningBanner";
};
```

--
Ganesh

**Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.**

☐ IT Professional    ☐ General Computer    ☐ Design and Graphics    ☐ Linux    ☐ Freeware/Shareware

☐ Games    Intranet    Computer Hardware    Web Design    ☐ Computer Software

☐ UNIX    ☐ Web Software    ☐ Windows Software    ☐ Personal Finance    ☐ Programming Software

☐ Servers    ☐ C/C++    ☐ Powerbuilder    ☐ Perl    ☐ XML

☐ Java    ☐ SQL    ☐ Design/Graphics    ☐ Utilities    ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

**500 Java Tips**

**JavaFAQ.nu**

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Code Examples

Q: I need to write a daemon process in Java so that there can only be one such daemon process per box. If the user tries to start another daemon process, the second one will detect there is already one running and quietly exit.

I know on NT I can run the daemon process as a NT service. But is there a way to achieve this on all platforms?

Answer: I suggest the following:

(1) Try giving all of your processes a unique ID.
System.currentTimeMillis() can be used for this task.

If you want this to be more secure on muli-processor-
or very fast systems, you can add a File.createFile()
mechanism. Here is some code that I use:

```
static long processID(){
  long id = System.currentTimeMillis();
  final int RETRIES = 10;
  try{
    for(int i = 0; i < RETRIES; i ++){
      File lockFile = new File("" + id);
      if(Deploy.createNewFile(lockFile)){
        try{
          Thread.currentThread().sleep(100);
        }catch(InterruptedException ie){ }
        lockFile.delete();
        break;
      } id = System.currentTimeMillis();
    }
  }catch(Exception ioe){}
  return id;
```

}

(2) Create a file on your computer to which each of your processes writes his ID in regular time intervals. You will want to have one thread in your application that does just that.

(3) This writing thread reads the current process ID from the file before he writes his own. If the ID is not his own, another process was started and he ends his process.

(4) Don't forget RandomAccessFile#getFD().sync(); (or your preferred file access method's sync) to make sure that the other processes data is written.

(5) You might mix the written ID with the currentTimeMillis for a little bit more information on possible dead sessions.

(6) When a new process is started and finds an active session writing it's IDs to the file, it needs to give the other process time to terminate.

Another possibility for interprocess-communication would be to use ports. Depending on how you deploy your application you might not be able to judge, which ones are available, so searching for a convenient port could ruin the setup.

I am not aware of possibilites for Daemon processes to monitor all JVM sessions but I would be glad to learn more.

Kind regards,
Carl
---
Carl Rosenberger
db4o - database for objects - http://www.db4o.com

Q: I'm developing an application which has to get data from the server. Now, a method will go online and fetch approximately 35000 lines from the server... That can take some time, so I'd like to "show" how far the progress is.

At first I had a (renamed to keep it simple)

retrieveAllData() method, but it took way too long to finish, and I'd really like to constantly have a value on how far I am.

Now I've thought about having a

```
public int retrieveNextDataChunk(int count) {
    in.readLine();
    // Do some stuff.
    return count++;
}
```

that way I could have:

```
int number = 0;
while (number = retrieveNextDataChunk(number) {
    System.out.println("We're at data chunk number: " + number);
}
```

But one of my colleagues told me that he thought it was a "hack".

How would you do it?

Answer: Use the Java event pattern. That is:

```java
interface ProgressListener{
    public void progressMade(ProgressEvent event);
}

class ProgressEvent{
    private int chunkNumber;
    public ProgressEvent(int chunk){
        this.chunkNumber = chunk;
    }

    public int getChunkNumber(){
        return chunkNumber;
    }
}

class Transfer{
    private List progressListeners = new LinkedList();

...

    public void retrieve() {
        int count = 0;
        String nextLine;

        while ((line = in.readLine()) != null){
            // do something with line
            fireProgressMade(count++);
        }
    }

public void addProgressListener(ProgressListener listener){
    progressListeners.add(listener);
}

public void removeProgressListener(ProgressListener listener){
    progressListeners.remove(listener);
}

protected void fireProgressMade(int dataChunk){
    ProgressEvent event = new ProgressEvent(dataChunk);
    Iterator listeners = progressListeners.iterator();
    while (iterators.hasNext()){
        ProgressListener next = (ProgressListener) listeners.next();
        next.progressMade(event);
    }
}
}
```

And the client code:

...

```
Transfer transfer = new Transfer();
transfer.addProgressListener(new ProgressListener() {
public void progressMade(ProgressEvent event){
System.out.println("We're at data chunk number: " +
event.getChunkNumber());
}
});

transfer.retrieve();
// get the data and go on
```

The advantage of this over your own solution is that the public interface is not unnecessarily complicated by the presence of progress monitoring. You still just retrieve the data and use it. Now, though, if you *do* want to monitor progress, you can do it, using code that's written quite separately from the actual transfer code.

You can plug in any implementation of ProgressListener, of course, such as one that updates a progress bar in a GUI presentation. I chose one that matches the behaviour you get from your short and quick solution.
--
Chris Smith

Q: Can anyone answer this - basic, it seems, despite which the answer eludes me completely - question: how do you have multiple windows in Java without using JDesktopPanes and JInternalFrames??
I don't want that kind of environment. I basically want to be able to press a button/menu option to open up a small menu of options/input/buttons like the tools->internet options menu of IE, and I have no idea how to do it.
Is it actually possible without using JDPs and JIFs?
Is it as simple as creating a separate class for the menu 'mini-window' and creating an instance of it from the main system?

Answer: The example you mention is just a fancy dialog. Read documentation on Dialog/JDialog. Also a single application can instantiate and display multiple top level containers such as Frames/JFrames.

For example
```java
import java.awt.event.*;
import javax.swing.*;

public class MultiFrameTest extends JFrame {
   int x = 0;
   int y = 0;

   public MultiFrameTest(){
      super("multiFrame test");
      setSize(200,200);
      JPanel mainPanel = new JPanel();
      setContentPane(mainPanel);
```

```java
      JButton addFrameBtn = new JButton("Add a frame");
      addFrameBtn.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e){
          addFrame(x,y);
        }
      });

      mainPanel.add(addFrameBtn);
      addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent we){
          System.exit(0);
        }
      });
      setVisible(true);
    }

  public void addFrame(int a, int b){
    JFrame frame = new JFrame("Frame "+a);
    frame.setSize(100,100);
    frame.setLocation( b* 10, b * 10);
    x++;
    y +=10;
    frame.setVisible(true);
  }

  public static void main(String[] args){
    new MultiFrameTest();
  }
}
--
DB
```

📧 Q: How do I use the DataInputStream and DataOutputStream to transfer a file from the server to the client using sockets in JAVA?

Answer: This will run on a single computer.

```java
// Client.java

import java.net.*;
import java.io.*;

public class Client {
  public static void main(String[] args)
  throws IOException {
    InetAddress addr =
      InetAddress.getByName(null);
    System.out.println("addr = " + addr);
    Socket socket =
      new Socket("127.0.0.1", 8080);
     try {
       System.out.println("socket = " + socket);
       BufferedReader in =
```

```
            new BufferedReader(
             new InputStreamReader(
              socket.getInputStream())));
        PrintWriter out =
           new PrintWriter(
             new BufferedWriter(
               new OutputStreamWriter(
                 socket.getOutputStream())),true);
      } finally {
         System.out.println("closing...");
          socket.close();
      }
    }
}

// Server.java

import java.io.*;
import java.net.*;

public class Server{

    public static final int PORT = 8080;
    public static void main(String[] args)
    throws IOException {
       ServerSocket s = new ServerSocket(PORT);
       System.out.println("Started: " + s);
       try {
          Socket socket = s.accept();
           try {
             System.out.println(
                "Connection accepted: "+ socket);
                BufferedReader in =
                  new BufferedReader(
                    new InputStreamReader(
                      socket.getInputStream())));
            PrintWriter out =
              new PrintWriter(
                new BufferedWriter(
                  new OutputStreamWriter(
                    socket.getOutputStream())),true);
             } finally {
               System.out.println("closing...");
              socket.close();
              }
           } finally {
         s.close();
       }
    }
 }
```

by Bob Randall <<u>brandall@pon.net</u>>

📖 Q: I have seen several Java pages on the net that give examples of launch the preferred application under Windows for a given URL.

These examples, including one at JGuru, look something like this:

Runtime.getRuntime().exec("start http://www.somecompany.com/foo.html");

This does not work for me. I get an IOException with "error 2", whatever that is. If I try to run other apps in my path, it works fine:

Runtime.getRuntime().exec("winhlp32.exe");

Is there some way to get "start" to work, or are these examples full of hot air? If "start" won't work, is there another way?

Answer: READ before this articles!!!
To answer my own question, there is an excellent JavaWorld tip that covers this issue:

When Runtime.exec() won't
http://www.javaworld.com/javaworld/jw-12-2000/jw-1229-traps.html

Basically, you have to perform an OS check and then execute the command interpreter since "start" is a command, not an executable. Under NT/2000, you would do:

Runtime.getRuntime().exec("cmd.exe /C start http://www.somecompany.com/foo.html");

The downside of this is that the command window appears briefly before the web browser opens.
--
Ken Carpenter
Also! http://www.javaworld.com/javaworld/javatips/jw-javatip66.html

📖 Q: Hi, it would be appreciated if some one could tell me where I can find a Java sample code for a draggable image, i.e. using mouse left button to drag a bitmap from one location on a dialog box and drop it on another location of the same dialog box.

Answer:

Example:
import javax.swing.*;
import java.awt.event.*;
import java.awt.Point;
import java.net.*;
import java.awt.*;

// test of dragging various components also mouse event tests


public class DragTest extends JFrame{

```
        int xPos;
        int yPos;
        int lastXPos;
        int lastYPos;
        boolean first = true;

        JLabel b;
        URL url;
        Image myImage;

        public DragTest(){
          super("Drag Test");
          JPanel p = (JPanel)getContentPane();
          p.setLayout(null);
          try{
             //myImage = Toolkit.getDefaultToolkit().getImage(new URL
               ("http://www.javasoft.com//images//logos//javalogo52x88.gif"));
             url = new URL
                ("http://www.javasoft.com//images//logos//javalogo52x88.gif");
          }
          catch (Exception e){System.out.println(e);}

          ImageIcon icon = new ImageIcon(url, "Here");
          System.out.println("got image "+icon.getImageLoadStatus());
          b = new JLabel(icon);
          b.addMouseListener(new MouseAdapter(){
            public void mouseClicked(MouseEvent me){
              int times = me.getClickCount();
              if ( times <= 1){
                System.out.println("Single "+me.getX()
                   +" "+me.getY());
                Point here = b.getLocation();
                System.out.println("Button is
                   at "+here.x+" "+here.y);
              }
              if (times == 2){
                System.out.println("Double");
                me.consume();
              }
              //System.out.println("Clicked = "+times);
            }
            public void mousePressed(MouseEvent me){
              System.out.println("Pressed");
              lastXPos = me.getX();
              lastYPos = me.getY();
            }
          });
          b.setEnabled(true);
          b.setSize(b.getPreferredSize());
          //b.setLocation(0,0);
          p.add(b);
          b.addMouseMotionListener(new MouseMotionAdapter(){
```

```
public void mouseDragged(MouseEvent me){
  // b.setEnabled(false);
  Point currentPos = b.getLocation();
  int curX = currentPos.x;
  int curY = currentPos.y;

  xPos = me.getX();
  yPos = me.getY()-24;
  if (first){
    lastXPos = xPos;
    lastYPos = yPos;
    first = false;
  }
  System.out.println("y = "+yPos+"lastY
        = "+lastYPos+" "+first);
  int deltaX = xPos - lastXPos;
  int deltaY = yPos - lastYPos;
  try{
    Thread.sleep(30);
      if ((Math.abs(deltaX) < 3)&&(Math.abs(deltaY) < 3)){
        System.out.println("Made it");
        b.setLocation(curX+deltaX,curY+deltaY);
        if (Math.abs(deltaX)< 3){
            lastXPos = xPos;
        }
      if (Math.abs(deltaY )< 3){
        lastYPos = yPos;
      }
    }
  }
  catch (Exception e){}
  // b.setEnabled(true);
  }
});

addMouseListener( new MouseAdapter(){
  public void mouseClicked(MouseEvent me){
    System.out.println("Frame "+me.getX()+" "+me.getY());
  }
});

addWindowListener(new WindowAdapter(){
  public void windowClosing( WindowEvent we){
    dispose();
    System.exit(0);
  }
});
setSize(200,200);
setVisible(true);
}
```

```
    public static void main(String[] args){
      new DragTest();
    }
  }
--
DB
```

....add to PDF page only!!!
A: PDF java
http://etymon.com/pj/
IBM's approach:

http://www-106.ibm.com/developerworks/education/transforming-xml/xmltopdf/index.html

Q: Do any of you know a free web host supporting JSP pages?

Answer: check out http://www.mycgiserver.com
Its offering free application deployment for java(includes jsp1.1), xml, wml
Try http://www.webappcabaret.com , jsp's , servlets and EJB1.1 hosting. And its free
:)


Q: When settext method is used in JButton, lots of button is wasted.
How can I set the margin or something like that so that I can utilize all the space of
the button to write something?

Suppose I want to create a button with "Ok"as text, by default a button with some
size is created with lot of space around "Ok" text being unused.
If I reduce the button size little, the text disappears showing "..."

How can I get around with problem? Any ideas?
Answer: Subclass JButton and override the getInsets method. getInsets is what the
layout manager calls when it wants to render the button.
Here is a sample:

```
import java.awt.*;
import javax.swing.*;

public class TestApplet extends JApplet {
  MButton b;

  public void init() {
    getContentPane().setLayout(new FlowLayout());
    b=new MButton("OK");
    getContentPane().add(b);
  }
}

class MButton extends JButton {
  public MButton(String t) {
    super(t);
```

```
    }

    public Insets getInsets() {
        return new Insets(0,0,0,0);
    }
}
```

📕 Q: I would like to know how I can display a gif image on a normal AWT button.
I need a button which displays an Image for my project.
I know that swings button can do this but I am forced to work with AWT. Can you
offer any suggestions?

Answer:

```java
import java.awt.*;
import java.awt.event.*;

//class to make an animated button using images.
//Written by Mark Bernard
class ImageButton extends Button implements MouseListener {
    Image i[];
    int select=1;
    int w=0;
    int h=0;
    int iw,ih;

    //The constructor requires 4 images as described below.
    // 1. Greyed out image of the button
    // 2. Normal/unselected image
    // 3. Hover image(if mouse is hovering over the button
    // 4. Pressed image
    //Please note that the image will always take up the entire
    //display of the button. If layout managers are used
    //the image will be stretched to fit the area layed out.
    public ImageButton(Image im[]) {
        super(" ");
        i=new Image[4];
        i=im;
        iw=i[0].getWidth(this);
        ih=i[0].getHeight(this);
        setSize(iw,ih);
        addMouseListener(this);
    }

    public Dimension getPreferredSize() {
        return new Dimension(iw,ih);
    }

    public Dimension getMinimumSize() {
        return new Dimension(iw,ih);
    }

    public void setBounds(int x,int y,int width, int height) {
```

```
        w=width;
        h=height;
        super.setBounds(x,y,width,height);
      }

      public void setSize(int width,int height) {
        w=width;
        h=height;
        super.setSize(width,height);
      }

      public void setEnabled(boolean e) {
        if(e) {
          select=1;
        }
        else {
          select=0;
        }
        repaint();
        super.setEnabled(e);
      }

      public void paint(Graphics g) {
        g.drawImage(i[select],0,0,w,h,this);
      }

      public void mouseClicked(MouseEvent e) {}
      public void mouseEntered(MouseEvent e) {
        if(select!=0) {
          select=2;
          repaint();
        }
      }
      public void mouseExited(MouseEvent e) {
        if(select!=0) {
          select=1;
          repaint();
        }
      }
      public void mousePressed(MouseEvent e) {
        if(select!=0) {
          select=3;
          repaint();
        }
      }
      public void mouseReleased(MouseEvent e) {
        if(select!=0) {
          select=2;
          repaint();
        }
      }
    }
  }
```

📖 Q: I have a method with the following signature:

public Element process(java.io.Reader reader)

I usually call this with a java.io.FileReader, to process files from local disk.

Like this:

Element root = null;
FileReader fr = new FileReader("config.xml");
root = process(fr);

Now, I need to process a file residing on a Http-server, and I have a reference to this file in a java.net.URL.

Element root = null;
URL configURL = new URL(http://servername/Path/config.xml);

????

root = process(??);

How do I convert/call my URL to a Reader object so I can call process(Reader reader)???

Answer: The following example might be helpful to you:

```java
import java.net.*;
import java.io.*;

class Dag {
  static URL url = null;
  static int[] letterCount = new int[256];

  public static void main(String[] args) {
    try {
      url = new URL("http://www.orion.no");
    } catch (MalformedURLException e) { }
    try {
      letterCount = process(new
      InputStreamReader(url.openConnection().getInputStream()));
    } catch (IOException e) { }

    for (int i=0; i<256; i++) {
      if (letterCount[i]>0) {
        System.out.print((char)i+" "+letterCount[i]+",\t");
      }
    }
  }

  public static int[] process(Reader reader) {
```

```
      int c = 0;
      int[] counters = new int[256];

      while (true) {
        try {
          c = reader.read(); System.out.print( (char) c );
        } catch (IOException e) { }
        if (c<0) { break;}
        counters[c] ++;
      } return counters;
    }
}
```

The program reads from the webpage at http://www.orion.no and outputs the frequencies of all (present) letters.
--
/ Lars-Ake

Q: I would like to create a internal timer with 10ms resolution. I have created the following class from java.util.TimerTask 'TimerTask' class:

```
import java.util.TimerTask;

class MyTimerTask extends TimerTask {
  private long tvalue=0;
  private long delay;
  private long old_current_time;

  public MyTimerTask (long d) {
    delay=d;
    old_current_time=System.currentTimeMillis();;
  }

  // Run timer
  public void run() {
    long current_time = System.currentTimeMillis();

    tvalue += delay;

    // Every 1s, print local timer value and relative system time value
    if (tvalue % 1000 == 0) {
      System.out.println (tvalue+" "+(current_time-old_current_time));
      old_current_time = current_time;
    }
  }

  public long get_tvalue () {
    return tvalue;
  }
}
```

I use the followin main program to test my class:

```
import java.util.Timer;
import java.util.TimerTask;

class MyTimerTest {
  public static void main(String args[]) {
    long d;

    // Get period, default is 1000 ms
    if (args.length != 0)
      d = Long.parseLong (args[0]);
    else
      d = 1000;

    // Init timer
    MyTimerTask rt = new MyTimerTask (d);
    Timer t = new Timer ();

    // Start it
    t.schedule (rt, 0, d);
  }
}
```

If the period is not too short, >= 100ms, the result is not so bad:

```
[pierre@ca-ol-bordeaux-6-161 test]$ java MyTimerTest 100
1000 1454
2000 1100
3000 1100
4000 1101
5000 1099
...
```

But if I use a short 10ms period it looks like timer works very slowy: difference between n and n+1 print is 2000ms instead of 1000ms in system time:

```
[pierre@ca-ol-bordeaux-6-161 test]$ java MyTimerTest 10
1000 1994
2000 2000
3000 2000
4000 2000
5000 2000
6000 2000
...
```

📄 Q: Could you give me simplest example how to do print in Java? I will work out it myself :-)

Answer: Please compile and run it! It will draw empty rectangle (you see I save your inks!)

```java
import java.awt.*;

public class print {

  public static void main(String args[]){
    Frame frm = new Frame("JavaFAQ_test");
    frm.pack();
    PrintJob printJob =
        frm.getToolkit().getPrintJob(frm, "print", null);

    if (printJob != null) {
      Graphics grphcs = printJob.getGraphics();
      grphcs.drawRect(50, 50, 150, 100);
      grphcs.dispose();
      printJob.end();
    }
    System.exit(0);
  }
}
```

--
AP. (J.A.)

Q: I have small advice how to avoid the usage Date for measurement the time difference between two events.
The main idea is that the Garbage Collector collecting only objects that were created by using new().

So if you need to measure the time difference between two events use this:
long eventOne = System.currentTimeMills();
long diff = eventOne - System.currentTimeMills();

Give the rest to you Garbage Collector!
--
Andrey S.
P.S. This advice was sent directly to us, to info@javafaq.nu
Have you such? Please send!

| | | | | |
|---|---|---|---|---|
| ☐ IT Professional | ☐ General Computer | ☐ Design and Graphics | ☐ Linux | ☐ Freeware/Shareware |
| ☐ Games | Intranet | Computer Hardware | Web Design | ☐ Computer Software |
| ☐ UNIX | ☐ Web Software | ☐ Windows Software | ☐ Personal Finance | ☐ Programming Software |
| ☐ Servers | ☐ C/C++ | ☐ Powerbuilder | ☐ Perl | ☐ XML |
| ☐ Java | ☐ SQL | ☐ Design/Graphics | ☐ Utilities | ☐ Handhelds |

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Databases & beans

Q: Anybody does know a freeware JDBC driver for a dsn-less connection to MS SQL-Server? Would even consider a "cheapware" version.

Answer: Go to http://industry.java.sun.com/products/jdbc/drivers and search for Microsoft SQL Server. Any Type4 (i.e. pure Java) driver should work without a DSN.

The only free one I'm aware of is at http://www.freetds.org - but it is rather limited in what it can do. You'd need to try it out to see whether it fits your requirements.
--
Stefan
P.S. DSN - Data Source Name

Q: I just want to know which programs and virtual machines you have to have to make and run enterprise java beans...

Answer: To compile and run Enterprise JavaBeans, you need a couple of things.

First, you need the J2EE SDK. This kit includes APIs full of packages which are considered extensions to the standard Java language APIs, as well as other tools, which come with the J2SE SDK, which you should already have. Install the SDK and make sure its jar file is in your development environment's classpath.

Second, you need a container, which in this case you can also refer to as an application server, though technically a container is just one part of the server. The container acts as a liaison between the client object and the Enterprise JavaBean. When you talk to an Enterprise JavaBean, you actually talk to a proxy (a substitute), and the proxy, which knows how to do networking stuff, talks to the container, which in turn talks to the actual implementation object which is what you think of when you think of an Enterprise JavaBean.

The J2EE SDK, fortunately, comes with a server/container, as well as a GUI-based tool which allows you to deploy your Enterprise JavaBeans in the server. See java.sun.com/j2ee.

Third, you need a lot of patience. The learning curve is rather steep unless you have a lot of experience doing network
programming. Enterprise JavaBeans are designed to abstract out networking and storage logic, which ends up being very helpful, but is confusing at first, because so much happens behind the scenes that is not explicitly controlled by your code. For example, when you deal with a single Enterprise JavaBean, at least five different objects are actually being instantiated!

But it's great once you get past the initial learning stage, which can last a while. There are lots of good books on EJB, but I found Ed Roman's "Mastering Enterprise JavaBeans" to be a great primer.
--
Erik

📖 Q: I'm having a hard time figuring out what are the differences between Enterprise Java Beans and Java Beans.

Is there a definitive difference between the two?

Answer: Definitely. JavaBeans are really nothing more than classes that have no-args constructors and follow certain naming conventions (and/or provide a BeanInfo class) to identify properties, methods, and events. JavaBeans were designed for plugging into GUI design tools. Technically speaking, lots of things are JavaBeans... though whether they are intended to be used that way is another matter altogether.

Enterprise JavaBeans are not really used as JavaBeans at all. They run on a different server, in a special EJB container that provides a bunch of restrictions on their class hierarchy, their fields and object relationships (particularly if CMP is used), the Java language features that can be used if you write them, etc.
They are, of course, far from a GUI design thing.
<editorial>They are also, IMHO, far from a good idea and I'd avoid them if at all possible.</editorial>

--> to be continued tomorrow (end of Part 1)

📖 Part 2.
> Please explain why to avoid them in more depth...

Basically, the problem is that Sun has started reflection "activity". I use that term to describe the unnecessary use of reflection when a good solution with static typing would be preferable.

Examples: Standard RMI has always used interfaces to act as common types between client and server. EJBs abandon this, and just say "well, define all the same methods" with no static checking that you've done so. You then have to write a bunch of "ejbCreate" methods, with exactly the same signature as "create" methods, but needlessly renamed. (Well, you kinda would have to name them in order to stuff them into a class where they don't belong, which is exactly what EJB does.) Then EJBs examine my object's fields (which are supposed to be private), and sometimes the whole thing stops working because I have a field of a type that the container doesn't like.

The result is something that shares basic syntax with Java, but which is really a different beast altogether, with no well-organized bit of documentation on use (because everything is dynamic through reflection instead of working with well-defined methods in well-defined classes for which documentation can be written), and strange and unreasonable constraints on coding.

That said, EJB containers are the only environments that provide anything like CMP, for example. I'm working on a better-designed replacement for EJBs' CMP that doesn't rely quite so heavily on reflection, but I'm unaware of anything widely available.
--
Chris Smith

--> to be continued tomorrow (end of Part 2)

📑 Part 3.
It was a poor naming decision on Sun's part. There is no similarity between Java Beans and Enterprise Java Beans except for their name.
All they managed to do was cause confusion just because they liked the cutesy name.

The confusion is reduced when you realize that "Bean" in Java is the cutesy name for that software engineering term, aka buzz word "component", that is something intended to be re-usable as is without change, the software equivalent of the hardware engineers chip.

One note: a JavaBean is intended for use in some type of building tool, but a bean or a set of beans may not have anything to do with the GUI of the resulting application. Most often the intent is that the bean is _configurable_ using some GUI. The typical examples of non-graphical beans are Database access JavaBeans which result in nothing in a GUI, but may communicate with other Beans which are part of the application GUI.

I was annoyed a few months back when I attended a Sun training, which included coffee bean looking icons, plus various other curious bits of graphics which to me just added clutter. I still don't know what some of the silly figures where supposed to be :-). Some simple UML (some boxes and arrows) would have been much clearer.
--
Comments by Tom Almy, Paul Hill

📑 Q: I want to use MS Access databases in my java app. Where do I start and where can I get the drivers, I am fluent in SQL with Active Server Page experience, so that isn't a program, its just setting up the DB connection in Java.

Answer:
1. From the ODBC control panel, define a system data source name for the specific MS Access database you want to use. Make sure it's a system DSN, not a user DSN. Assume you name it "foo".

2. In your Java application, load the JDBC driver:
```
try {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

```
}
catch (ClassNotFoundException e) {
    // Shouldn't happen - it comes with the JRE
}
```

3. Create the DB connection using the name of your system DSN from step 1:
```
Connection con = DriverManager.getConnection("jdbc:odbc:foo");
```

That's it. From there on, the connection behaves like a JDBC connection to any other kind of database. See the tutorial for details.
Remember to close the connection - it's best to enclose step 3 in a try/finally block:

```
Connection con = null;
try {
    con = DriverManager.getConnection("jdbc:odbc:foo");
    // Do your stuff
}
finally {
    if (con != null)
    con.close();
}
```
--
Phil Hanna
Author of JSP: The Complete Reference
http://www.philhanna.com

Q: I don't think my brain just isn't getting the whole picture...

If I write a Java app that is going to be talking to Oracle, is the only method of getting the Oracle server to do anything via SQL calls?

Assuming that my Java App is on a laptop, and talking (I guess TCP/IP) to the Oracle Server, what processing/data access options do I have?
The Oracle site seems to talk about a Java API to Oracle, but inside how would my laptop app be getting the data? Is this API just a wrapper around SQL calls? Is there an "Included into my Laptop App" something that will allow me to create objects based upon rows of data stored at the server, or do I have to get the data, and then construct the Java objects myself?

I guess my simple question is,
"Is SQL the only data access method to Oracle that I've got?"

If there are any successes I would like to hear them (not from Oracle though, I have a hard time buying people blowing their own horns).

Answer: If you want to know how to get >data< interaction between an Oracle database (or any other 'industrial' database) and your Java app running on a laptop, the primary & recommended way is to use a SQL API. The possibilities include JDBC and SQLJ. Both are industry standard APIs and are explained at
http://technet.oracle.com/tech/java/sqlj_jdbc/

Other methods for interacting for interacting with the JVM inside the database include XML, JSP, CORBA, EJB, etc. But most fall back on JDBC and/or SQLJ for actual interaction with database objects. Recommended page to start looking at this is

http://technet.oracle.com/tech/content.html

Oracle does provide a few things to help you out including (but not limited to):
JDeveloper is their Java IDE and it has a number of enhancements & wizards to try to make life a lot simpler for various types of operations;
and BC4J (Business Components for Java) provides a framework for creating and using 'business components' that require database interactions.

Oracle database built-in capabilities also minimize or eliminate special custom Java code; intermedia is included in the database and handles virtually all multimedia interactions as well as text indexing and text searches a lot easier than custom Java code;
spatial option simplifies interaction with polygon and spatial coordinate manipulation, management and searching;
iFS does a fair bit of standard document management while making the database look like a NFS disk (or network drive in Microsoft environment).
However, most of these use some form of SQL API (again JDBC or SQLJ) as the Java interface.

And since the Oracle9i Application Server includes a report and ad-hoc query capability, that can eliminate or minimize Java for information distribution.
--
Hans Forbrich

📋 Q: I used JDBC driver to connect Microsoft SQL server Database. It is no problem besides display Chinese word...

Answer: You can try

```
public String fromDB(String in) {
    try{
        return new String(in.getBytes ("Iso8859-1"),"Big5");
    }
    catch(Exception e){
        return "";
    }
}
```
This is the problem at the encoding of string.

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

# Distributed systems

## Q: Has anyone ever tried anything like this or am I asking for trouble trying to write a program like this?....

I plan to use JBuilder to create a Java GUI that will use Perl to invoke system calls.
The GUI will be run in Windows(NT) while the system calls will be invoked in Unix.

Answer: Sure, why not? Seems to me it should be quite doable. Use Java code to build the GUI and
cross the network (for instance using RMI), then invoke the Perl interpreter as an external process, or possibly use JPerl (see
http://www.perl.com/CPAN-local/authors/id/S/SB/SBALA/ ) from there. Or use a different distributed objects architecture to connect Java and Perl objects over the network...

## About serialization...

If I have a class that implements the Serializable interface, but it has member variables which reference objects that do not implement the Serializable interface, it appears that I can't serialize an instance of the class. I keep getting:

java.io.NotSerializableException

for one of the objects referenced by a member variable.

Am I correct, or am I just missing something. Also, if anyone knows a work-around to serialize non-serializable objects, I'd like to hear about it. Unfortunately, I have no control over the classes I'm trying to serialize, so I tried putting a serializable wrapper around them , but that didn't work.

Answer: Do you really need to serialize those members of your class which aren't serializable? In other words, make them private:

class Foo implements Serializable {
    private Bar bar;

}

Do you *need* to maintain the state of the 'bar' variable when serializing/deserializing Foo? If not, simply declare 'bar' as 'transient' and it will be ingored during serialization.

### 📕 RMI versus Socket communication

I wish to get Java talking to C++ across a network.
Does anyone have any thoughts in terms of performance, ease of development etc. in :
Wrapping the C++ side with JNI and using RMI for the communications.
versus
Writing sockets code and communicating via http?

Answer: It depends of what kind of application you're writing but I think about the following :

- with RMI you can have remote REFERENCE instead of having to transfer all the object through the network. The object has just to implement Remote. So it spare bandwith and is good for performance. This is impossible to do if you do through a socket connection, you've to send the all object.
- You've not to take in charge the serialization (which could be not so easy depending of your object structure), neither the connections, etc... All of that is taken in charge by RMI.
- the performance are GOOD (even a bit more than that)
three good points to use RMI, isn't it?

The difficulty added by RMI is the configuration of both client and server (distribution of stubs, rmiregistry, what's happen if firewall). Depending of the environment all of that can be either easy or complicate.
But once that all of that is in place you can extend your application
easily, so it's much more flexible and scalable.

If your needs are small perhaps that you could do your own connection system (but for me it's less scalable and more bandwith consuming and so less performant).
--
François Malgrève

Answer 2: I have done both. If your communication scenarios are diverse and could keep changing, using a remote technology like RMI can help. If the operations are few and/or not likely to change you can save the JNI complexity. Not that it is really hard it just can be fun keeping the JNI code in sinc  with the C++ code.
--
Bret Hansen

📕 Q: I need to communicate some data (string) from a Java Applet to an other ASP page in the same frameset. I would like to avoid a server roundtrip and do it all with JavaScript if possible.

Therefore I would like to call some javascript from a Java Applet. It looks like it is not possible without a netscape package. Is that true? Is there a simple implementation of the same functionality (source code) which I could incorporate in my applet?

Answer: Those Netscape packages are part of the current VM of both Microsoft IE 4+ and Netscape 4+. So, by adding the MAYSCRIPT tag to your Applet declaration, in the Java code you can obtain a handle to the document and call functions in it.
by Tom Hall

Q: I'm researching methods by which one JVM can interact with another JVM, which is running on the same machine.

I know that there are various network models, which can be applied if a JVM needs to talk to another one across a network, but in addition to these (which could I guess be applied to JVMs on the same machine) I wondered if you knew of a system of JVM communication that requires less system resources, where the JVMs are both running on the same system.

Answer: CORBA, RMI, HTTP, sockets....
But if you have no TCP/IP stack on your platform, so for Windows it could be clipboard...
--
by dmitry

Q: I have a question about sending a reference to the object via the socket...

I have a question about sending a reference to the object via the socket. Two threads are communicating via sockets running on the same machine. I don't need to send the whole object, but I need to send just
a reference.
Does anyone knows how to do that?
Answer: Reference to an Object? A reference is only valid within the same memory space! If you want to be able to invoke methods on an object remotely, then you will need to use a remote technology like RMI, CORBA, or some such.
--
by Bret Hansen

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## File Systems - I

How does Java read the text files? Can Java read the files that are in other formats? Is the read file method in Java only recognizes the file in .txt or other text format?

Answer: Java can read any text file ( using a java.io.FileReader for example ), the attribute at the end is an indictor and thus is not relevant as long as the actual code read is in the correct format. I It can read files that are in other formats bytes etc and if you have a wierd format you could extend the IO mechanism with some work to work with that.

Q: What's the preferred way to copy files in Java?

Renaming is easy, since java.io.File provides a method for that.
But I didn't find a method for copying.

Answer:

```
public static void copyFile(File srcFile, File dstFile)
     throws IOException {
  BufferedInputStream in =
     new BufferedInputStream(new FileInputStream(srcFile));
  BufferedOutputStream out =
     new BufferedOutputStream(new FileOutputStream(dstFile));
  byte buffer[] = new byte[BUFFER_SIZE];
  int count;

  while ((count = in.read(buffer)) != -1) {
    out.write(buffer, 0, count);
  }

  in.close();
  out.close();
}
```

Note that this doesn't close either of the streams if an exception is thrown - which is precisely the kind of thing that finally blocks are for. (Be careful to make sure you close both streams even if one of the calls to close throws an exception though, and not to try to call close() on null if the streams haven't been set up...).
It's not a full-fledged solution. My proposal is only a draft.
--
Peter

Q: Is it possible to choose a directory path instead of a file path in a  swing FileChooser dialog box?

Answer: Assume you have instance chooser of type JFileChooser, invoke...
chooser.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);

Q: Is there anyway to find out the creation date of an existing file?

Answer: The only thing you can get with pure Java is the last-modified date, which may or may not be the same as creation date.

Probably the reason the API is limited in this fashion is because not all file systems even store the creation date.

Windows does store it, but you'll need JNI (or possibly some horribly ugly hack with Runtime.exec() and the DOS "dir" command) to get at it.

Q: Does anyone know the way to retrieve the path from where has been loaded the main class of an application, not the path from where has been launched the virtual machine?

Answer: Please use something like this:
path = System.getProperty("user.dir");

Q: How to erase the content of (text) file without create/open file again (only do "new FileOutputStream(...)" once)?

Answer: Try java.io.RandomAccessFile.setLength(0) if you're using JDK 1.2 or higher.

If you don't have to keep the file, it may be easier to use java.io.File.delete().

Q: When creating a new file, is it possible to control whether or not an existing file with the same name is/is not overwritten? I haven't been able to answer this by looking at the java.io api

Answer:  There is a method in File to atomically create a new file which will fail if the file exists... You can use this to try creating the file, and if it exists already don't ever open the FileOutputStream to write contents.

Q: Do exist Compound Files in Java?

Microsoft has made an API where you can have a whole "virtual" filesystem inside a single file on the real filesystem. I think they call it Compound Files - I have also heard about the concept under the name of Structured Storage.

Do any of you know if something like that exists in a Java-library? I have to work for all platforms! (It can of cause be implemented with the use of Compound Files in its Windows implementation)
Answer: jar files are essentially the same.

Have a look at java.util.jar.JarFile
--
by Bret Hansen

Q: Is it possible to redirect the System.out.println to a file?

Answer: Connect a PrintStream to the file, and then call System.setOut (PrintStream out) that reassigns the "standard" output stream.

Q: Could some kind person please tell me how to save the object as a file so as the same program can load it again?

Answer: try this program. It saves obect into file:

```java
import java.io.File;
import java.io.FileOutputStream;
import java.io.ObjectOutputStream;
import java.io.IOException;

public class Save{
  public void saveMyObject(String filename, Object obj) {
    File myFile = new File(filename);
    try {
       FileOutputStream fileOutStr = new FileOutputStream(myFile);
       ObjectOutputStream outStr = new ObjectOutputStream(fileOutStr);
       outStr.writeObject(obj);
       outStr.close();
    }catch (IOException e){
       System.out.println("?!!!!!!");
    }
  }
   public static void main (String args[]) {
    Save s = new Save();
    Object myObject = new Object();
    String test = "test";
    myObject = (Object)test;
    s.saveMyObject("myfile", myObject);
  }
}
```

If you open myfile you will see that this object includes our string "test"
In the same manner you can read this object from file...

Q: Can anyone write me a short method that lets me know what files are in a particular directory?

For example, I want to know that directory, d:/temp/aaa, has files a.txt, b.java, b.class.
Also related to this, how do I find out what folders I have?

Thanks in advance.
Answer:  use our program as a base and add checking for the files and directories you need to find!

here it is:

```java
import java.io.File;
public class Save{
   public void showDirectoryList() {
      File dir = new File("d:/temp/aaa");
      File[] list = dir.listFiles();
      for (int i=0; i<list.length; i++) {
         if (list[i].isFile()) {
            System.out.println("File "+list[i].getName());
         } else if (list[i].isDirectory()) {
            System.out.println("Directory "+list[i].getName());
         }
      }
   }
   public static void main (String args[]) {
      Save s = new Save();
      s.showDirectoryList();
   }
}
```

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

# File Systems - II

Q: How do I delete a file in Java? I have programmed a Java application that needs to delete a file, but I couldn't find anything on the topic.

Answer:  in java.io.File there is delete
public boolean delete()

Deletes the file or directory denoted by this abstract pathname. If this pathname denotes a directory, then the directory must be empty in order to be deleted. Create a new File object representing the file, and then use the delete () method. If you use the Microsoft JVM, make sure the file is not shared before you delete it, otherwise that will not work (the delete () method returns "false").

Q: I need to read C:\test.doc for example and store it somewhere for my future use. I don't know anything about security. Part 1

Answer:  If you want to read a file on the local file system in an applet, you are going to digitally sign the applet, and the user of the applet is going to indicate trust in the signature. How you go about this depends on a number of questions, like:
In which browser(s) will the users be running the applet?
Will a Java plug-in be installed in the browser?
Can the user install a plug-in?

Here are a few tips on some of the sticky points of signing applets...
--------------------------------------------------------------------

If you are using the jar signing tools that come with the JDK, you'll find that they will only work with the Java plug-in. Netscape and Microsoft IE don't salute the Sun way of signing applets.

If you can rely on your users to install the Java plug in, you can sign applets using the JDK 1.1.X javakey command. You will also have to have them import your certificate into their 'identitydb.obj' file in their home directory, and mark it as trusted. If importing the certificate is too much to ask, and it usually is, you could create an

'identitydb.obj' containing the certificate, already marked as trusted, and have them download this to their home directory. Be careful not to distribute the 'identitydb.obj' that you used to sign the jar, since that would allow anyone to sign code as you!

JDK1.2.X works in a similar way, but the 'javakey' command has been replaced by 'jarsigner' and 'keytool'. In addition, 1.2 adds granularity to the security model, so that you can assign privileges outside of the sandbox via a policy file. For example you could allow an applet to read files, but not write them. JDK 1.1.X security is kind of an all or nothing proposition.

If you can't count on the Plugin being there, but you know your users will be using Netscape, you can sign use the Netscape signtool. Netscape's signtool is a little different than Sun's. With Netscape's, you put all of your class files into a directory, like signdir, and then run something like this:
./signtool -k MyCert -Z MyApplet.jar signdir
The 'MyApplet.jar' file will be created, and, of course, you would already have to have MyCert in your security database. You can generate a test certificate through signtool, but it's only good for a couple of months.
--
Joe Morrogh - Excite@Home - The Leader in Broadband http://home.com/faster

Q: I need to read C:\test.doc for example and store it somewhere for my future use. I don't know anything about security. Part 2, Part 1 was published yesterday

Answer 2: I think they want you to go out and buy a cert from Verisign or Thawte, etc. You can also generate your own signing certificate through openSSL, or some other package, but, if you do, you'll need to have them import it into their database as a trusted signer. This can be done with a simple Perl script which sends the certificate with a header of:
"Content-Type: application/x-x509-ca-cert"
You must also add some code to your applet to use the Netscape way of signing. You can download the file 'capsapi_classes.zip' from Netscape. This file contains the Netscape Capabilities API. Basically, all you need to do is import the package and enable privileges as you need them. It is important to enable privileges in the same method (ie. same frame stack)
in which you are going to use them. You cannot just enable all the privileges you need in the init method of your applet. The code looks something like this:

```
import netscape.security.*;
public void MyMethod() {

  try {
    PrivilegeManager.enablePrivilege("UniversalFileRead");
    FileInputStream fis = new FileInputStream("myfile");
  } catch (Exception e) {
    System.out.println(e);
  }
} // end MyMethod
```

You'll need to add 'capsapi_classes.zip' to the archive attribute of your applet tag. I also believe it is possible to grant privileges to a codebase in Netscape without signing. This is done by editing the 'prefs.js' file on the client. Obviously, this is not

possible on the internet, but, for an intranet environment, this could be a solution. Anyway, when an applet tries to enable a privilege, the Netscape browser will pop up a window asking the user if they wish to grant the privilege. This decision can be persistent.
The user can manage the privileges they have granted by clicking on the padlock icon in the bottom left corner of their browser window, and selecting the "Java/JavaScript" section.

If your users may be using Microsoft IE as well as Netscape, you'll have to learn the Microsoft security API. I haven't done this yet, but I believe it is possible to code an applet so that it will work in either browser.
Another alternative is to code different versions of the applet, and build the page containing the applet dynamically, based on the "User-Agent". I have found that it is possible in IE to grant universal privileges to an applet, even if it is not signed, but I sure wouldn't recommend doing that.
--
Joe Morrogh - Excite@Home - The Leader in Broadband http://home.com/faster

Q: Is there any method like writeLine()?

Answer:  For writing text to files, you can use a FileWriter; for binary data, use a FileOutputStream. Both are in the java.io package. Have a look in the docs, all you need is there. Oh, you might want to wrap your writer/outputstream in a buffered one for more efficiency. Something like this:

```
FileWriter fw = new FileWriter("D:\temp\test.txt");
BufferedWriter bw = new BufferedWriter(fw);
bw.write("Some text on the first line");
bw.newLine(); // have the Writer insert the newline
bw.write("Some more text, followed by an explicit newline\n");
bw.close();
```

Note, that while \n will probably work, newLine() inserts the platform's own line separator (ie, \r on Mac, \r\n on Windows, \n on *nix).
--
Michiel


Q: I want to copy a set of gif files from one directory to another. If I use any of the File Reader/Write classes of java.io.package, gif files are being copied but are distorted.

Most probably because Java reads in character mode. So, how can I copy gifs ?

Answer: use FileInputStream FileOutputStream instead.

Q: Is there a newsgroup on how to use Java for CDs? I mean not CD that install someting but CDs that you browse.

Answer: Strictly speaking, there is no difference between "CDs that install something" and "CDs that you browse". All CDs contain the same file structure. It's up to the operating system to decide what it does with those files. Windows includes functionality to start a special file on the CD automatically, typically to install something.
You read them just as you would any other disk files. To create them, usually you

provide a list of files to a creator program.
--
Joona Palaste

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

☐ IT Professional  ☐ General Computer  ☐ Design and Graphics  ☐ Linux  ☐ Freeware/Shareware

☐ Games  Intranet  Computer Hardware  Web Design  ☐ Computer Software

☐ UNIX  ☐ Web Software  ☐ Windows Software  ☐ Personal Finance  ☐ Programming Software

☐ Servers  ☐ C/C++  ☐ Powerbuilder  ☐ Perl  ☐ XML

☐ Java  ☐ SQL  ☐ Design/Graphics  ☐ Utilities  ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

# 📕 Graphics, AWT, Swing - I

📘 Q: How can I minimise "Flicker" in animation? Solution 1:

Answer: Solution 1:

Override update() : Flickering in animation occurs because default update() method clears the screen of any existing contents and then calls paint(). To reduce flickering, therefore, override update(). Here is how just add the following code to your applet:

```
public void update(Graphics g) {
    paint(g);
}
```

What the update is now doing is just calling paint() and not clearing it, a further refinement to the above will be to override update() method and painting only the region where the changes are taking place. Here is how:

```
public void update(Graphics g) {
    g.clipRect(x, y, w, h);
    paint(g);
}
```

Solution 2  will be described tomorrow

📘 Q: How can I minimise "Flicker" in animation? Solution 2:

Solution 1 was described in our tip yesterday
Solution 2:

Use double-buffering : double buffering is the process of doing all your drawing to an offscreen and then displaying the entire screen at once. It is called double buffering because there are two drawing buffers and you switch between them. Use double buffering only if the above solution alone does not work. The following code snippet describes how to do it.

```
Image offscreenImage;
Graphics offscreenGraphics;
offscreenImage = createImage(size().width, size().height);
offscreenGraphics = offscreenImage.getGraphics();
offscreenGraphics.drawImage(img, 10, 10, this);
g.drawImage(offscreenImage, 0, 0, this);
```

Q: The following app works fine, but when I start it, I cannot close it using the X at the right top of the form...

Please help me on the following. I'm just starting to use java (JDK1.3). The following app works fine, but when I start it, I cannot close it using the X at the right top of the form. What should I add to fix this problem? The source is shown below.

```
import java.awt.*;
import java.awt.event.*;

public class MyApplication extends Frame{
  public static void main(String args[]){
    Frame f = new Frame("MyApp");
    f.setLayout(new FlowLayout());
    f.add(new Button("A"));
    f.setVisible(true);
  }
}
```

Answer: You should add a listener to handle the closing of the window when the X box gets clicked on.

```
f.addWindowListener(new WindowAdapter(){
  public void windowClosing(WindowEvent evt){
    System.exit(0);
  }
});
```

Read on this, and other ways to do it, in the sections dealing with event handling, in whichever Java text you are using.

Q: How can I set a JFrame to be full screen at the start of a program? I want no borders, no titles and I just want to use the entire screen.

Answer: Try using JWindow instead, that one can be customized to have no borders or titles... as for size, I think you can use setBounds(GraphicsEnvironment.getLocalGraphicsEnvironment().

getDefaultScreenDevice().getDefaultConfiguration().getBounds());
to fill out the entire screen.

Q: Why can not I mix AWT and Swing?

Recently, I have been hearing a lot of people from various newsgroups and website saying, java swing and awt can't be in the same application. They will not work

together and they might produce unexpected results. At the same time, i don't hear people saying "why" you shouldn't use swing and awt together. Could someone out there shed some light for me. Is their any logical reason why we shouldn't mix swing and awt in the same application/applet. If there is a problem mixing swing and awt... what are the results, what can happen? I design using IBM's Visual Age for Java 3.0, and I mix swing and awt in the same application/applet, it works fine when testing in the IDE (I haven't tested it outside of the IDE yet). If you have tested application/applets outside of the IDE, please let me know what happened?

Answer: There are fundamental incompatibilities in the way they draw themselves. AWT java classes are not "pure" Java classes, they use underlaying C/C++ native code (dependable on operation system) that can cause different appearence in different OSs.
Swing is pure Java implementation and has no native code at all. Swing applications look the same.
> If there is a problem mixing swing and awt... what are the results,
> what can happen?

Some objects drawn on top of others are not properly occluded. This is most obvious with drop down menus, which have a tendency to stay visible even after you have selected a menu item. Another problem is that if you use AWT components on a JTabbedPane they will not disappear when you switch tabs. There are many similar issues.

Q: Again about difference between AWT and Swing

I have a question: What are the architectural differences between Swing and AWT??

Answer: by Odd Vinje
There are no big architectural differences, the class hierarchy is almost the same. The reason is that Swing is built upon AWT.

The most significant difference is how the components are drawn to the screen. AWT is so called heavyweight components and have their own viewport which sends the output to the screen. Swing is ligthweight components and does not write itself to the screen, but redirect it to the component it builds on. Heavyweight components also have their own z-ordering. This is the reason why you can't combine AWT and Swing in the same container. If you do, AWT will always be drawn on top of the Swing components.

You can combine AWT and Swing, just don't do it in the same container (e.g. panel, groupbox, etc.) and don't put a heavyweight component inside a lightweight.

Another difference is that Swing is pure Java, and therefore platform independent. Swing looks identically on all platforms, while AWT looks different on different platforms.

Q: I have a JFrame for customer registration with a lot of input fields. In this screen you can create a new customer, get customer with specified customer number and you can update a customer.

In this JFrame is it possible to clear all fields without specifying each field?

Answer: This snippet is for TextFields, you should be able to adapt for JtextFields

very easily.

```
public static void resetTextFields(Container c) {
  Component [] components = c.getComponents();
  for (int i = 0; i < components.length; i++ ) {
    if (components[i] instanceof Container)
      resetTextFields((Container) components[i]) ;
    else if (components[i] instanceof TextField)
      ((TextField) components[i]).setText("") ;
  }
}
```

Bye.
---
Real Gagnon from Quebec, Canada
* Looking for Java or PB snippets ? Visit Real's How-to
* http://www.rgagnon.com/howto.html

Q: Swing is "lightweight" components. Its graphics primitives are implemented in 100% Pure Java. How does it draw pixels on screen?

Does it still rely on any native code to access the frame buffer?

Answer:  We mentioned before in our tips that Swing components is 100% pure Java.
It isn't fully correctly if we are speaking about containers on which they can draw. Swing is still based on AWT, and even Swing components must have at least one heavyweight container. In other words, JFrame, JApplet are *not* lightweight.

Q: I need to put in a JtextField characters like cyrilic ...

How can I do this?

Answer: You use the unicode string \u???? corresponding to your character.
A table of codes can be found at www.unicode.org/charts/
by Charles Megnin

Q: Is there any way to accept password from a java application. As it is a password, IT SHOULD NOT BE SHOWN(ECHOED) WHILE IT IS ENTERED.

Answer: If you are using swing, you can use JPasswordField, and set your echo character.
by myparu

Q: How can I change the default icon on application window (java cup) to my own?

Answer:
window.setIconImage(Toolkit.getDefaultToolkit().getImage("image.gif"));

Q: How can I read a Tab delimited text file into a JList?

All I need is the first item in each record. I import each line with a BufferedReader.Readline() and then look at each char with record.substring(n,n+1). How can I check if the next char is a Tab?

Answer: StringTokenizer st = new StringTokenizer(record, "\t");
while (st.hasMoreTokens()) {
    String token = st.nextToken();
    // ... do something with the token
}

Q: How do I generate mouse click event but without a real click?

Could you tell me how I can generate an event (like a mouse click) from a Java application? (I want generate a mouse click event but without a real click on the mouse).

Answer: See "simulate keyboard pressing" below: You can use java.awt.Robot" since java1.3

Q: Why would anyone use AWT and not Swing today?

Answer: There are several handheld devices that do not have a lot of memory or CPU power available, AWT is more suitable for them (Smaller footprint).

Q: Need I to remove all the listeners on all of the components in the Frame when I close window of the Frame?

I've got a Frame, which has in it a bunch of components, some of which themselves may be containers. Many of the components in this frame have listeners.

When somebody closes the window of the Frame, I want to completely get rid of the Frame, so that the garbage collector will later clean it up, freeing it's memory. However, I haven't yet figured out a way to do this without tracking every single component by myself. Just calling dispose() on the main Frame doesn't seem to be good enough. Calling  "RemoveAll()", even recursively, doesn't seem to be good enough.

I *suspect* that the problem may be that I have to remove all the listeners on all of the components in the Frame. Is there a way to get a list of the listeners out of a given component? I really don't want to have to track every single button I add to every single Frame, and every single Listener I add to every single component. If I have to keep track of all of that, it sort of defeats a lot of the advantages of a well-defined object oriented system.

Answer: I think you're slightly confused here. When another object registers as a listener for a component within the frame, it's the component within the frame that is holding a reference to the external object in its listeners list - not vice versa.

Which means you don't really need to explicitly remove every listener that any other object registered on components in the frame.

What you need to worry about is however the listeners that the frame itself registered with some other components that are still alive. If the frame gets disposed without removing these, the objects that were being listened to will retain a reference to the frame and this can cause the frame to stay around as long as these objects which hold these references stay alive.

So look for all Listeners that your frame registered itself as on components external to itself (should be fairly easy to see since you normally do it in your constructor) and remove those before disposing off the frame.
--
Kala

Q: Main disadvantage of Java GUI is that it is often slow. But I have seen also very fast working GUIs. Unfortunately the code is hidden and I do not know the technique for writing of fast Java GUIs.

Answer: I can describe one of main technique you can use. It does not give you full solution, but will certainly speed up your GUI.

The main idea is to use "lazy" initialization. Instead of creating and initializing of all GUI components in constructors during start up time postpone it to later time until you really need. Let say, you have a lot of tab panels with many elements on each tab panel.
Your constructors should be "quite" empty and do not create and initialize those small elements until your tab panel is chosen. You should have very small constructor for tab panel itself and additional lazy constructor for the rest. It should be called when user clicks on that particular tab.
This does not decrease the full time of initialization, but spreads it up. The user will not feel big delays during start up of your application. You should use these lazy constructors just before you are going to call any paint method.
--
AP. (J.S.)

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

| | | | | |
|---|---|---|---|---|
| ☐ IT Professional | ☐ General Computer | ☐ Design and Graphics | ☐ Linux | ☐ Freeware/Shareware |
| ☐ Games | Intranet | Computer Hardware | Web Design | ☐ Computer Software |
| ☐ UNIX | ☐ Web Software | ☐ Windows Software | ☐ Personal Finance | ☐ Programming Software |
| ☐ Servers | ☐ C/C++ | ☐ Powerbuilder | ☐ Perl | ☐ XML |
| ☐ Java | ☐ SQL | ☐ Design/Graphics | ☐ Utilities | ☐ Handhelds |

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📕 Graphics, AWT, Swing - II

📗 **Q: I've a question where skillful java programmers sure will laugh about, but I ask nevertheless.**
**What exactly means >>> Graphics g <<<? How must I use it and why is it so important to use?**

Answer: It basically says: "where should I paint? On the screen, on some RAM, on a printer buffer?"
It is sort of a conduit to that medium.
Graphics class is the set of tools, which allows you to set background, colours, fonts; draw lines, circles, arcs, polygons etc to a component. Here is the definition from the class javadoc documentation:

"The Graphics class is the abstract base class for all graphics contexts that allow an application to draw onto components that are realized on various devices, as well as onto off-screen images.

A Graphics object encapsulates state information needed for the basic rendering operations that Java supports.
This state information includes the following properties:

The Component object on which to draw.
A translation origin for rendering and clipping coordinates.
The current clip.
The current color.
The current font.
The current logical pixel operation function (XOR or Paint).
The current XOR alternation color (see setXORMode(java.awt.Color)). "
--R. Green, Gilgames

📗 Q: Can anyone tell me how I could quickly draw a (changing) pixel array (int [] px) to the screen in an applet?

The only way I know is to create an image and then draw this image to the screen:

createImage(new MemoryImageSource(w, h,
ColorModel.getRGBdefault(), px, 0, w));

but that is incredible slow :( so there must be a much quicker way to draw these pixels. I need all this to achieve some special effects on the fly.

Answer: MemoryImageSource is a straightforward implementation of ImageProducer. It basically just sends the address of an int (or byte) array to a list of ImageConsumers. This means that most of the CPU time is in the ImageConsumer, not MemoryImageSource.
The speed of most "built-in" ImageConsumers (like the one that creates Images in createImage()) is limited by the Java Virtual Machine (JVM). If you need much higher performance, perhaps you could call a plug-in that allows the use of DirectX or OpenGL drivers (assuming you are using a compatible platform).

Make sure that you compare the time spent updating the values in the int array with the time required to render the pixels to an Image or display.
Many special effects spend more time calculating the new int array values.
Tricks to speed up rendering are platform dependent (i.e., they may not work on all system configurations). If you still find that it's the rendering time that limits performance, then here are a few things to try:

1. Only update the regions that are changing.
2. Using the default color model and an int array (versus a byte array) is usually faster, although using a byte array instead of an int array may reduce the amount of data transferred and be faster under some circumstances (even though a color-lookup table is involved).
3. Try another version of Java. I've heard that some graphics operations in Java 1.2 are slower than in Java 1.1 or Java 1.3.
4. Try using a color model that doesn't have transparency (an alpha channel). Some JVMs may notice this and use more efficient code when rendering the pixels on the screen.
5. If you can rapidly create a *.gif or *.jpg file in memory, you can create an Image directly from the data. Creating the Image may be slightly faster, but I think that the time spent placing the data in *.gif or *.jpg format may erase any performance gains.
6. If you are using double buffering, write directly to the off-screen Image (instead of writing to a separate Image, then rendering this Image to the off-screen buffer using drawImage).

--

Carl G.

Q: I'm resetting the TitleBorder's title via setTitle, but no change occurs....

I'm trying to make a dialog box containing a panel whose TitleBorder's text dynamically reflects the function of the list box within it. The title text and the list box contents should change according to the buttons pressed elsewhere in the dialog box. I'm resetting the TitleBorder's title via setTitle, but no change occurs. Do I need to update the dialog? If so, how do I do this?

Answer: Yes, you need to update the dialog.
How? Call the validate() and invalidate () methods

Q: everytime you create a dialog-object, the dialog pops up in the upper left corner of the screen. Now I want it to be automatically placed in the center of the application's frame.

Answer: private void center() {
   Dimension screenDim = Toolkit.getDefaultToolkit().getScreenSize();
   setLocation( ( ( screenDim.width - getSize().width ) / 2 ),
   ( ( screenDim.height - getSize().height ) / 2) );
}

You can call this anytime after calling pack() on the JDialog, before setVisible().
--
Dave Neuendorf

Q: How can i disable the right click on the label?

I have a JTree and a JPopup: i've forced the code to be able to select the node with a right click and show the popup. The problem is that i can right click on the popup labels to select them, but i'd like to select them only with a left click. How can i disable the right click on the label? Can somebody help me please?

Answer: You can add in an if loop that only executes if the left mouse button is down using code like the following:

MouseEvent e;
if ((e.getModifiers() & e.BUTTON1_MASK) != 0)
{ // code for left button click }

and just so you know,
InputEvent.BUTTON2_MASK is the middle or scroller button
InputEvent.BUTTON3_MASK is the right button in windows
--
by MSW

Q: How do I count the width in pixels of the text before drawing them into graphics?

I have the Font and the String and I am using them in an applet:

graphics.setFont(font);
graphics.drawString(str);

How do I count the width in pixels of the text before drawing them into graphics?

Answer: import java.awt.*;

Toolkit tk = java.awt.Toolkit.getDefaultToolkit();
FontMetrics fm = tk.getFontMetrics(new Font("Arial", Font.PLAIN, 10));
int width = fm.stringWidth("Your String");
--
Jarkko

Q: If I do (CustomFrame extends Frame) smth. like this:

Frame newFrame = new CustomFrame();

what are the advantages of doing this over:

CustomFrame newFrame = new CustomFrame();

Answer:
There is one difference.

If the reference is a Frame type you can always downcast it, but not if it
is a CustomFrame type.

Might be nice sometimes.
--
Soren ' Disky ' Reinke

Q: How can I have a filtered copy and paste in JTextPane. I am trying to copy and
paste styled text and remove all the new line '\n' characters from the copied text. So
when I paste the text in must still have its styles as well as no '\n' characters.

Answer: You could override the paste() method of JTextPane. (This is inherited from
JTextComponent). You can look at the source code of JTextComponent to see what it
does currently and what modifications will be needed for the modified action.
--
BK

Q: I want to know what is the difference between JScrollPane and JScrollBar...is
both same or not..?

Answer: No. A JScrollPane is a container for components which can grow larger than
the visible area that can be displayed (i.e., a JTable or a JTree, for example). When you
place a component like these into a JScrollPane then the visible area is displayed and
appropriate scrollbars presented to allow scrolling to non-visible areas.

A JScrollBar is just that, a scroll bar.

Read the Java docs for both classes and the differences should be most apparent.
--
Darryl L. http://welcome.to/mcpierce

Q: Is it possible to change delays that affect appearing, keeping and disappearing of
tooltip?

Answer: It was difficult to find the answer but finally I found in "Swing" book that is free
to you on our site.
The ToolTipManager is a service class that maintains a shared instance registered with
AppContext. We can access
the ToolTipManager directly by calling its static sharedInstance() method:

ToolTipManager toolTipManager = ToolTipManager.sharedInstance();

Internally this class uses three non-repeating Timers with delay times defaulting to 750,

500, and 4000. ToolTipManager uses these Timer's in coordination with mouse listeners to determine if and when to display a JToolTip with a component's specified tooltip text. When the mouse enters a components bounds ToolTipManager
will detect this and wait 750ms until displaying a JToolTip for that component. This is referred to as the initial delay time.
A JToolTip will stay visible for 4000ms or until we move the mouse outside of that component's bounds, whichever comes first. This is referred to as the dismiss delay time. The 500ms Timer represents the reshow delay time which specifies how soon the JToolTip we have just seen will appear again when this component is re-entered.
Each of these delay times can be set using ToolTipManager's setDismissDelay(), setInitialDelay(), and setReshowDelay() methods.

ToolTipManager is a very nice service to have implemented for us, but it does have significant limitations. When we construct our polygonal buttons we will find that it is not robust enough to support non-rectangular components. It is
also the case that JToolTips are only designed to allow a single line of text.

Q: How can I test if a JTextField is empty. My code is below but does not work.

if (myJTextField.getText() == " ")
    return true;
else
    return false;

Answer: You have to compare String content, not String handler...so you have to use the equals() method.
For example:

if (myJTextField.getText().equals(""))
    return true;
else
    return false;
Or
you can test whether text == null,
then check if myJTextField.getText().length() == 0
if it is not, do text=text.trim() to remove spaces,
then check text.equals("") to see if it is an empty String
--
Lucio Benfante, Arnaud Berger and Jorge Jordao

Q: Whats the difference between a Frame and a canvas? Can someone tell me the difference?

Answer:
A Frame is , well , a frame - a window, with borders,close/minimize/maximize buttons (usually), and can not contain other elements, such as a menubar ,buttons panels, etc.
a Canvas is a user interface element (which you can place in e.g. a frame) onto which you can draw/display something..
--
by Nils O. Selåsdal

Q: I know how to make a text box using Visual Basic that can only accept certain keys for example if I wanted to make a box where you can type only numbers in it or letters. However, when it comes to Java I have not been able to figure this out. Is there a

way to do this and make a textbox where you can only type letters in it?

Answer: First of all, 'ASCII' is not a synonym for 'text'. ASCII is just one of many ways of encoding text to numeric values. It is 7-bit and only includes values from 0 through 127.

Now, on to your question. With ASCII, 'numbers' and 'letters' are a little easier to check, since the only letters are 'A'-'Z' and 'a'-'z'. However, Java uses Unicode and has all sorts of fun extra characters.

But you do have a few handy methods on the java.lang.Character class such as isDigit and isLetter

Now, if you are using swing, then to achieve what I'm guessing you want is quite simple. Just hook a custom text document to the JTextField you're using and have it do the filtering.

** Here's the part of The Java Tutorial that goes over it
http://java.sun.com/docs/books/tutorial/uiswing/components/textfield.html#validation

Here's someone's Swing FAQ on that
http://users.vnet.net/wwake/swing/faq.html#text_check

Here's a JavaWorld article that describes how to do that to limit the length of input. Shouldn't be too hard for you to change to your specific needs.

http://www.javaworld.com/javaworld/javaqa/2000-05/02-qa-0512-textbox_p.html

All these links were from the first page of a search on http://www.google.com
Google is your friend!
--
John A.Cruz

Q: How can I force Java not to draw the background?

I have the following problem: I display a .gif image on a panel with the g.drawIcon() method. If I now display the image of a car on a track then the background of the image (black) gets displayed, too.
How can I force java not to draw the background?
I tried the setXORMode(Color col) method which filters the given color out but this ruins my image.

Answer: Just make your GIF image have a transparent background. Java already handles drawing GIFs with transparency. The black color in your car image is not the transparency color of your image.
--
dalal

Q: I was wondering how to create a Directory Dialog, just like the FileDialog class, but for selecting directories instead of files.

I'm sure this has been done, and I apologize if it is in a FAQ. Just point me in a direction, and I will go. Swing or AWT is fine.

Answer: In Swing, you can use JFileChooser and set it up to choose a directory. See the

API docs for details.
--
Chris Smith

📋 Q: Can any one please tell me whether I can create a Window/JFrame in Java that is not rectangular in shape?

If the answer is yes, then where can I get the information to do that?

Answer: Unfortunately answer is "No"...

📋 Q: I'm trying to do some image manipulation (mainly resizing images) in a Linux Servlet/JSP environment, to do that I use the Java 2D API.

This works fine on my Windows development PC, but does not work on an X-less Linux hosting platform. My hosting company is not running X and has no plans to do so. So my question is, it is possible to use the Java 2D API on a Linux box without an X server running?

Answer Part 1: A frequently asked question in the context of loading images is that of the missing X server. Unix users who just want to load an image via the AWT without displaying it get an error message if there is no X server installed. In fact, the X server is required for that.

java.awt.Graphics methods such as drawLine (), fillOval (), drawString (),... are implemented in the default JVM with native graphical functions (except in some cases for Java2D) : That means that drawLine () finally calls a GDI system function on Windows or X11 function on a X11/UNIX machine even if the drawing is done in an off-screen image using the class java.awt.Image. This ensures the best performance for drawing graphics with Java.
When no X11 Display is available on UNIX machines or when GDI resources are low on Windows, it is impossible to compute off-screen images with java.awt.Graphics methods, even if your program doesn't need to display these images. Typically, this situation happens for servlets returning dynamically generated images like pies, charts or web counters.
It is also impossible to compute off-screen images when the Java security manager forbids access to any Toolkit or the AWT library.

Possible solutions will be published tomorrow! (end of Part 1)

📋 Part 2 (Part 1 in yesterday's tip):

Possible solutions:

1. Install a virtual framebuffer X Windows System server for XFree86. Xvfb (X Virtual Frame Buffer) is an X server that is able to run on machines with no display hardware and no physical input devices.
Xvfb emulates a dumb framebuffer using virtual memory. Xvfb doesn't open any devices, but behaves otherwise as an X display. Xvfb is normally used for testing servers. Using Xvfb, the mfb or cfb code for any depth can be exercised without using real hardware that supports the desired depths. Xvfb has also been used to test X clients against unusual depths and screen configurations, to do batch processing with Xvfb as a background rendering engine, to do load
testing, to help with porting an X server to a new platform, and to provide an unobtrusive way of running applications which really don't need an X server but insist on having one.

2. Use PJA Toolkit available for free at:
http://www.eteks.com/pja/en/

3. Use Java 1.4. beta which available now. This version can work without X-server

📄 Q: how do I print a frame? I need this because I'm doing some graphics with java, and need to print them on paper. I'm using the AWT..

Answer: You would rather paint your graphics on a panel/canvas/JPanel than on a Frame/JFrame, I suppose, and the best would be to use the 2D-Print-API for this, there are a lot of example at Sun available and I have some at my own WebSite, if you need some examples:

http://java.sun.com/docs/books/tutorial/2d/printing/
http://www.jalice.net/discussion.htm

If you are restricted to use the java.awt.PrintJob, you can use something like this:

PrintJob myJob = getToolkit().getPrintJob(MyClass.this, "My Print", null);

```
if (myJob != null) {
   Graphics printgraph =
   myJob.getGraphics();
   if (printgraph != null) {
      printAll (myJob);
      printgraph.dispose();
   }
   myJob.end();
}
--
Linda Radecke
```

📄 Q: I'm currently developing my first actual graphical application in Java, using Swing. I feel that I've got a pretty good grip at object-oriented design and analysis, but I'm somewhat uncertain when it comes to connecting the user interface with the application logic.

Should the GUI components be in the main application class, separated into several classes depending on their position on the screen, or included in the "application logic"-classes? What about the different EventListener I need?

Answer: The biggest, biggest mistake people frequently make is to embed application logic right in with their GUI code. It is almost inevitable with most programmers, unfortunately.

One idea is to use top-level packages, perhaps GUI, kernel, and database.
In the GUI package, have one package for each frame or main thing, like gui/main, gui/login, gui/help, gui/detail, etc.

From all the many tiny pieces of gui code that see button pushes, get events, enable or disable things based on user authorization level, etc, do *NOT* embed application logic there.

Make all these tiny pieces of gui code call a separate NON-GUI arbiter interface (not class) that has methods like:
* void initialize ()

* void beginSearch (String phoneNumber)
* void login (String user, String password)
* void help (String subject)
* void exitApplication ()

The GUI calls an interface (the arbiter), and the arbiter calls a simple interface to the GUI. This is one scheme that separates the GUI from application logic.
--
Max Polk

Q: I would like to display images of various sizes using JInternalFrame.

1. I want to create a JPanel and paint the image on this JPanel.
2. Create a JScrollPane and add the JPanel to the JScrollPane.
3. Add the JScrollPane into the JInternalFrame.
4. I want to set the size of JInternalFrame to fit the image size.

I am having trouble with the setSize() method of JInternalFrame.
If I don't use this method, the JInternalFrame doesn't appear.
But I cannot seem to find a good way to set its size to match
the image size.

Answer: Basically you have three ways to set the size of JInternalFrame:
You can use either: setSize, pack, or setBounds. When you call pack() instead of setSize, the JInternalFrame calculate the size based on it's content inside.
One way would be to use ImageIcons wrapped into JLabels, add them to your component and call pack() on the JInternalFrame instead of a fixed size.

As an alternative way instead of overriding the paintComponent() of your JPanel, what you could do as well, of course and decide whether you like to draw the image on it's original size, scaled or tiled. But you will then need to determine in turn the size of your JPanel, IMO. Thus, I think, it might be the better approach to use ImageIcons instead.
--
Linda

Q: While compiling I get an error that says something about a JPanel that is accessed from within inner class and needs to be declared final. That's OK. It can be as final as it wants to. BUT...

What does it mean when a JPanel is final?

and...

Why in the name of the Lord does the compiler want it to be declared as final?

Answer: A JPanel isn't ever final. The compiler is telling you that the variable referencing the JPanel must be final. When a variable is declared as final, it means that the variable cannot be pointed at some other object (or contain some other value in the case of primitives).
For instance, this code would produce an error:
final int foo = 42;
foo = 37;

When using inner classes, any local variables which are used inside the inner class

must be declared final. The compiler requires that the reference be final because the inner class makes a copy of the local variable value so that it can use it. This is fine as long as the two copies of the value don't get 'out-of-synch'.

By requiring that the variables be final, the copies are guaranteed to always point to the same object (or contain the same value for primitives).

See the Inner Class Specification for details:

http://java.sun.com/products/jdk/1.1/docs/guide/innerclasses/spec/innerclasses.doc.html

--

Jim Sculley

Q: I need some picture icons for my application that I'm building. Could someone point me to some place where I could get some pictures?

Answer: We have the copy of Java TM Look and Feel Graphics Repository on our site here:

http://javafaq.nu/java/graphics/java-look-and-feel.shtml

--

John

Q: I need to create image from two smaller images (two parts of map) and save it. I tried to do it with JAI, but I was able only to paste images over each other.

Answer: The usual way of doing this is creating BufferedImage and painting those two images on its Graphics.

1. you create empty BufferedImage of size of total size your two source images.

2. get graphics

3. draw your source images in appropriate location

4. enjoy result

--

Dmitry R

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## General Java Questions - I

### Q: Is JavaScript the same as Java?

Answer: NO! An Amazingly large number of people, including many web designers, don't understand the difference between Java and JavaScript. Briefly it can be summed up as follows:

Java was developed by Sun Microsystems. Java is a full-fledged object-oriented programming language. It can be used to create standalone applications and applet. Applets are downloaded as separate files to your browser alongside an HTML document, and provide an infinite variety of added functionality to the Web site you are visiting. The displayed results of applets can appear to be embedded in an HTML page (e.g., the scrolling banner message that is so common on Java-enhanced sites), but the Java code arrives as a separate file.

JavaScript on the other hand was developed by Netscape, is a smaller and simpler scripting language that does not create applets or standalone applications. In its most common form today, JavaScript resides inside HTML documents, and can provide levels of interactivity far beyond typically flat HTML pages -- without the need for server-based CGI (Common Gateway Interface) programs.

Some server software, such as Netscape's SuiteSpot, lets web application developers write CGI programs in a server-side version of JavaScript. Both client-side and server-side JavaScript share the same core JavaScript language, but each side deals with different kinds of objects. Client-side objects are predominantly the components of an HTML web page (e.g., forms, text boxes, buttons). Server-side objects are those that facilitate the handling of requests that come from clients, as well as connectivity to databases.

### Q: Is Java open source as distributed by Sun, i.e., all the modules including JVMs?

If not, is anyone else doing an open source implementation?

Answer: Java is not open source project. Though you can get the full source code under a Sun license.

http://kaffe.org does open source implementation.
I read there: "Kaffe is a cleanroom, open source implementation of a Java virtual machine and class libraries. It is also a fun project that was started by Tim Wilkinson and was made successful by the contributions of numerous people from all over the world.
But Kaffe is not finished yet! You can help by developing new and missing functionality, porting Kaffe to new platforms, and testing your Java applications under Kaffe.
Kaffe mostly complies with JDK 1.1, except for a few missing parts.
Parts of it are already JDK 1.2 (Java 2) compatible."
--
John
http://gcc.gnu.org/java/ - The GNU Compiler for the Javatm Programming Language

Q: I will be thankful if anyone tells me why JVM is called virtual machine.

Answer: JVM is called a virtual machine because there is no real hardware which interprets the byte code. If you have done any assembly programming for any microprocessor/microcontroller you will able to understand this. A microprocessor has builtin instruction set to interpret the assemly code. Similarly the JVM is similar to a microprocessor in the sense it has its own instruction set but it implemented in software. That is why it is called a virtual machine!

Q: Do anyone know the difference between java and C#.

Answer: They are different languages. Java has been around for about five years. C# has not been publicly released yet. One is written by Sun Microsystems, one my Microsoft. They are fairly similar languages with C# having a few extra bits added on to it.
--
Phil
C# bytecodes can be compiled to native exe files just as Java bytecodes can be. But C# is expected to be more closely tied to the Windows operating system and standard interfaces that are part and parcel of Windows. Writing a native compiler that collects all these interfaces and combines them into a unified whole that can run on ANY operating system may require compiling proprietary windows components which Microsoft will make sure is hard to do and against its licensing policies. So you can expect to see native compilers that compile for Windows platforms but not to other operating systems.
--
alankarmisra

Q: I read PHP 4 times faster than JSP. Why then do we need JSP?

Answer: These tools fill somewhat different niches and seldom directly compete.

PHP is good for situations where your page can interact more or less directly with a database, without the need for complex business logic. Its strength is that it can be used to build pages VERY quickly. And, as you note, they run very quickly as well.

The JSP/Servlet model is more geared toward distributed n-tier applications where

there is at least logical, and possibly physical, separation of model, view, and controller functions. It is more complex than PHP, but also more scalable, and well-written Java apps may be a great deal more maintainable because of the separation of logical tiers.

They're both nice tools, and I use both, but I don't think either one is going to kill the other anytime soon.
--
Joe

Q: My question is : is JSP as powerful as servlet?

I heard that JSP will eventually compile into servlet class file.
One thing can be done by servlet, can it be done by JSP too? In terms of http.

Answer: Everything a servlet does can be done in JSP and vice versa. Good programming practice (you will see some articles over the last year in JavaPro) dictates to combine servlets and JSP in any significant web application.
JSP should be mainly HTML (or XML, or WML or whateverML) with little Java inside.
Servlets should be Java with few or not at all lines like this:
out.println( "<html>" );
out.printlb( "<body>" );
This creates a more or less clean separation between presentation (JSP) and business logic (servlet).
Java beans also have a role in this. I strongly recommend the JavaPro articles or whatever text on the MVC model you can find.
--
eugene aresteanu

Q: I do not understand the difference between clone() and new(). Why shouldn't I just call new()?
Can apply new() to cloned object?

Answer: The difference is that clone() creates the object that possibly has changed fields (after start up) and this object is willing to be cloned by implementation Cloneable interface.
New() method can be applied without objects "permission" and create a new instance with fields just initialized and not changed. They will be changed later during runtime.
Clone() "mirroring" this modified object.
That's why you NEVER should call new() on cloned object. You can destroy the clone...
--
AP. (J.A.)

Q: Doesn't Java have C's preprocessor functions? At least #define's? If not, the only way to define constants is with 'final'? So... no macros at all??

Isn't there, then, a separate "preprocessor" utility (at least), if I want to define things like
#define proc public void
or so?

Answer: The answer to that is either "No" or "No, thank God". Code gets read many, many more times than it gets written. You should be constantly making things easy

for the developers that have to maintain this after you've finished it (after all, it's probably going to be you).
That's way Sun pushes its Java coding standards so hard.
Believe it or not, the only reason that I don't do much C++ work boils down to three things that C++ has that makes its code hideous- the preprocessor, typedef, and operator overloading. All three of those are wholly unnecessary and mung C++ code six ways to Sunday.

When C# was announced, I was excited because I'd heard its preprocessor was more restrictive. It is, in that it doesn't permit macros. It still lets you do all sorts of whacky "conditional compiling" stuff, though, that makes code unreadable again.
--
Cuplan

Q: I'm just starting to learn Java on my own. Should I first learn AWT or should I jump directly into the Swing of things?

Answer: Will you be wanting to code applets that are easy for anyone to run in their browser? If so, you'll probably have to go with the AWT for now.

The AWT isn't so bad, but Swing makes a lot of things much easier, so if you want to ship native-code applications or suchlike then I'd go with Swing. I still use the AWT, but I find myself having to code a lot of 'standard' things myself.
--
Mark
Swing make things easier but IE doesn't support it.

Q: I can't manipulate inodes on my linux box ... in fact I can't even get real info about a file! Java is a bad hack and is for kids who aren't sharp enough to do C++.

Answer: Think of Java in the same terms as COBOL and VB, and you've got the right idea. Start thinking of it as a replacement for C++ and you're on the wrong track. Don't expect this portable language to be a tool for low-level coding with hooks into the OS and hardware internals. It just wasn't designed for that. It's an excellent *applications* language, not a *systems* language like C or assembler.

On the other hand, if any pesky Java programmers tell you that C++ is dead and that Java can do everything C++ does, and more, you may howl with laugher and tell them to eat their JVM.
--
David Ehrens

Q: I wonder what happened if I remove "deprecations" from my code, for example size() and put getSize().

Don't the programs work any more on older browsers (e.g. IE3)?

Answer:  Check the docs to see whether they say "Since 1.1", "Since 1.2" or "Since 1.3" - if so, they will not work in the oldest MS VM.
Keep in mind that Sun's programmers haven't been very good at actually documenting this for all classes and methods.
or directly here:

To check your code against any version of the JRE (1.1, 1.2, 1.3), use

JavaPureCheck: http://java.sun.com/100percent/
--
Marco

📋 Q: How do we exchange data between Java and JavaScript and vice-versa?

Answer: Public variable and methods of Java Applet are visible to a HTML document. So using JavaScript you can access the public variables and public functions.

The syntax is:

var some_var = document.appletname.variable_name

With this you will get the value of the variable variable_name in your JavaScript variable some_var.

📋 Q: Constructors and methods: are they the same?

I need a little help here...I have been teaching that constructors are not methods. This is for several reasons, but mainly because JLS says "constructors are not members" and members are "classes, interfaces, fields, and methods."
So, now the rest of the staff is ganging up on me and making life a little nasty. They quote Deitel and Deitel, and Core Java (which references "constructor methods") and who knows how many other books.
The one we are teaching in is loaded with so many errors that even though it calls constructors methods NOBODY will quote it as an authority.
How can so many people call constructors methods if they aren't.

Okay. Are they or aren't they? I holding to the definition that they are not unless it is so common to call them that, that I will have to change.
Comments?

Answer: If you go by the JLS (Java Language Specification) and the Java API (and you should) , then no, constructors are not methods. Consider that Class.getMethods() returns an array of Method instances and Class.getConstructors() returns an array of Constructor instances, and Constructor and Method or not interchangeable (one is not derived from the other), but both implement the Member interface. Seems to me that Java is going out of its way to differentiate them.

Besides, the mechanics of constructors are so different from the mechanics of methods, there seems to be no value to considering one a member of the set of the other.
Now, as far as teaching the language goes:

Methods:
+ return types
+ called by name
+ executed multiple times

Constructors:
+ super(...) or this(...) as first instructor (often implicit)
- everything else

I very much do not like trying to lump the two concepts together, especially in introductory courses. Conceptually they are very, very different things.
A constructor is a unique element (even if there are several). It has the name of the class, its declaration is different, and it doesn't have the same syntax as a method. It plays a unique role. You can call a method from a constructor, but you cannot call a constructor from a method.

I say constructors are distinct from methods, and for students, blurring the distinction will lead to problems.
--
by Chuck McCorvey, Chris Wolfe, Paul Lutus

Q: I the see method getClass() in java.lang.Object. Do we need it? I know all my classes.

Answer: Exactly. If you know - you do need it. But if you do not know then it helps you. For example if you get some object and would like to instantiate it:

```
Object createNewInstOf(Object obj) {
   return obj.getClass().newInstance();
}
```
--
Igor

Q:I know that a default constructor is being defined from line 6 to 9 (see below). But I don't quite understand the syntax: this(blah, blah). Surely I know that "this" refers to the current object. But what on earth does "this(blah, blah, blah)" mean? Would you please help explain in what kind of situation we need to use this kind of statement?

```
1. public class ThreadPool implements Runnable
2. { private final int DEFAULT_MINIMUM_SIZE=5;
3. private final int DEFAULT_MAXIMUM_SIZE=10;
4. private final int DEFAULT_RELEASE_DELAY=10*1000;
5. ...
6. public ThreadPool()
7. { this(DEFAULT_MINIMUM_SIZE, DEFAULT_MAXIMUM_SIZE,
8. DEFAULT_RELEASE_DELAY);
9. }
10. ......
11. }
```

Answer: Whenever you encounter the :

this(blah, blah)

syntax, it means that another constructor should be called first:

```
public class MyClass{
   MyClass(){
      this(2,2); // here you make a call to the other constructor
```

```
  }

  MyClass(int a, int b){
  }
}
```

The point herecan be, that even if the user doesn't know which parameters to pass to MyClass(int, int) , she gets a default constructor which indirectly gives default parameters.

Important notes :

-calling this(...) is very similar to calling super(...) .
-indeed, this(..) may only be used inside a constructor, and may only be placed as the first instruction in a constructor (all like super(...) ).

note that super(...) will call some constructor from the parent class.
--
Arnaud.

Q: Simple question: why constructor doesn't work in following example?

```
class Start {
  public void Start() {
    System.out.println("Konstruktor START");
  }
}

public class Test {
  public static void main(String[] args) {
    Start s = new Start();
  }
}
```

Answer: Because you have included the return-type 'void' in the method declaration, it becomes a normal method, that just happens to have the same name as the class - so it won't get used as a constructor. Remove the 'void' and it should work.
--
Vince Bowdren

P.S. by John: If you do not specifically define any constructors, the compiler inserts an invisible zero parameter constructor "behind the scenes". Often this is of only theoretical importance, but the important qualification is that you only get a default zero parameter constructor if you do not create any of your own.
Your program used this zero parameter constructor and you saw nothing...

Q: Why we can not declare constructor as final?

Answer: The keyword final when dealing with methods means the method cannot be overridden.
Because constructors are never inherited and so will never have the oportunity to be overridden, final would have no meaning to a constructor.

Q: In Java, does exist a function like sprintf in C?

Answer: http://www.efd.lth.se/~d93hb/java/printf/index.html a free Java version of fprintf(), printf() and sprintf() - hb.format package

Q: If I declare an array of an objects, say Dogs, is that memory taken when I create the array or when I create the objects in the aray when I declare this array:

Dog[] dog = new Dog[100];

or does it take the memory when I actually create the Dogs in the array eg:

for(int i = 0;i<dog.length;i++)dog[i] = new Dog();

Answer: The statement above is actually two-fold. It is the declaration and initialisation of the array. Dog[] dog is the declaration, and all this does is declare a variable of type Dog[], currently pointing to null.
 You then initialise the array with new Dog[100], which will create 100 elements in the array, all of them referencing null.
It is important to realise that the elements of an array are not actually objects, they only reference objects which exist elsewhere in memory. When you actually create the Dog objects with new Dog(), these objects are created somewhere in memory and the elements in the array now point to these objects.
Pedant point:
Nothing ever points to null. It is a constant that represents the value of a reference variable that is not a pointer to some object  new Dog[100] creates an array of 100 null Dog references.

Q: How do I return more than one value using the return command?

Answer: You could make a new object/class that contains these two values and return it. For example:
Define an object like this:

```
class MyObj {
   public int myInt;
   public double myDouble;
}
```

Then, in your method create one of these, set the corresponding values, and return it.

```
MyObj yourMethod() {
   MyObj obj = new MyObj()
   obj.myInt = 20;
   obj.myDouble = 1.0003
   return obj;
}
```

Q: How do I use object serializtion for an object that has other objects as data member? Do both the class need to implement serialize?
How about static data?

```
class A{
}
class B{
   public A a;
}
```

Answer: Both the object and all the object references it contains need to belong to classes that implement Serializable.
Static and transient fields are not serialized. For more, see,
http://java.sun.com/docs/books/tutorial/essential/io/serialization.html

Q: I recently learned a bit about "inner classes" but this seems to be different...

I'm a bit new to Java programming so bear with me. My employer bought a package of java graphics library programs to support some chart applets we want to create. We have the source code. I'm trying to create a jar file with all the files I need to run the applet. When I currently run the applet, the browser java tool says that it can't find "TextComponent$1.class". I recently learned a bit about "inner classes" but this seems to be different. The "TextComponent.java" file does contain some inner classes, but not a class called "1". I'm confused. Is this an inner class? Or is it something else. Any help would be appreciated. Thanks...

Answer: The TextComponent$1.class is the first anonymous class defined in TextComponent.java. Since nested (inner) classes are compiled to their own .class file, they needed unique names. The javac compiler is just creating a unique file name for an anonymous nested class.

Q: Hi there, does anybody know a good source of design patterns written in JAVA?

Answer: A pretty good (free to download) book.

http://www.patterndepot.com/put/8/JavaPatterns.htm

Q: Whats the difference between the two: System.err. and System.out? When should we use System.err?

Answer 1: System.out leads the output to the standard output stream (normally mapped to your console screen), System.err leads the output to the standard error stream (by default the console, too). the standard output should be used for regular program output, the standard error for errormessages. If you start your console program regularly both message types will appear on your screen.

But you may redirect both streams to different destinations (e.g. files), e.g. if you want to create an error log file where you don't want to be the regualr output in.

On an UNIX you may redirect the output as follows:

java yourprog.class >output.log 2>error.log

this causes your regular output (using System.out) to be stored in output.log and your error messages (using System.err) to be stored in error.log

Answer 2: System.err is a "special" pipe that usually is directed to the standard consolle. You can redirect the System.out with the normal pipe control (| or >), but System.err no. If you want to put both the "normal" output and the "error" output to a file you must use the special redirect 2>.

This allow you to send normal messages into a file or in the /null black hole, but still receive the error messages on the console.

 What is the essential difference between an abstract class and an interface? What dictates the choice of one over the other?

Answer: You can only extend one class (abstract or not) whereas you can always implement one or more interfaces. Interfaces are Java's way to support multiple inheritance.

 Does anyone know how could I get the size of an Enumeration object? The API for Enumeration only contains getNext() and next().

Answer 1: You can't. Theoretically, some classes that implement Enumeration may also provide some way to get a size, but you'd have to know about the more specific run-time type and cast to it... and none of the standard java.util Collections classes nor Vector or such provide these methods in their Enumeration implementations.

Answer 2:  You can make your own class like this:

```java
import java.util.*;

public class MyEnumeration{
  int size;
  int index = 0;
  Enumeration e;

  public MyEnumeration(Vector v){
    size = v.size();
    e = v.elements();
    index = 0;
  }

  public boolean hasMoreElements(){
    return e.hasMoreElements();
  }

  public Object nextElement(){
    index++;
    return e.nextElement();
  }

  public int size(){
    return size;
  }

  public int getIndex(){
```

```
      return index;
    }
}
```
--
by Nicolas Delbing and Victor Vishnyakov

Q: Is there a way to provide values for a Vector in the source code, analogous to array initializers?

Answer: The Vector class constuctors take no arguments other than Collection (since JDK 1.2), which is abstract, and since a Vector is a structure whose size can change dynamically, it's contents can only be initialaized through member methods.
--
Mike Lundy

Q: How would I add a help file to a java application?

Would it have to be platform specific, or is there a Java api for making help files? If so, what is it?

Answer: See JavaHelp at http://www.javasoft.com/products/javahelp/
 you create HTML pages for the main text, and add some XML files for a hierarchical table of contents and a map from TOC tags to relative URLs giving document locations.

Q: What is a Just-In-Time(JIT) compiler?

Answer: It is a JVM that compiles Java instructions (called bytecode) into native machine instructions at run time and then uses this compiled native code when the corresponding Java code is needed. This eliminates the constant overhead of interpretation which tradition first generation JVM's used.
--
Dave Lee

Q: Is there a collection object like the hashmap or hashtable that stores values in an ordered path? Vector does this but i need the key/value functionality. hashmaps do not guarantee the order of the objects.

Answer: Take a look at java.util.TreeMap.
Red-Black tree based implementation of the SortedMap interface. This class guarantees that the map will be in ascending key order, sorted according to the natural order for the key's class (see Comparable), or by the comparator provided at creation time, depending on which constructor is used.

Note that this implementation is not synchronized. If multiple threads access a map concurrently, and at least one of the threads modifies the map structurally, it must be synchronized externally.

Q: Most people asked why there is an error, but my question is why this is NOT an error...

Please take a look:

r is a number and s is a character, why can I put them together to make a comparison without compilation error? Could somebody tell me... thank you

```
double r = 34.5;
char s = 'c';
if (r > s) {
    System.out.println("r > s");
} else {
    System.out.println("r < s");
}
```

Answer 2: Yes, char is indeed a 16-bit value. However, the actual answer is in the Java
Language Specification, section 5.6.2, which is at the following URL:
http://java.sun.com/docs/books/jls/second_edition/html/conversions.doc.html#170983

In summary, the char is automagically promoted to a double. No explicit cast is necessary since the language rules say that it gets "promoted" to a double
by John O'Conner

Q: == and equals ()... These two still make me confuse a lot of time.

Can somebody give me some thumb rule or explain it to me?

Answer: When you use == with a primitive -int, double, char, ... you are checking that the values are identical. But if you use == with an object, you are checking that the 2 objects are stored at the same address. In other words the references pointing to the same object...

Method equals () is different.

It is the same as ==, if it isn't overriden by the object class.

Many classes override the method equals (). In this case this method will check that content of the object is the same or not, not addresses.

Q: Why do I get message like "wrong magic number" when I am trying to run applet? What is magic number?

Answer: The first thing a JVM does when it loads a class is check that the first four bytes are (in hex) CA FE BA BE. This is the "magic number" and thats why you are getting that error, you are trying to load a file that isnt a class and so the class loader in the JVM is throwing out that exception.

Make sure you transfer the class files to site in binary mode, rather than text or ASCII mode.
An error from the browser saying "cannot start applet ... bad magic number" usually means that one of the class files on the server is corrupted. '
Replace your class binary files on the web server; clean up the cache of your browser, and reload your applet.

Q: In java, I found a lot of methods, which let you enter a line (read (), readLine () e.c.t). They all wait until return is pressed, and then start providing you the information.

Does anyone know if there is a read method available whit the desired behaviour, i.e. which doesn't wait for return being pressed?

Answer: Java does not provide it, the terminal itself waits until return is pressed before sending the entered line to Java.

You need to use some platform specific mechanism to change the terminal settings.

Q: Can I pass a function as a parameter in Java? If so, how?

Answer: No, you cannot. But you can pass an object with method and then just use this method like this:

myFunction (object); // object has method useMyMethod(){ do smth here...}
..
..
somewhere in another place where use this function..
object.useMyMethod();

Q: What is difference capacity() and size() methods for vector?

 What is difference between

public final int capacity()
Returns the current capacity of this vector.

and

public final int size()
Returns the number of components in this vector.

Answer: please read this method: ensureCapacity
public final synchronized void ensureCapacity(int minCapacity)

Increases the capacity of this vector, if necessary, to ensure that it can hold at least the number of components specified by the minimum capacity argument.
Parameters:
minCapacity - the desired minimum capacity.

Q: What're the differences between classpath and import in the java application?

Do I need to have the package in the classpath first before importing in a java application or need not?

Answer: Classpath is an environment variable of your OS, you have to set it (or better: Java sets it) to tell Java where to search for classes.
You use import in a program to let Java search for the specified classes within the classpath. This implies that the package must be in the classpath.
--
Stephan Effelsberg

Q: What is difference between Iterator and Enumeration?

First of all Java FAQ Team wish you !!!HAPPY NEW YEAR!!! and then

Answer: from
http://java.sun.com/docs/books/tutorial/collections/interfaces/collection.html

The object returned by the iterator method deserves special mention. It is an Iterator,

which is very similar to an Enumeration, but differs in two respects:
Iterator allows the caller to remove elements from the underlying collection during the
iteration with well-defined semantics.
Method names have been improved.
The first point is important: There was no safe way to remove elements from a
collection while traversing it with an Enumeration. The semantics of this operation
were ill-defined, and differed from implementation to implementation.
The Iterator interface is shown below:

```
public interface Iterator {
   boolean hasNext();
   Object next();
   void remove(); // Optional
}
```

The hasNext method is identical in function to Enumeration.hasMoreElements, and
the next method is identical in function to Enumeration.nextElement. The remove
method removes from the underlying Collection the last element that was returned by
next. The remove method may be called only once per call to next, and throws an
exception if this condition is violated. Note that Iterator.remove is the only safe way to
modify a collection during iteration; the behavior is unspecified if the underlying
collection is modified in any other way while the iteration is in progress.
The following snippet shows you how to use an Iterator to filter a Collection, that is, to
traverse the collection, removing every element that does not satisfy some condition:

```
static void filter(Collection c) {
   for (Iterator i = c.iterator(); i.hasNext(); )
      if (!cond(i.next()))
      i.remove();
}
```

Two things should be kept in mind when looking at this simple piece of code:
The code is polymorphic: it works for any Collection that supports element removal,
regardless of implementation. That's how easy it is to write a polymorphic algorithm
under the collections framework!
It would have been impossible to write this using Enumeration instead of Iterator,
because there's no safe way to remove an element from a collection while traversing
it with an Enumeration.

Q: How can I find the first dimension length of the 2-dimenstions array? I have use
the array[].length but it does not work, how can I solve this problem?

Answer: Java doesn't really have "multidimensional arrays", only arrays of arrays. So
try: array[0].length and you will get this dimension.

Q: I guess what I'm asking is "Is java.util.Hashtable thread safe?"

It's been a while since I've used hashtables for anything significant, but I seem to
recall the get() and put() methods being synchronized.
The JavaDocs don't reflect this. They simply say that the class Hashtable is
synchronized. What can I assume? If several threads access the hashtable at the
same time (assuming they are not modifying the same entry), the operations will
succeed, right? I guess what I'm asking is "Is java.util.Hashtable thread safe?"

Answer: That is right! It is recommendable, if you have questions like these, always look at source for the API, it's freely available.

Q: I try to copy an object of my own using the clone() method from java.lang.Object, but this is a protected method so I can't use it. Is there some other way to get my objective of duplicating an arbitrary object?

Answer: If you want to clone your object, you need to make it cloneable. To achieve this, you need to do two things:

1. implement the interface Cloneable

2. override the method clone(), so that it
a. becomes public
b. calls super.clone()
c. if necessary, clones any members, or
d. if a member can't be cloned, creates a new instance.


Simple example:

```
public MyClass implements Cloneable {
   int someNumber;
   String someString;

   public Object clone() {
   // primitives and Strings are no
   // problem
      return super.clone();
   }
}
```

In this case the method clone() of the class MyClass returns a new instance of MyClass, where all members have exactly the same value. That means, the object reference 'someString' points to the same object. This is called a shallow copy. In many cases this is no problem. Strings are immutable and you do not need a new copy. But if you need new copies of members, you have to do it in the clone() method. Here is another simple example:


```
public class SomeMember implements Cloneable {
   long someLong;

   public Object clone() {
      return super.clone();
   }
}
```


```
public AnotherClass extends MyClass {
   SomeMember someMember;
```

```
    public Object clone() {
        AnotherClass ac = (AnotherClass)(super.clone());
        if (someMember != null) {
            ac.someMember = (SomeMember)(someMember.clone());
        }
        return ac;
    }
}
```

Note that the class AnotherClass, that extends MyClass, automatically becomes Cloneable, because MyClass is Cloneable.

Also note, that super.clone() always returns an Object of the type of the actual object, although the superclass doesn't know anything about that sub class. The reason is, that Object.clone() is a native method, which just allocates new memory for the new object and copies the bytes to that memory. Native code has it's own ways of finding out which type to return ;-)
--
Karl Schmidt

Q: I was just wondering about the usefulness of Interfaces...

I was just wondering about the usefulness of Interfaces. I was under the impression that interfaces could be used to perform multiple inheritance. But an interface only declares a method - in a very abstract way.

A class that implements an interface needs to define its own implementation of a certain method. What is the use of having an interface when nothing is being gained...?

Answer: If two classes implements the same interface, you can get a reference to the interface instead of the effective class without bother what class are you managing.

This is very useful in RMI (for example) or in any condition when you have to take an object without knowing exactly his class, but only the interface that it implement.

For example:
public void recurseList( List l )

the generic List ensure that you can use every List for this method (ArrayList, AbstractList, Vector...), so your calling method can be:

ArrayList l = new ArrayList(); or
Vector l = new Vector();

recurseList( l );

Without any problem.

by Davide Bianchi

Q: I got a problem with an array/vector...

I got a problem with an array/vector.

my class contains a member:
static Vector quad[][];
....

in my constructor I got:
Vector quad[][] = new Vector[row][col];
for (int i = 0; i < row; i++){
   for (int j = 0; j < col; j++){
      quad[i][j] = new Vector (0,1);
   }
}

// row and col are int between (10..50) -- it's a big I know, but that
might not be the problem

My PROBLEM (and I don't know what to do, really), I can't access quad[x][y] outside
of the constructor!!!! Within the constructor I've got full access on quad[x][x]. Java
(1.2) returns a NullPointerException on any method except within the constructor!!!

What's my fault!???

Answer: I guess you shouldn't write Vector here:
Vector quad[][] = new Vector[row][col];
so, the correct variant may be:
quad[][] = new Vector[row][col];

I guess You just overridden your static variable with one defined in your constructor:
Vector quad[][].
Thus, you're initializing NOT your class-scope static variable but your
constructor-scope quad. It's not reachable outside the constructor. And as for static
quad, it has never been initialized! And a first reference to it causes
NullPointerException. I guess. I hope I'm right :)
--
Xao Rom

Q: I am looking for more efficient compression APIs esp. for *.doc and *.html files.

I need some Java APIs wherein I can implement various compression algorithms. I
have already the GZIP compression technique shipped with JDK's java.util.zip
package. I am looking for more efficient compression APIs esp. for *.doc and *.html
files.

Answer: You often get better compression results for a large number of smaller files
by concatenating the uncompressed files and then compressing the result. That's
why tar.gz is often superior to zip for large numbers of html files (like API docs).

You will not get much better general-purpose compression than Zip, at least not until
you have a specific class of data and know very much about its structure. Even then
you'll need some experience and time working on a better compression method.

You might want to take a look at this Java implementation of bzip2:

http://www.aftexsw.com/aftex/products/java/bzip/
It uses more CPU cycles and memory but typically compresses better
than zip.
--
Marco

Q: I need to programmatically replace an entry in a zip file.

I could not quite get it using the ZipOutputStream because it simply creates a new file
and write only that entry for me. The rest of the original entries are gone.
Does anyone have a solution for this?

Answer:

1) Read the file (myfile.properties) out of the original Zip Archive (original.zip)
2) Make your changes, write the file to the file system
3) Create a New Archive (originalNew.zip)
4) Write your edited file (myfile.properties) to originalNew.zip
5) loop through all the entries in the original Zip archive (original.zip), adding them to
the new archive (originalNew.zip) EXCEPT for the file you are replacing
(myfile.properties)
6) When you're done, erase the original and rename your new one to original.zip.

I believe that this may be the only way to do this, since there doesn't seem to be any
random access in the ZIP file.
--
Kevin T. Smith

Q: About the order of my elements in hastable...

I save some data into hashtable.
    For example,

```
hashtable.put ( "1", "one" );
hashtable.put ( "2", "two" );
hashtable.put ("3" , "three" );

when I get back the element from hashtable using Enumeration class ,
Enumeraton e = hashtable.keys();
while ( e.hasMoreElements() ){
   Object k = e.nextElement();
   Object v = hashtable.get (k );
}
```

the default result is
"3","three"
"2", "two"
"1", "one"

here I want to get the data sorted as ascending. ( the order as I insert ) such as
"1", "one"
"2", "two"
"3", "three"

or is it possible to get the data from end of hashtable?

Answer: When you insert elements into a Hashtable, they will not be stored in the order you insert them. They are stored in a way that makes it easy to find them by the key you specify.

So, you must either use another data structure (e.g. Vector) or sort them after you retrieve them from the Hashtable (e.g. by using java.util.Collections, java.util.Arrays).
--
Marco Schmidt

Q: What is better to use: array or vector?

Just wondering as I am using Vectors to store large amounts of objects from 50 to 4000 and each one has to be "looked at" every time paint is called...

Just wondering if it would be better to use an array, list etc?

Answer 1: Since the Vector method uses an array for storage but has extra steps involved in getting an element, use an array for fastest access.
--
WBB Java Cert mock exams http://www.lanw.com/java/javacert/

Answer 2: arrays are faster, vectors are more dynamic.
This should be evident just looking at the amount of code you need to traverse one versus the other. It might also be beneficial to write a linkedlist class and use that. That way you have a dynamic container which has potential to be faster than a vector (though still not as fast as an array). The problem with arrays is that if you need more space than the current size, you have to hardcode their copying into a bigger array. Conversely, if you never (or rarely) use the entire array, its a waste of space and memory.

The following are benchmark test results of vector vs. array (ran on a 200-Mhz Pentium w/ 96 Mbytes of memory and Windows95 ):

Allocating vector elements: 17910 milliseconds
Allocating array elements: 4220 milliseconds
Accessing Vector elements: 18130 milliseconds
Accessing array elements: 10110 milliseconds

One other reason for vectors being slower that I did not mention above is that vector methods are synchronized, which creates a performance bottleneck.

Hope this helps
--
MSW

Q: Would anyone know the performance issues regarding Vector's?

I am actually talking about resource pooling. I have objects that wait in a queue. It is a vector that keeps growing, as the queue gets bigger.

Do Vectors have much performance hit? Is there a better way to implement vectors to get the best out of them? Or am I better of creating a fixed size array?

Answer 1:
If you just want a LIFO or LILO queue, you may be better off with
LinkedList than with Vector, as then you'll never have to wait for the contents to be
copied.

Vectors perform pretty well, but if you know (even roughly) how big you're going to
need it to be, specifying that in the constructor call can help.

How sure are you that this will be a performance bottleneck? Premature optimisation
is the root of all evil...

The Vector class is thread-safe. By that I mean that there is no way to corrupt the
internal representation of the data by accessing the vector from more than one
thread. However, it is still possible, very easy in fact, to use a vector in a way that is
not thread safe.

Consider this code:

```
for (int i = 0; i < vector.size(); i++) {
System.out.println(vector.elementAt(i));
}
```

It looks safe, but there's a subtle flaw...

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

# General Java Questions - II

Q: I wonder what happened if I remove "deprecations" from my code, for example size() and put getSize().

Don't the programs work any more on older browsers (e.g. IE3)?

Answer: Check the docs to see whether they say "Since 1.1", "Since 1.2" or "Since 1.3" - if so, they will not work in the oldest MS VM.
Keep in mind that Sun's programmers haven't been very good at actually documenting this for all classes and methods.
or directly here:

To check your code against any version of the JRE (1.1, 1.2, 1.3), use JavaPureCheck: http://java.sun.com/100percent/
--
Marco

Q: I mainly use C++ but I have a program to write and java seems to be the right tool for this job. That said my java skills are a bit rusty. I downloaded a free IDE and am ready to go but I'm not a fan of putting my method implementations in the class definition.

Is there a way to separate the implementation from the class definition?
Like in c++ there are .h files and .cpp files.

Answer: You can use an interface and an implementation class, if you like.
The interface has just the method signatures; the class has their implementations.
You can then declare variables having the type of the interface, and assign objects of the implementation class created with "new", or by a factory method.

Java is very different from C++ in that the interface of a class is defined by a compiled view of that class, not by textual inclusion of source for a separate interface description. Therefore the concept of splitting interface from implementation does have the same meaning in Java as in C++.

In Java, you can separate the description of pure interfaces (or abstract classes) from concrete classes that implement (or complete) them. This is a useful technique - compare java.util.Collection and java.util.List to java.util.ArrayList and java.util.LinkedList.
Also, it is traditional to define the return type from factory methods and other such producers to be pure interfaces to allow the factory or other class to decide which implementation to give you - this is used extensively in java.net and java.sql packages.
--
Phil Hanna (Author of JSP: The Complete Reference
http://www.philhanna.com), Chuck

Q: How do we exchange data between Java and JavaScript and vice-versa?

Answer: Public variable and methods of Java Applet are visible to a HTML document. So using JavaScript you can access the public variables and public functions.

The syntax is:

var some_var = document.appletname.variable_name

With this you will get the value of the variable variable_name in your JavaScript variable some_var.

Q: Constructors and methods: are they the same?

I need a little help here...I have been teaching that constructors are not methods. This is for several reasons, but mainly because JLS says "constructors are not members" and members are "classes, interfaces, fields, and methods."
So, now the rest of the staff is ganging up on me and making life a little nasty. They quote Deitel and Deitel, and Core Java (which references "constructor methods") and who knows how many other books.
The one we are teaching in is loaded with so many errors that even though it calls constructors methods NOBODY will quote it as an authority.
How can so many people call constructors methods if they aren't.

Okay. Are they or aren't they? I holding to the definition that they are not unless it is so common to call them that, that I will have to change.
Comments?

Answer: If you go by the JLS (Java Language Specification) and the Java API (and you should) , then no, constructors are not methods. Consider that Class.getMethods() returns an array of Method instances and Class.getConstructors() returns an array of Constructor instances, and Constructor and Method or not interchangeable (one is not derived from the other), but both implement the Member interface. Seems to me that Java is going out of its way to differentiate them.

Besides, the mechanics of constructors are so different from the mechanics of methods, there seems to be no value to considering one a member of the set of the other.
Now, as far as teaching the language goes:

Methods:
+ return types
+ called by name
+ executed multiple times

Constructors:
+ super(...) or this(...) as first instructor (often implicit)
- everything else

I very much do not like trying to lump the two concepts together, especially in introductory courses. Conceptually they are very, very different things.
A constructor is a unique element (even if there are several). It has the name of the class, its declaration is different, and it doesn't have the same syntax as a method. It plays a unique role. You can call a method from a constructor, but you cannot call a constructor from a method.

I say constructors are distinct from methods, and for students, blurring the distinction will lead to problems.
--
by Chuck McCorvey, Chris Wolfe, Paul Lutus

Q: Simple question: why constructor doesn't work in following example?

```
class Start {
  public void Start() {
    System.out.println("Konstruktor START");
  }
}

public class Test {
  public static void main(String[] args) {
    Start s = new Start();
  }
}
```

Answer: Because you have included the return-type 'void' in the method declaration, it becomes a normal method, that just happens to have the same name as the class - so it won't get used as a constructor. Remove the 'void' and it should work.
--
Vince Bowdren

P.S. by John: If you do not specifically define any constructors, the compiler inserts an invisible zero parameter constructor "behind the scenes". Often this is of only theoretical importance, but the important qualification is that you only get a default zero parameter constructor if you do not create any of your own.
Your program used this zero parameter constructor and you saw nothing...

Q: Why we can not declare constructor as final?

Answer: The keyword final when dealing with methods means the method cannot be overridden.
Because constructors are never inherited and so will never have the oportunity to be overridden, final would have no meaning to a constructor.

Q: In Java, does exist a function like sprintf in C?

Answer: http://www.efd.lth.se/~d93hb/java/printf/index.html a free Java version of fprintf(), printf() and sprintf() - hb.format package

Q: If I declare an array of an objects, say Dogs, is that memory taken when I create the array or when I create the objects in the aray when I declare this array:

Dog[] dog = new Dog[100];

or does it take the memory when I actually create the Dogs in the array eg:

for(int i = 0;i<dog.length;i++)dog[i] = new Dog();

Answer: The statement above is actually two-fold. It is the declaration and initialisation of the array. Dog[] dog is the declaration, and all this does is declare a variable of type Dog[], currently pointing to null.
 You then initialise the array with new Dog[100], which will create 100 elements in the array, all of them referencing null.
It is important to realise that the elements of an array are not actually objects, they only reference objects which exist elsewhere in memory. When you actually create the Dog objects with new Dog(), these objects are created somewhere in memory and the elements in the array now point to these objects.
Pedant point:
Nothing ever points to null. It is a constant that represents the value of a reference variable that is not a pointer to some object  new Dog[100] creates an array of 100 null Dog references.

Q: How do I return more than one value using the return command?

Answer: You could make a new object/class that contains these two values and return it. For example:
Define an object like this:

```
class MyObj {
   public int myInt;
   public double myDouble;
}
```

Then, in your method create one of these, set the corresponding values, and return it.

```
MyObj yourMethod() {
   MyObj obj = new MyObj()
   obj.myInt = 20;
   obj.myDouble = 1.0003
   return obj;
}
```

Q: How do I use object serializtion for an object that has other objects as data member? Do both the class need to implement serialize?
How about static data?

```
class A{
}
class B{
public A a;
}
```

Answer: Both the object and all the object references it contains need to belong to classes that implement Serializable.
Static and transient fields are not serialized. For more, see,
http://java.sun.com/docs/books/tutorial/essential/io/serialization.html

Q: I recently learned a bit about "inner classes" but this seems to be different...

I'm a bit new to Java programming so bear with me. My employer bought a package of java graphics library programs to support some chart applets we want to create. We have the source code. I'm trying to create a jar file with all the files I need to run the applet. When I currently run the applet, the browser java tool says that it can't find "TextComponent$1.class". I recently learned a bit about "inner classes" but this seems to be different. The "TextComponent.java" file does contain some inner classes, but not a class called "1". I'm confused. Is this an inner class? Or is it something else. Any help would be appreciated. Thanks...

Answer: The TextComponent$1.class is the first anonymous class defined in TextComponent.java. Since nested (inner) classes are compiled to their own .class file, they needed unique names. The javac compiler is just creating a unique file name for an anonymous nested class.

Q: Hi there, does anybody know a good source of design patterns written in Java?

Answer: A pretty good (free to download) book.

http://www.patterndepot.com/put/8/JavaPatterns.htm

Q: Whats the difference between the two: System.err. and System.out? When should we use System.err?

Answer 1: System.out leads the output to the standard output stream (normally mapped to your console screen), System.err leads the output to the standard error stream (by default the console, too). the standard output should be used for regular program output, the standard error for errormessages. If you start your console program regularly both message types will appear on your screen.

But you may redirect both streams to different destinations (e.g. files), e.g. if you want to create an error log file where you don't want to be the regualr output in.

On an UNIX you may redirect the output as follows:

java yourprog.class >output.log 2>error.log

this causes your regular output (using System.out) to be stored in output.log and your error messages (using System.err) to be stored in error.log

Answer 2: System.err is a "special" pipe that usually is directed to the standard consolle. You can redirect the System.out with the normal pipe control (| or >), but System.err no. If you want to put both the "normal" output and the "error" output to a file you must use the special redirect 2>.

This allow you to send normal messages into a file or in the /null black hole, but still receive the error messages on the console.

Q: What is the essential difference between an abstract class and an interface? What dictates the choice of one over the other?

Answer: You can only extend one class (abstract or not) whereas you can always implement one or more interfaces. Interfaces are Java's way to support multiple inheritance.

Q: Does anyone know how could I get the size of an Enumeration object? The API for Enumeration only contains getNext() and next().

Answer 1: You can't. Theoretically, some classes that implement Enumeration may also provide some way to get a size, but you'd have to know about the more specific run-time type and cast to it... and none of the standard java.util Collections classes nor Vector or such provide these methods in their Enumeration implementations.

Answer 2:

you can make your own class like this:

```java
import java.util.*;

public class MyEnumeration{
  int size;
  int index = 0;
  Enumeration e;

  public MyEnumeration(Vector v){
    size = v.size();
    e = v.elements();
    index = 0;
  }

  public boolean hasMoreElements(){
    return e.hasMoreElements();
  }

  public Object nextElement(){
    index++;
    return e.nextElement();
  }

  public int size(){
    return size;
  }
```

```
    public int getIndex(){
      return index;
    }
}
```
--
by Nicolas Delbing and Victor Vishnyakov

📰 Q: Is there a way to provide values for a Vector in the source code, analogous to array initializers?

Answer: The Vector class constuctors take no arguments other than Collection (since JDK 1.2), which is abstract, and since a Vector is a structure whose size can change dynamically, it's contents can only be initialaized through member methods.
--
Mike Lundy

📰 Q: How would I add a help file to a java application?

Would it have to be platform specific, or is there a Java api for making help files?
If so, what is it?

Answer: See JavaHelp at http://www.javasoft.com/products/javahelp/
 you create HTML pages for the main text, and add some XML files for a hierarchical table of contents and a map from TOC tags to relative URLs giving document locations.

📰 Q: What is a Just-In-Time(JIT) compiler?

Answer: It is a JVM that compiles Java instructions (called bytecode) into native machine instructions at run time and then uses this compiled native code when the corresponding Java code is needed. This eliminates the constant overhead of interpretation which tradition first generation JVM's used.
--
Dave Lee

📰 Q: Ok, I know that one cannot put primitive types into a hashmap (only objects or references to them) and I know how to deal with that (write some kind of wrapper class).

What I'm interested in is: 'Why is that?' Why can I not put a primitive type into a hashmap? Something to do with this 'heap' thing… Right?

Answer: HashMap requires a key or a value to be assignment- compatible with java.lang.Object (i.e. to be an object or an array). Primitive types aren't ones.
You can use wrappers like java.lang.Integer for this purpose. All container classes require some basic operations to be defined for all of its contained objects in order for it to organize the data.

For example, a hashmap requires a hashing method hashCode(). A primitive type doesn't have any methods associated with it.
The distinction between primitive types (int) and their wrapper classes (Integer) is made purely for optimisation purposes. For example, we wouldn't want to have to instantiate actual objects in an array instead of just allocating a block of memory for an array of integers.
--

Gary

📋 Q: Is there a collection object like the hashmap or hashtable that stores values in an ordered path? Vector does this but i need the key/value functionality. hashmaps do not guarantee the order of the objects.

Answer: Take a look at java.util.TreeMap.
Red-Black tree based implementation of the SortedMap interface. This class guarantees that the map will be in ascending key order, sorted according to the natural order for the key's class (see Comparable), or by the comparator provided at creation time, depending on which constructor is used.

Note that this implementation is not synchronized. If multiple threads access a map concurrently, and at least one of the threads modifies the map structurally, it must be synchronized externally.

📋 Q: Most people asked why there is an error, but my question is why this is NOT an error

Please take a look:

r is a number and s is a character, why can I put them together to make a comparison without compilation error? Could somebody tell me... thank you

```
double r = 34.5;
char s = 'c';
if (r > s) {
   System.out.println("r > s");
} else {
   System.out.println("r < s");
}
```

Answer 1: Yes, char is indeed a 16-bit value. However, the actual answer is in the Java
Language Specification, section 5.6.2, which is at the following URL:
http://java.sun.com/docs/books/jls/second_edition/html/conversions.doc.html#170983

In summary, the char is automagically promoted to a double. No explicit cast is necessary since the language rules say that it gets "promoted" to a double
by John O'Conner

📋 Q: == and equals ()... These two still make me confuse a lot of time.

Can somebody give me some thumb rule or explain it to me?

Answer: When you use == with a primitive -int, double, char, ... you are checking that the values are identical. But if you use == with an object, you are checking that the 2 objects are stored at the same address. In other words the references pointing to the same object...

Method equals () is different.

It is the same as ==, if it isn't overriden by the object class.

Many classes override the method equals (). In this case this method will check that content of the object is the same or not, not addresses.

📰 Q: Why do I get message like "wrong magic number" when I am trying to run applet? What is magic number?

Answer: The first thing a JVM does when it loads a class is check that the first four bytes are (in hex) CA FE BA BE. This is the "magic number" and thats why you are getting that error, you are trying to load a file that isnt a class and so the class loader in the JVM is throwing out that exception.

Make sure you transfer the class files to site in binary mode, rather than text or ASCII mode.
An error from the browser saying "cannot start applet ... bad magic number" usually means that one of the class files on the server is corrupted. '
Replace your class binary files on the web server; clean up the cache of your browser, and reload your applet.

📰 Q: In java, I found a lot of methods, which let you enter a line (read (), readLine () e.c.t). They all wait until return is pressed, and then start providing you the information.

Does anyone know if there is a read method available whit the desired behaviour, i.e. which doesn't wait for return being pressed?

Answer: Java does not provide it, the terminal itself waits until return is pressed before sending the entered line to Java.

You need to use some platform specific mechanism to change the terminal settings.

📰 Q: Can I pass a function as a parameter in Java? If so, how?

Answer: No, you cannot. But you can pass an object with method and then just use this method like this:

myFunction (object); // object has method useMyMethod(){ do smth here...}
..
..
somewhere in another place where use this function..
object.useMyMethod();

📰 Q: What is difference capacity() and size() methods for vector?

 What is difference between

public final int capacity()
Returns the current capacity of this vector.

and

public final int size()
Returns the number of components in this vector.

Answer: please read this method: ensureCapacity
public final synchronized void ensureCapacity(int minCapacity)

Increases the capacity of this vector, if necessary, to ensure that it can hold at least

the number of components specified by the minimum capacity argument.
Parameters:
minCapacity - the desired minimum capacity.

📋 Q: What're the differences between classpath and import in the java application?

Do I need to have the package in the classpath first before importing in a java application or need not?

Answer: Classpath is an environment variable of your OS, you have to set it (or better: Java sets it) to tell Java where to search for classes.
You use import in a program to let Java search for the specified classes within the classpath. This implies that the package must be in the classpath.
--
Stephan Effelsberg

📋 Q: What is difference between Iterator and Enumeration?

First of all Java FAQ Team wish you !!!HAPPY NEW YEAR!!! and then

Answer: from
http://java.sun.com/docs/books/tutorial/collections/interfaces/collection.html

The object returned by the iterator method deserves special mention. It is an Iterator, which is very similar to an Enumeration, but differs in two respects:
Iterator allows the caller to remove elements from the underlying collection during the iteration with well-defined semantics.
Method names have been improved.
The first point is important: There was no safe way to remove elements from a collection while traversing it with an Enumeration. The semantics of this operation were ill-defined, and differed from implementation to implementation.
The Iterator interface is shown below:

```
public interface Iterator {
   boolean hasNext();
   Object next();
   void remove(); // Optional
}
```

The hasNext method is identical in function to Enumeration.hasMoreElements, and the next method is identical in function to Enumeration.nextElement. The remove method removes from the underlying Collection the last element that was returned by next. The remove method may be called only once per call to next, and throws an exception if this condition is violated. Note that Iterator.remove is the only safe way to modify a collection during iteration; the behavior is unspecified if the underlying collection is modified in any other way while the iteration is in progress.
The following snippet shows you how to use an Iterator to filter a Collection, that is, to traverse the collection, removing every element that does not satisfy some condition:

```
static void filter(Collection c) {
   for (Iterator i = c.iterator(); i.hasNext(); )
      if (!cond(i.next()))
      i.remove();
}
```

Two things should be kept in mind when looking at this simple piece of code:
The code is polymorphic: it works for any Collection that supports element removal, regardless of implementation. That's how easy it is to write a polymorphic algorithm under the collections framework!
It would have been impossible to write this using Enumeration instead of Iterator, because there's no safe way to remove an element from a collection while traversing it with an Enumeration.

**Q: How can I find the first dimension length of the 2-dimenstions array? I have use the array[].length but it does not work, how can I solve this problem?**

Answer: Java doesn't really have "multidimensional arrays", only arrays of arrays. So try: array[0].length and you will get this dimension.

**Q: I guess what I'm asking is "Is java.util.Hashtable thread safe?"**

It's been a while since I've used hashtables for anything significant, but I seem to recall the get() and put() methods being synchronized.
The JavaDocs don't reflect this. They simply say that the class Hashtable is synchronized. What can I assume? If several threads access the hashtable at the same time (assuming they are not modifying the same entry), the operations will succeed, right? I guess what I'm asking is "Is java.util.Hashtable thread safe?"

Answer: That is right! It is recommendable, if you have questions like these, always look at source for the API, it's freely available.

**Q: I try to copy an object of my own using the clone() method from java.lang.Object, but this is a protected method so I can't use it. Is there some other way to get my objective of duplicating an arbitrary object?**

Answer: If you want to clone your object, you need to make it cloneable. To achieve this, you need to do two things:

1. implement the interface Cloneable

2. override the method clone(), so that it
a. becomes public
b. calls super.clone()
c. if necessary, clones any members, or
d. if a member can't be cloned, creates a new instance.


Simple example:

```
public MyClass implements Cloneable {
  int someNumber;
  String someString;

  public Object clone() {
  // primitives and Strings are no
  // problem
    return super.clone();
  }
```

```
}
```

In this case the method clone() of the class MyClass returns a new instance of MyClass, where all members have exactly the same value. That means, the object reference 'someString' points to the same object. This is called a shallow copy. In many cases this is no problem. Strings are immutable and you do not need a new copy. But if you need new copies of members, you have to do it in the clone() method. Here is another simple example:

```
public class SomeMember implements Cloneable {
  long someLong;

  public Object clone() {
    return super.clone();
  }
}

public AnotherClass extends MyClass {
   SomeMember someMember;

   public Object clone() {
      AnotherClass ac = (AnotherClass)(super.clone());
      if (someMember != null) {
         ac.someMember = (SomeMember)(someMember.clone());
      }
      return ac;
   }
}
```

Note that the class AnotherClass, that extends MyClass, automatically becomes Cloneable, because MyClass is Cloneable.

Also note, that super.clone() always returns an Object of the type of the actual object, although the superclass doesn't know anything about that sub class. The reason is, that Object.clone() is a native method, which just allocates new memory for the new object and copies the bytes to that memory. Native code has it's own ways of finding out which type to return ;-)
--
Karl Schmidt

Q: I was just wondering about the usefulness of Interfaces...

I was just wondering about the usefulness of Interfaces. I was under the impression that interfaces could be used to perform multiple inheritance. But an interface only declares a method - in a very abstract way.

A class that implements an interface needs to define its own implementation of a certain method. What is the use of having an interface when nothing is being gained...?

Answer: If two classes implements the same interface, you can get a reference to the interface instead of the effective class without bother what class are you managing.

This is very useful in RMI (for example) or in any condition when you have to take an object without knowing exactly his class, but only the interface that it implement.

For example:
public void recurseList( List l )

the generic List ensure that you can use every List for this method (ArrayList, AbstractList, Vector...), so your calling method can be:

ArrayList l = new ArrayList(); or
Vector l = new Vector();

recurseList( l );

Without any problem.

by Davide Bianchi

Q: I got a problem with an array/vector...

I got a problem with an array/vector.

my class contains a member:
static Vector quad[][];
....

in my constructor I got:
Vector quad[][] = new Vector[row][col];
for (int i = 0; i < row; i++){
   for (int j = 0; j < col; j++){
      quad[i][j] = new Vector (0,1);
   }
}

// row and col are int between (10..50) -- it's a big I know, but that
might not be the problem

My PROBLEM (and I don't know what to do, really), I can't access quad[x][y] outside of the constructor!!!! Within the constructor I've got full access on quad[x][x]. Java (1.2) returns a NullPointerException on any method except within the constructor!!!

What's my fault!???

Answer: I guess you shouldn't write Vector here:
Vector quad[][] = new Vector[row][col];
so, the correct variant may be:
quad[][] = new Vector[row][col];

I guess You just overridden your static variable with one defined in your constructor: Vector quad[][].
Thus, you're initializing NOT your class-scope static variable but your constructor-scope quad. It's not reachable outside the constructor. And as for static quad, it has never been initialized! And a first reference to it causes

NullPointerException. I guess. I hope I'm right :)

--

Xao Rom

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

- [ ] IT Professional
- [ ] General Computer
- [ ] Design and Graphics
- [ ] Linux
- [ ] Freeware/Shareware
- [ ] Games
- [ ] Intranet
- [ ] Computer Hardware
- [ ] Web Design
- [ ] Computer Software
- [ ] UNIX
- [ ] Web Software
- [ ] Windows Software
- [ ] Personal Finance
- [ ] Programming Software
- [ ] Servers
- [ ] C/C++
- [ ] Powerbuilder
- [ ] Perl
- [ ] XML
- [ ] Java
- [ ] SQL
- [ ] Design/Graphics
- [ ] Utilities
- [ ] Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

# General Java Questions - III

## Q: I propose that Java should allow multiple inheritance if...

Everyone knows the potential problem with multiple inheritance is when you run into the problem of having two instances of a grand parent super class.

For example:
class A extends D {int i; }
class B extends D {int i; }
class C extends A,B {}

Potentially, you could have two copies of D for each instance of C.

However, I propose that Java should allow multiple inheritance if there are no instance variables associated with the abstracts that the base class is extending.

abstract class A { public setX(); public setY(); public setAll() {setX (); setY(); }
abstract class B { public setC(); public setD(); public setBoth()
{setC(); setD(); }

class C extends A,B {}

You won't have two instances of some grandfather class, since A and B doesn't have instances variables.

I hope the next versions of Java explores this issue.

Answer: It does. They're called interfaces:

interface A { public void setX(); public void setY(); public void setAll(); }
interface B { public void setC(); public void setD(); public void setBoth(); }

interface C extends A,B {};

```
public abstract class D implements C {
}
```

--
jim

**Q: Is there a way to know from class X which class called the method foo()?**

If class A and class B are calling a method foo() on class X, is there a way to know from class X which class called the method foo() (they can be either A or B). I know that this can be done by capturing the stack trace and examining it, but that solution looks expensive as I have to create a new Throwable object every time and capture stack trace (And I do this quite frequently).

Is there any other elegant solution to do this, any help and direction is appreciated.

Answer: Pass a reference to the class to the foo() method.
```
foo(Object x){
    System.out.println(x.getClass());
}
```

should do it.

**Q: Why does this simple application never exit?**

```
public class UIQuitTest {
   public static void main (String[] args) {
      java.awt.Frame f = new java.awt.Frame();
      f.dispose();
      f = null;
   } // end of main ()
}
```

The application above never quits, is it a bug or a (mis)feature? Win98, JRE 1.3.0

Answer: By creating an AWT object, you now have started the AWT thread. In order to end the application now, you have to do a System.exit(0) that will kill all non-daemon threads, including the AWT thread.

**Q: Is it possible to stop an object from being created during construction?**

For example if an error occurs inside the constructor (e.g. the parameters pass in were invalid) and I wanted to stop an object being created would it be possible to return null rather than a reference to a new object. (I know the term return is technically correct in this case but you know what I mean).
Basically, is it possible to cancel object creation?

Answer: Yes, have the constructor throw an exception. Formally, an object _will_ be created (since the constructor is a method invoked after the actual method creation), but nothing useful will be returned to the program, and the dead object will be later reclaimed by Garbage Collector.

But the clean way is as another reply suggests, that you leave calls to the

constructor to a static factory method which can check the parameters and return null when needed.

Note that a constructor - or any method in general - throwing an exception will not "return null", but will leave the "assign target" as it was.
Tor Iver Wilhelmsen

**Q: What does mean "volatile"?**

For the past couple of hours, I've seen quite a few set of codes that has the _volatile_ keyword.

E.g.

private volatile somethingsomething....

What does it mean?

Answer: See JLS 2nd Edition, which just came out last year, still mentions it.

http://java.sun.com/docs/books/jls/second_edition/html/classes.doc.html#36930

"A field may be declared volatile, in which case a thread must reconcile its working copy of the field with the master copy every time it accesses the variable. Moreover, operations on the master copies of one or more volatile variables on behalf of a thread are performed by the main memory in exactly the order that the thread requested."

Seems just like the idea in C++ and appears still to me present in the language.

Synchronization certainly has it place in many applications, that doesn't mean that volatile is not longer used or part of the language.
--
Paul Hill

**Q: suppose I put a file a.txt in package com.xyz and the try access it like following. Will it work?**

```
import com.xyz.*;
public class Hello{
   File f = new File("a.txt");
   ...
}
```

it is not working for me. Is there any workaround?

Answer: If the source and the text file are in the jar file, then you access the file by:

URL fileURL = getClass().getResource("file.txt");

You can then read the file by using a reader (or whatever you choose),
e.g.:

_istream = new BufferedReader( new

InputStreamReader(fileURL.openStream()) );
--
j o h n e w e b e r

Or, simpler
getClass().getResourcesAsStream("file.txt"), but you must be sure that
file is in the same directory ( package ) as your class, otherwise you
need play with getClassLoader().getResourceAsStream( <path>"/file.txt" );
--
Oleg

**Q: Difference between loading and instantiating a class???**

Well, the subject says it all. What is the difference between loading and instantiating
a class in a JVM.

**Second question:** What would happen if at runtime I update a class file? Will the JVM know
to use that instead?

Answer: The difference is that when a class is loaded by a ClassLoader it is read in
as a stream of bytes, presumably from a file, but it could just as easily be from over
the network, and then processed or "cooked" into a representation that the VM can
use to make instances of Objects of that classes type. This last part is the
instantiation. You can load a class at runtime with:

Class.forName( "MyClass" );

and instantiate one with:

MyClass mc = Class.forName( "MyClass" ).newInstance();

Cool, ehh. You don't have to know the name of a class at compile time.

>Second question: What would happen if at runtime I update a class file?
>Will the JVM know to use that instead?

Loaded classes are cached because it's quite costly to do the "cooking" I mentioned
above. So it will not be loaded. You may create a separate ClassLoader with new
SecureClassLoader but that will cause all classes _it_ loads to be loaded from this
new ClassLoader but that's not what
you want.

I don't know if you can specify that a class should be loaded from disk again using
the normal ClassLoader. You could very easily make your own ClassLoader in which
case you would have explicit control over such things. Look at java.lang.ClassLoader
and java.lang.Class.
--
Michael B. Allen

📑 Q: Why developers should not write programs that call 'sun' packages?

Answer: Java Software supports into the future only classes in java.* packages, not sun.* packages. In general, API in sun.* is subject to change at any time without notice.
A Java program that directly calls into sun.* packages is not guaranteed to work on all Java-compatible platforms. In fact, such a program is not guaranteed to work even in future versions on the same platform.

For these reasons, there is no documentation available for the sun.* classes. Platform-independence is one of the great advantages of developing in the Java programming language. Furthermore, Sun and our licensees of Java technology are committed to maintaining backward compatibility of the APIs for future versions of the Java platform. (Except for code that relies on serious bugs that we later fix.) This means that once your program is written, the class files will work in future releases.

For more details, see the article Why Developers Should Not Write Programs That Call 'sun' Packages.
http://java.sun.com/products/jdk/faq/faq-sun-packages.html

📑 Q: Can garbage collector remove my singleton?

A usually singleton..

```
public class Single{
   private static Single single;
   private Single {}
   public static Single getInstance(){
     if(single==null){
        single = new Single();
     }
     return single;
  }
}
```

Well,, seems good ?
But classes are objects too...so do Java 2 v1.3 class garbagecollecting? Meaning my singleton could dissapear if i dont keep a refrence to it (or the class itself) somewhere ?

If classes is not garbagecollected, that's pretty stupid, I dont want classes taking up memory when i perhaps never will use it again....

Answer: No. Classes can define objects. That is, only the dynamic part of the class defines objects. The static part exists only in one place in memory and can not be duplicated. You can call the getInstance() method from anywhere in your program. Java requires however that you tell where to find the method, in this case in the Single class.
Therefore, you should use

Single.getInstance()

to get the instance. This is (though it looks much like it) not an execution of a method

on an object, but just a method call without object. Single is only used to find out which getInstance() method should be used, and where it is.

You could add a delete() method if you don't need the instance anymore:

```
public class Single{
   private static Single single;
   private Single {}
   public static Single getInstance(){
     if(single==null)
        single = new Single();
     return single;
    }

   public static delete(){
     single = null;
    }
}
```

The garbage collector can now remove the single object after delete() is called if memory is needed.
--
Rijk-Jan van Haaften
Dept of Computer Science, Utrecht University, The Netherlands

P.S by John:
Doing more deeper investigation of this question I found one very good article about this topic. Everybody who are interested can read full article here:
http://developer.java.sun.com/developer/technicalArticles/Programming/singletons/

For the rest of our audience shortly:
A Singleton class can be garbage collected and when
".. a Singleton class is garbage-collected and then reloaded, a new Singleton instance is created. Any class can be garbage-collected when no other object holds reference to the class or its instances. If no object holds a reference to the ' Singleton object, then the Singleton class may disappear, later to be reloaded when the Singleton is again needed. In that case, a new Singleton object will be created. Any static or instance
fields saved for the object will be lost and reinitialized.

This problems exists in older JavaTM Virtual Machines1. JDK 1.2 VMs, in particular, conform to a newer class garbage collection model that forbids any class in a given classloader to be collected until all are unreferenced"

And you "... can also set your VM to have no class garbage collection (-Xnoclassgc on the JRE 1.3, or -noclassgc on the IBM JVM). Keep in mind that if you have a long-running program that frequently reloads classes (perhaps through special class loaders such as the remote class loaders), you have to consider whether that could cause a problematic buildup of garbage classes in the VM."

Also some people asked what is a Singleton and what is relation has it to Java. Shortly a Singleton is one of classical design patterns that are used in software

development. More please read in free book here:
http://www.patterndepot.com/put/8/JavaPatterns.htm

Q: I study patterns now and would be lost and forget everything very soon! Help!

I am very new to design patterns and just bought the GoF book to learn about it.
But as I complete the first couple of chapters right away, I see that they have 23
different patterns and I would be lost and forget everything very soon if I sequentially
(page by page) read the entire book!.

Do any of you have recommendations on how to proceed with this book so that I 'll
remember at least some of them by the time I am done with the book? I can see that
many of the classes in java API use composition, facade etc...
But other than that I don't think I 'll be able to gather anything unless I am able to
remember where & when to use particular pattern A, B or C...

Answer: Glad to hear you got the GoF book, it's a great reference manual for
patterns. As you've found, however, it's a bit heavy to just "read." What I recommend
to people is that they pick a few of the easier, more commonly used patterns:
Singleton is a no-brainer that pops up a lot. Adapter tends to get used here and there
and isn't that difficult to understand. If you're doing Swing, then definitely go learn the
Observer pattern. It'll help to keep you from mixing data and interface code. Once
you've learned three or four and have used them a few times, then as you start new
projects, look back to the text to see if there are opportunities in your project where
other patterns can be used.

You'll find that over time you'll use more and more of the patterns (some a lot more
than others, obviously). I've often found cases where I missed a pattern during
design and had "the light go on" after I'd written a bunch of code and realized I was
either using a known pattern by accident, or could have used a pattern to my
advantage. When possible, I then go back and adjust the design/code to match the
pattern.

Keep in mind that the patterns generally don't appear as "absolute." It's expected
that you may have variations to satisfy your application's needs.
It's really helpful to others, however, if you make a quick note in your design
doc/code about what pattern you were using (which helps them learn patterns too,
and helps them understand what you were up to if they know the pattern already).
--
Rico Trooper

Q: When I used java.util.Hashtable, I also used interface Enumeration. But I
found that this interface contains defined methods.

How can an interface contain defined methods? Isn't an interface supposed to hold
abstract methods only?

Answer:
> How can an interface contain defined methods?
No.
> Isn't an interface supposed to hold abstract methods only?
Yes. (and fields). I'm not sure why you think Enumeration contains defined methods,
since it does not (I checked the source to be sure). Keep in mind that an
implementing class is also of type Enumeration. So Hashtable's keys method which

returns an Enumeration will actually be returning an inner class that implements Enumeration
(and therefore defining hasMoreElements and nextElement in a manner that is specific to traversing a Hashtable).

If this doesn't help, maybe you could clear up what you mean by Enumeration having defined methods.
--
Eric

Q: can anyone provide an example on how to use clone() and clonable interface?

Answer:

```
//Class Point uses default clone implementation that returns
//shallow copy of Point (which is sufficient in this case).
class Point implements Cloneable {
    int x, y;
}
class ImmutableRectangle {
   private Point lowerLeft = new Point();
   private Point upperRight = new Point();
   public Point getLowerLeftPoint() {
     try {
       return lowerLeft.clone();
     }
     catch(CloneNotSupportedException e) {
        //shouldn't occur since Point is cloneable
        throw new InternalError();
     }
   }
}
```

we don't want Rectangle points to be changed outside its class. the best way would be to create a copy of Point object and pass it.

i5mast
MVP for Java 2
http://www.brainbench.com

☐ IT Professional  ☐ General Computer  ☐ Design and Graphics  ☐ Linux  ☐ Freeware/Shareware

☐ Games  Intranet  Computer Hardware  Web Design  ☐ Computer Software

☐ UNIX  ☐ Web Software  ☐ Windows Software  ☐ Personal Finance  ☐ Programming Software

☐ Servers  ☐ C/C++  ☐ Powerbuilder  ☐ Perl  ☐ XML

☐ Java  ☐ SQL  ☐ Design/Graphics  ☐ Utilities  ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

# General Java Questions - IV

Q: I am looking for more efficient compression APIs esp. for *.doc and *.html files.

I need some Java APIs wherein I can implement various compression algorithms. I have already the GZIP compression technique shipped with JDK's java.util.zip package. I am looking for more efficient compression APIs esp. for *.doc and *.html files.

Answer: You often get better compression results for a large number of smaller files by concatenating the uncompressed files and then compressing the result. That's why tar.gz is often superior to zip for large numbers of html files (like API docs).

You will not get much better general-purpose compression than Zip, at least not until you have a specific class of data and know very much about its structure. Even then you'll need some experience and time working on a better compression method.

You might want to take a look at this Java implementation of bzip2:
http://www.aftexsw.com/aftex/products/java/bzip/
It uses more CPU cycles and memory but typically compresses better than zip.
--
Marco

Q: I need to programmatically replace an entry in a zip file.

I could not quite get it using the ZipOutputStream because it simply creates a new file and write only that entry for me. The rest of the original entries are gone. Does anyone have a solution for this?

Answer:

1) Read the file (myfile.properties) out of the original Zip Archive (original.zip)
2) Make your changes, write the file to the file system
3) Create a New Archive (originalNew.zip)

4) Write your edited file (myfile.properties) to originalNew.zip
5) loop through all the entries in the original Zip archive (original.zip), adding them to the new archive (originalNew.zip) EXCEPT for the file you are replacing (myfile.properties)
6) When you're done, erase the original and rename your new one to original.zip.

I believe that this may be the only way to do this, since there doesn't seem to be any random access in the ZIP file.
--
Kevin T. Smith

📘 Q: About the order of my elements in hastable...

I save some data into hashtable.
    For example,

hashtable.put ( "1", "one" );
hashtable.put ( "2", "two" );
hashtable.put ("3" , "three" );

when I get back the element from hashtable using Enumeration class ,
Enumeraton e = hashtable.keys();
while ( e.hasMoreElements() ){
  Object k = e.nextElement();
  Object v = hashtable.get (k );
}

the default result is
"3","three"
"2", "two"
"1", "one"

here I want to get the data sorted as ascending. ( the order as I insert ) such as
"1", "one"
"2", "two"
"3", "three"

or is it possible to get the data from end of hashtable?

Answer: When you insert elements into a Hashtable, they will not be stored in the order you insert them. They are stored in a way that makes it easy to find them by the key you specify.

So, you must either use another data structure (e.g. Vector) or sort them after you retrieve them from the Hashtable (e.g. by using java.util.Collections, java.util.Arrays).
--
Marco Schmidt

📘 Q: What is better to use: array or vector?

Just wondering as I am using Vectors to store large amounts of objects from 50 to 4000 and each one has to be "looked at" every time paint is called...

Just wondering if it would be better to use an array, list etc?

Answer 1: Since the Vector method uses an array for storage but has extra steps involved in getting an element, use an array for fastest access.
--
WBB Java Cert mock exams http://www.lanw.com/java/javacert/

Answer 2: arrays are faster, vectors are more dynamic.
This should be evident just looking at the amount of code you need to traverse one versus the other. It might also be beneficial to write a linkedlist class and use that. That way you have a dynamic container which has potential to be faster than a vector (though still not as fast as an array). The problem with arrays is that if you need more space than the current size, you have to hardcode their copying into a bigger array.
Conversely, if you never (or rarely) use the entire array, its a waste of space and memory.

The following are benchmark test results of vector vs. array (ran on a 200-Mhz Pentium w/ 96 Mbytes of memory and Windows95 ):

Allocating vector elements: 17910 milliseconds
Allocating array elements: 4220 milliseconds
Accessing Vector elements: 18130 milliseconds
Accessing array elements: 10110 milliseconds

One other reason for vectors being slower that I did not mention above is that vector methods are synchronized, which creates a performance bottleneck.

Hope this helps
--
MSW

Q: Would anyone know the performance issues regarding Vector's?

I am actually talking about resource pooling. I have objects that wait in a queue. It is a vector that keeps growing, as the queue gets bigger.

Do Vectors have much performance hit? Is there a better way to implement vectors to get the best out of them? Or am I better of creating a fixed size array?

Answer 1:
If you just want a LIFO or LILO queue, you may be better off with LinkedList than with Vector, as then you'll never have to wait for the contents to be copied.

Vectors perform pretty well, but if you know (even roughly) how big you're going to need it to be, specifying that in the constructor call can help.

How sure are you that this will be a performance bottleneck? Premature optimisation is the root of all evil...

The Vector class is thread-safe. By that I mean that there is no way to corrupt the internal representation of the data by accessing the vector from more than one thread. However, it is still possible, very easy in fact, to use a vector in a way that is not thread safe.

Consider this code:

```
for (int i = 0; i < vector.size(); i++) {
System.out.println(vector.elementAt(i));
}
```

It looks safe, but there's a subtle flaw...

Q: Isn't the Vector class included in Java 1?

I'm writing an applet where I need to use something like Arraylist or Vector. Is there anything in Java1 I can use? When I try using the Vector Explorer gives me the message:
java.lang.NoSuchMEthodError: java/util/Vector: method add (Ljava/lang/Object;)Z not found

Answer: Before Java2 there was only the method addElement(Object). The method add(Object) was included with the List interface.
I suggest you downloaded the JDK1.1.8 documentation and referred to that (to decide which methods to use).
To avoid such problems in a future please check the 1.1 Vector doc online at:
http://java.sun.com/products/jdk/1.1/docs/api/java.util.Vector.html or, better still, install the 1.1 docs (download from http://java.sun.com/products/jdk/1.1/docs.html).

Q: How do I copy one array to another?

Given that I have an byte array defined like this:
byte byteSmall = new byte[23];
and another larger byte array defined like this:
byte byteBig = new byte[30];

How do I copy byteSmall into byteBig starting at index 7 without a for loop like this:

```
for(int i = 0; i < 23; i++){
   byteBig[i + 7] = byteSmall;
}
?
```

Answer: See System.arraycopy:
"Copies an array from the specified source array, beginning at the specified position, to the specified position of the destination array. A subsequence of array components are copied from the source array referenced by src to the destination array referenced by dst. The number of components copied is equal to the length argument. The components at positions srcOffset through srcOffset+length-1 in the source array are copied into positions dstOffset through dstOffset+length-1, respectively, of the destination array.

If the src and dst arguments refer to the same array object, then the copying is performed as if the components at positions srcOffset through srcOffset+length-1 were first copied to a temporary array with length components and then the contents of the temporary array were copied into positions dstOffset through dstOffset+length-1 of the argument array."

📋 Q: More about Robot! I met with a problem in using class Robot.mousePress...

The compiling process is successful. But when I run it, I receive
"IllegalArgumentException:
Invalid combination of button flags". I don't quit understand this information. Part of
my code is as following:

Robot rMouse=new Robot();
int button=1;
rMouse.mousePress(button);
rMouse.mouseRelease(button);

I am really confused. Will you please give me some advice? Thank you in advance!

Answer: You are not using a valid value for the argument to the mousePress() and
mouseRelease() methods. If you check the API documentation, you'll find the valid
values are a combination of one or more of the following constants:

InputEvent.BUTTON1_MASK
InputEvent.BUTTON2_MASK
InputEvent.BUTTON3_MASK

plus others which represent the Ctrl, Alt, and Shift keys. To press the left
mouse button, you want to use:

rMouse.mousePress(InputEvent.BUTTON1_MASK);
--
Lee Weiner

📋 Q: In what situation an exception has to be caught otherwise the compiler will
complain?

e.g. IOException does NOT have to be explicitly caught, however, SQLException
has to be caught otherwise VisalAge will not compile the program.

Answer: The only unchecked exceptions in Java are RuntimeException and its
subclasses. This includes such familiar classes as NullPointerException,
ClassCastException, and IndexOutOfBoundsException.

IOException is not one of these, and *does* have to be explicitly caught or thrown
--
jeff_robertson

📋 Q: Is it possible to use switch with range of values?

Ex:
switch (iBegins){
   case 2 to 10:
}
Answer:
not exactly like this but:
switch (iBegins){
   case 2:
   case 3:

```
    case 4:
    case 5:
    case 6:
    case 7:
    case 8:
    case 9:
    case 10:
        /* do stuff */
        break;
    case 11:
        /* other stuff */

}
```
Jan Schaumann <http://www.netmeister.org>

📑 Q: Is there a general reason why nullpointer exceptions occur?

Answer: Of course there is. A NullPointerException happens when you have a reference, it's set to null (or is a class or instance variable and has never been initialized), and you try to perform some operation on the object it points to. Since it really doesn't point to an object, you can't do that.
--
Chris Smith

📑 Q: I am aware that the exceptions in the catch statements must be of the same type or a super type of the exception thrown in the try block.
My question is this:
Is there any significance in the ordering of the catch statements after a try block?

Answer: It is required to design you catch statement sequences with most derived exception class type first and progressing to the highest superclass. Else, the code will not compile.
For a obvious reason, if the a catch block for a given class precedes a catch block for a type that is derived from the first, the second catch block will never be executed.
--
D. Michael Nelson

📑 Q: I wrote a program that use a few RS232 ports. The operators are unskilled and often start multiple instances of the program. Will someone please be so kind and tell me how I can prevent them doing it?

Answer 1: The first instance might write a file. Subsequent instances could check for the existence of that file, or else check it's contents.

Another method could involve creating a server socket on a specific port. Subsequent efforts to create a socket on that same port would throw an exception.

Answer 2: Actually a better way is to (on launch):

1) Check if the file exists. If not, create it, open it and run. Leave it open until you quit, upon which time you close it.

2) If the file _does_ exist, try to delete it. If the delete fails, then someone else has it open, which means another copy of your app is running. Inform the user and quit.

3) If you succeeded in deleting it, then you are the first. Now create, open and run.

Doing the above prevents the problem of having the semaphore file left around when the system crashes. I implemented it recently in one of our apps, and it works like a charm.
--
Burt Johnson
MindStorm Productions, Inc.
http://www.mindstorm-inc.com

Q: Can you call a class static abstract method from an abstract class or does it need to be extended and then called from its concrete class?

I've been told that abstract classes do not actually have any code in them cause they are like a placeholder, so I guess you wouldn't bother calling a static method in an abstract class cause it wouldn't have any code to begin with....?

Answer: You have been told wrong. Abstract classes can and do have code in them. See, for example, java.awt.Component, an abstract class with a lot of code and no abstract methods at all. If a class has any abstract method member, directly declared or inherited, it is required to be declared abstract. If not, it is the programmer's decision and should be based on whether it would make sense to have an instance of that class.

Perhaps whoever told you was confusing abstract classes with interfaces, which don't contain implementation, only abstract method and constant declarations.

You cannot declare a method to be both static and abstract. Abstract requires the method to be overridden before you can have a concrete class, static prevents overriding. You can have a static method in an abstract class - such a method could be called without creating an instance of the class, the only thing that is prohibited for abstract classes.
And when a subclass of an abstract method has been instantiated, all the methods from the original class will keep the same code in the instance. Most of the time an abstract class will have abstract methods.
However, there are several examples of abstract classes that don't have any abstract methods at all. Some examples are Component and FontMetrics from the AWT. It doesn't make sense to have just a Component that's not a specific type of component. It doesn't make sense to have a FontMetrics that doesn't measure any specific kind of Font.

Also being abstract never prevents overriding, it just requires overriding in order to derive a non-abstract subclass. And if a class is a subclass of an abstract class, it only MUST override those methods declared abstract. The other methods do not require overriding.

Q: I write java about 2 years, but I still confuse one thing that is why should we use interface???

If I need to implement an interface and just use its every methods name.
Why shouldn't just write every methods statments in a class, not in interface??
I only can think about that if I extend a class, than can implement another or the

others interface.
As you can saw, I really confuse about this. And I do see many books for the reasons , but I can't get the answer, please tell me !

Answer: "Interface" is the Java way to do multiple inheritance, or a better way to think of it is as a way to design plug-ins.

For example, let's say we have an application that monitors a network of computers. Our monitors might check for web pages, or they may check for other ports, or they may have hooks for hardware checks.

The interface to our main control panel is always the same: We need some means to poll the monitor object for an answer. This is the "NetworkMonitor" interface and all network monitors will share this interface, but they may have a class heirarchy that is very different, for example, port-monitors may all fork a thread that periodically checks whereas our control panel interface just asks for the most recent answer; hardware monitors may ask for their data in real-time or over RPC and thus have no need of inheriting from Thread.

Because they share the same Interface definition, the control panel application does not need to know if they are polling monitors or real-time monitors because, from the control panel's point of view, it does not matter
--
...
P.S. by John
Also interfaces make our life (I mean programmers) much easier.
Imagine a big project ( a lot of programmers, many packages, thousands of files): it is impossible all the time to be aware about particular implementation of some method in specific class in another package!
Much easier to define interfaces between packages as some kind of bridges or gates into another package and their functionality, with hidden (for us) implementation. We just know interface and method names. It is enough to use those methods.
How it is implemented there does no matter... It is working!

Q: I find such function, but in Class class. Is it possible to use it (or something like this) in another class? Exactly I would like call: super.super.method1();

Answer: No, you can only access methods of your direct ancestor.
The only way to do this is by using reflection:

import java.lang.reflect.*;

Class superduper = this.getClass().
getSuperClass().
getSuperClass();
Method ssMethod = superduper.getMethod("method1",null);
ssMethod.invoke(this,null);

(I didn't check the exact syntax and parameters of the above code.)

However the necessity to skip a hierarchy shows that there is something wrong with the design of your application. You should fix this instead of using dirty tricks like the above.

---
Carl Rosenberger
db4o - database for objects - http://www.db4o.com

📘 Q: I am facing a problem that might sound common to you all. I have to write a switch statement with 40 case statement. I am wondering if there can be some way to reduce this. If anybody can provide some tips on how to design this smartly. ;)

Answer: The alternatives are:

an array you index into to get values or delegates.
an array you binary search.
--
Roedy Green

📘 Q: Does anyone know what the character limit for a class name would be?

I have my class name in a variable ft and I am doing a Class.forName(ft);

My class name with qualifiers is 44 characters long, I can only imagine that there must be a limit ....

Answer: From the JVM Spec 2 - 4.2
----------------------------------------
Class and interface names...fully qualified name...as
CONSTANT_Utf8_info.
----------------------------------------

That structure, CONSTANT_Utf8_info, places a limit of 64k on the length. But for a class this is the fully qualified name, so the maximum length of the name could be less.
--
jschell

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## General Java Questions - V

Q: I can't find the API documentation on any classes in the sun.* packages. Where is it?
Answer: The short answer is that SUN provides documentation only for the public classes in java.*. SUN does not provide documentation for sun.* because those are the Sun-specific implementation, and specifically not part of the Java technology API standard, and are therefore subject to change without notice.

In general, SUN doesn't provide javadoc documentation for sun.* classes in order to discourage developers from writing programs that use them. For further explanation, see the next question.

However, if you must have it, the documentation for sun.* is available separately, here: http://java.sun.com/communitysource/index.html For example, the doc comments for sun.net are in the source files located at:


/src/share/sun/sun/net/*.java

This source code release does not include javadoc-generated documentation. You would have to generate those docs yourself using javadoc.

source: http://java.sun.com/products/jdk/faq.html#A12

Q: How do I copy one array to another?

Given that I have an byte array defined like this:
byte byteSmall = new byte[23];
and another larger byte array defined like this:
byte byteBig = new byte[30];

How do I copy byteSmall into byteBig starting at index 7 without a for loop like this:

```
for(int i = 0; i < 23; i++){
   byteBig[i + 7] = byteSmall;
}
?
```

Answer: See System.arraycopy:
"Copies an array from the specified source array, beginning at the specified position, to the specified position of the destination array. A subsequence of array components are copied from the source array referenced by src to the destination array referenced by dst. The number of components copied is
equal to the length argument. The components at positions srcOffset through srcOffset+length-1 in the source array are copied into positions dstOffset  through dstOffset+length-1, respectively, of the destination array.

If the src and dst arguments refer to the same array object, then the copying is performed as if the components at positions srcOffset through  srcOffset+length-1 were first copied to a temporary array with length components and then the contents of the temporary array were copied into positions dstOffset through dstOffset+length-1 of the argument array."

Q: Where can I get source code for Java 2 platform?

Answer: In most cases you do need the code itself. Often it is enough all those tools and resources that are included into standard Java SDK.
Anyway you can always find it here:
http://www.sun.com/software/communitysource/java2

--
AP

Q: More about Robot! I met with a problem in using class Robot.mousePress...

The compiling process is successful. But when I run it, I receive "IllegalArgumentException:
Invalid combination of button flags". I don't quit understand this information. Part of my code is as following:

```
Robot rMouse=new Robot();
int button=1;
rMouse.mousePress(button);
rMouse.mouseRelease(button);
```

I am really confused. Will you please give me some advice? Thank you in advance!

Answer: You are not using a valid value for the argument to the mousePress() and mouseRelease() methods. If you check the API documentation, you'll find the valid values are a combination of one or more of the following constants:

InputEvent.BUTTON1_MASK
InputEvent.BUTTON2_MASK
InputEvent.BUTTON3_MASK

plus others which represent the Ctrl, Alt, and Shift keys. To press the left mouse button, you want to use:

rMouse.mousePress(InputEvent.BUTTON1_MASK);
--
Lee Weiner

📃 Q: Does the java class that call the JNI need to be in the same directory of the DLL?

I have the java class in a package and in a jar file. The DLL it uses is outside of the jar file. The JNI call does not work. I am confused with the path.

Answer: As long as the library is in your "path" you should be able to load it with java.lang.System.loadLibrary.

On Linux it should be in your LD_LIBRARY_PATH and on Windows in your PATH.

With java.lang.System.load you can load the library from a predefined location.
--
Flaps

📃 Q: ActionListener is an interface and we cannot instantiate an interface. Then how is it possible to do something like this (see below)...?

```
textField[0].addActionListener(new ActionListener(){
   public void actionPerformed(ActionEvent e){
      handleSearch();
   }
});
```

Answer: That notation is essentially shorthand.
The compiler fills in the equivalent of
XXXX extends Object implements ActionListener {
}
--
William Brodgen

📃 Q: By default an application has no security manager and Java runtime environment does not create automatically a security manager for my application. How then applet where I am not creating any security manager already prevented from many operations?

Answer: It is true - Java by default let you do whatever you want and then it is your responsibility to restrict something. In case with applets a little bit different story - applet viewers and browser have THEIR OWN security manager, not from JRE. That's why even if you did not define any security manager in the code of your applet, during the start up time browser/viewers will use their own security manager. This manager is built in into their application (browser, viewer), not your Java applet.
--
AP (J.A.)

📃 Q: Does java support asm?

Answer: You can't directly embed ASM code in your java source. However, you could just use the Java Native Interface to call a C (or whatever else you like) library that embeds the ASM code. This will make you platform-dependent!

--
Sebastian

📖 Q: I am a Java beginner. I just want to ask what's the difference between

int [] number;
   and
int number[];

Answer: The "postfix []" syntax is to make the language more comfortable with C programmers. In Java, the "postfix []" binds to the variable name, while the "prefix []" binds to the type name.
In C, there is only the "postfix []". But C also includes pointers, which Java doesn't. The pointers are identified by having a * between the type name and the variable name.
While there is no danger of ambiguity in this declaration:
int a[], b;
there IS danger of ambiguity in this declaration:
int *a, b;
In the latter case, the C rules are that a is declared a pointer, while b is not.
Java offers the following different ways of declaring arrays:
int a[], b; <--> int[] a, b;
These are not equivalent, because b is an array only on the right side.
Whereas with C-style pointers these would become:
int *a, b; <--> int* a, b;
which breaks the concept of "white space doesn't matter", so it will not be as easy to scan and parse.
So the designers of C decided to have only 1 form of defining variables with pointer types or array types, and this was that the * or the [] binds to the variable, not to the type.
Java carried over the variable binding, but also introduced the type binding, because there were no more pointers, so the ambiguity was removed.
--
Joona Palaste
And think also about marketability!
Making Java's syntax highly similar to C and C++'s during its early years:
* Made the language seem more familiar to C/C++ programmers.
* Decreased the learning curve for C/C++ programmers.
* Gained credibility through the frequent inference that it was a next-generation descendant of C++.
* Increased its pool of programmers by attracting C/C++ programmers through the effects above.

This isn't the only instance of anachronistic syntax; consider the optional semicolon at the end of a class declaration.
--
Andrew R.

📖 Q: Besides speed, what does the IBM JDK have that sun's does not? Are there any compatibility problems?

Answer: If there are problems, then they represent bugs in one JDK or the other. To the developer, they should present identical APIs and behaviour.

--
Jim Sculley

📋 Q: I've seen a few examples were for loops are used in different way

```
for(;;) {
}
```

does this do the same as a

while(true)

is this a speed issue?

Answer: The two are identical. It's simply a matter of style and personal taste.
--
Kevin Riff

📋 Q: When do I need to use overloading in a program? And why?

Answer: Overloading is best used when you want to have a function that has the same name but can accept different arguments. The only restriction is that it must return the same type.

e.g.

```
public String aName(String str) {
    String retStr;
    retStr = "Hello" + str;
    return retStr;
}
```

// Now the same function but overloaded

```
public String aName(String str, String str2){
    String retStr;
    retStr = "Hello " + str + str2;
    return retStr;
}
```

Both functions return the same type (a String) but have different signatures.

You will find this used a lot with constructors. See any recommended text on Java for a much better explanation.
--
Anthony Miller

📋 Q: Why people claim that java is a secure language...

Answer: The java programming language itself is just as secure or insecure as any other programming language. A Java Applet (which are used on the web) is a different matter. They are quite secure.

An applet does not have access to system resources outside of the directory that it is created in. There are ways that you can give an applet access to this information, but they are pretty explicit and you will most likely know that the applet is trying to do this.

I am not really familiar with the subject but did a little reading on the sun website. Check it out for more info:

http://developer.java.sun.com/developer/technicalArticles/Security/

You aren't supposed to be able to break the rules easily at run time. The elimination of pointer arithmetic is supposed to improve type safety.

The Code Red worm is a good example of the weakness of the 'old-style' languages. This worm uses a 'buffer overflow' attack to 'trick' the IIS program in to doing it's bidding.

The Java system is designed to make it harder to make the programming mistake that Code Red exploits. It is important that you understand that Code Red would not have been possible without careful attention to a fast buck by Mr William Gates III and his cohorts.
--
Dr Hackenbush

Q: This is a basic question... While looking at java API s I noticed that there are classes that start with java and javax (like java.lang.util). What is the basis for this?

Answer: The size of the core part of the Java platform has been growing steadily since the release of version 1.0. The first Java platform had 8 core packages, in version 1.1 there were 22 packages, and in version 1.2 there are over 50! The extensions framework provides a standard means to add functionality to the Java platform for use when needed, without having to increase the size of the core API. Standard extensions have names in the javax.* namespace.

However, just because a package name begins with javax does not guarantee that that package is an extension rather than a part of the core platform. The Swing packages have names beginning with javax indicating that they were non-core extensions prior to version 1.2 of the platform. They are part of the core 1.2 platform even though they are in the javax.* namespace.
For more details please see:
http://java.sun.com/products/jdk/1.2/docs/guide/extensions/ext_faq.html

Q: Does anyone know if there is a way to prevent System.exit() being called?

Answer: Look into java.lang.SecurityManager. It has a method called checkExit() with which you can prevent System.exit().
This method throws a SecurityException if the calling thread is not allowed to cause the Java Virtual Machine to halt with the specified status code.
--
David Zimmerman

# Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

☐ IT Professional   ☐ General Computer   ☐ Design and Graphics   ☐ Linux   ☐ Freeware/Shareware

☐ Games     Intranet     Computer Hardware     Web Design   ☐ Computer Software

☐ UNIX   ☐ Web Software   ☐ Windows Software   ☐ Personal Finance   ☐ Programming Software

☐ Servers   ☐ C/C++   ☐ Powerbuilder   ☐ Perl   ☐ XML

☐ Java   ☐ SQL   ☐ Design/Graphics   ☐ Utilities   ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Java HardWare (not software? are you sure?)

Q: Can anyone please direct me to some literature pertaining to the Java chip.

Is the project still being pursued by Sun? I remember hearing about it in 2000 but haven't heard of any companies adopting the chips. This makes me worry about the chip's cost and availability.

Answer: You can find some companies producing Suns java chip and different ones on the link section of http://www.jopdesign.com.

Sun has had at least two "Java chip" projects that I know of.
PicoJava was the earlier one -- I thought it died quietly, but I see that Fujitsu lists a "32-bit PicoJava-II microcontroller", so maybe not. The more recent one is MAJC, but it isn't aimed at small or low-power devices.

New ones keep coming out of the woodwork, e.g. this announcement yesterday:

http://dailynews.yahoo.com/h/nm/20010604/tc/tech_omron_chip_dc_1.html

If you want to experiment with one that's available now and does real-time Java, see

http://www.jstamp.com/

A fair number of us have also used the TINI board

http://www.ibutton.com/TINI/

though it is aimed at industrial controllers, not PDAs. (It's a conventional CPU running a JVM, not a "Java chip", but that may be a silly distinction).

Imsys makes a Cjip: http://www.imsys.se/

http://www.ptsc.com have had their PSC1000A 32bit 100MHz $10 'Java' chip

available for some time. I've been using it (but not with Java) for over a year. It rocks. They also make a complete module (using a PSC1000A) called Ignite1.
--
Thomas Maslen

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

| ☐ IT Professional | ☐ General Computer | ☐ Design and Graphics | ☐ Linux | ☐ Freeware/Shareware |
| ☐ Games | Intranet | Computer Hardware | Web Design | ☐ Computer Software |
| ☐ UNIX | ☐ Web Software | ☐ Windows Software | ☐ Personal Finance | ☐ Programming Software |
| ☐ Servers | ☐ C/C++ | ☐ Powerbuilder | ☐ Perl | ☐ XML |
| ☐ Java | ☐ SQL | ☐ Design/Graphics | ☐ Utilities | ☐ Handhelds |

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

Job, fun, other...

Q: Please clarify my confusion about Java versions. I hear Java 2 but JDK is 1.2
Now we are going to use JDK 1.3 on the job and I said that this is Java 3 and probably soon when JDK 1.4 will be released we will use Java 4.
My chief said me that I am wrong and he never heard about Java 4. He insists on it, but cannot explain why.
Maybe because of he is chief? Please explain...

Answer: He is right and not because of the fact that he is chief :-)
Although exist one old set of rules:
1. Chief is always right!
2. If chief is not right please see line number 1.

I think we have to wait a few years more until Java 3 comes to us.
It is SUN's numbering policy. All JDKs that have numbers 1.2, 1.3, 1.4 are Java 2 JDKs...
Difficult to understand but easy to remember...
--
AP. (J.A.)

Q: Hi, I am learning java and hoping to take the certification test this april. How important is it to gain this certification? And would employers be interested?

Answer:

1. Look at the job ads... do they ask for it?
2. If you are an interviewer with two identical candidates, but one had this. Who would you employ?

Q: ...However I am having a difficult time finding a job.

I have been studying Java since December and passed the SCJP a few weeks ago. However I am having a difficult time finding a job. I have found that most of the positions require a good deal of experience first. Also, even the entry level positions

have not responded to my resume.

Two questions:

Any ideas on how I can make it look better?
    and even more importantly,
What should I begin learning next?

(There are so many advanced concepts, JavaBeans, Swing, JSP, EJB etc...that I am a little lost)

Answer 1: I graduated with a bachelor's degree in college just a few months ago and I am currently working for a company that uses Java quite heavily. I was lucky that I got onto the staff that I am, but it wasn't all luck. Companies are looking for people who can not only program, but also communicate well and find the answers to questions on their own. Personally, if I were an employer that saw you completed the tests on your own, that would tell me that you have self-motivation and the yearn to learn: a definite plus considering how fast technology changes.

Bottom line is this: getting a job is tough, no matter what level experience you have. The main thing to keep in mind is that time is on your side. Your patience will win in the end.
--
by RyanDecker

Answer 2: Try with different government institutes (i.e. local, municipal, federal) when you are first starting out. Usually the more experienced programmers in the government go to the private industries after a few years and the government is looking for people to replace the ones who have left and this is usually a good place to start.
I hope this helps. Good luck in your search.
--
Robert

Q: I would like to know if the small programs that are attached in mail (like the small games for example) are usually coded in java or in any other language?

Answer: Usually not. Usually they're viruses.
--
Nils O. SelÅsdal

Q: An ordinary guy of ordinary intelligence, with some experience of html, do I stand a chance of learning java or do I need to be of an above average genius.

Any feedback would be welcome!

Answer: Java is extremely easy to learn...but will take a lifetime to be a master. Intelligence is not an issue.
--
drichard_007

Q: I have recently passed my Java Programmer Certification in the UK. Is there a

standard logo I can use for my homepage & resume?

I have contacted Sun with this matter, but have received no reply.

Answer: You will receive information concerning how to get the logo, along with a "logo use agreement" from Sun in your SCJP package. As I recall, according to the agreement that you have to sign and send back to Sun before you are given the URL,
Before, when you get old agreement, you agreed not to place the logo on a web page or on your resume.
With new agreement you may use the logo in your resume. The new agreement is now the standard. All the candidates who signed an old agreement will be under the rules of the new agreement.
--
Clyde, Tym Tyler

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

| | | | | |
|---|---|---|---|---|
| ☐ IT Professional | ☐ General Computer | ☐ Design and Graphics | ☐ Linux | ☐ Freeware/Shareware |
| ☐ Games | Intranet | Computer Hardware | Web Design | ☐ Computer Software |
| ☐ UNIX | ☐ Web Software | ☐ Windows Software | ☐ Personal Finance | ☐ Programming Software |
| ☐ Servers | ☐ C/C++ | ☐ Powerbuilder | ☐ Perl | ☐ XML |
| ☐ Java | ☐ SQL | ☐ Design/Graphics | ☐ Utilities | ☐ Handhelds |

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Miscellaneous - I

Q: What is the difference between Java and Microsoft Visual J++? Is there any specific difference? Please explain me I am novice to this field Part1

Answer: VJ++ was designed as a "spoiler" product -- Microsoft created some deliberate subtle incompatibilities so that when people wrote VJ++ code it would only run on Windows. This is not my opinion. This came out in evidence in Microsofts trial for anti-competitive behavior.
In email they said things like they were going to "pollute" Java, and "piss on" the Swing libraries.

Some of the incompatibilities were in removing some standard fields from the system libraries. Others were adding some fields. They also added some differences to the language itself, in the way event
handlers were registered.

When Sun found out about Microsoft's attempt at sabotage, they cut-off all code deliveries to Microsoft, and sued Microsoft. That lawsuit just ended with the payment of $20M by Microsoft to Sun.
So Microsoft VJ++ is several releases out of date, and does not have many of the most important libraries, such as RMI and beans.

Bottom line: even though it has a nice GUI IDE, if you want to program in Java, you are better off avoiding VJ++, and using any of the free IDEs mentioned in the Java FAQ.

by Peter van der Linden http://www.afu.com
P.S by John: please read second part of this tip tomorrow and you will see you can use it with JDK1.3!

Q: What is the difference between Java and Microsoft Visual J++? Is there any specific difference? Please explain me I am novice to this field Part2

Answer 2: Microsoft Visual J++ is a Java IDE for editing, compiling, and debugging Java source code. It also provides GUI editing tools that generate code for you you.

Visual J++ added some extensions to the pure Java language that you can easily disable.

The latest version of Visual J++ is 6.0. It supports JDK 1.1.
In order to use JDK 1.2, JDK 1.3, or beyond you must follow the procedure below:

Modify the CLASSPATH in the registry. Run RegEdit.exe and locate the following key:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Java VM

and modify the CLASSPATH string to include the rt.jar file from you JDK distribution. For example:

c:\Program Files\JavaSoft\JRE\1.3\lib\rt.jar

needs to be added to the end of the CLASSPATH string, and don't forget the semi-colon separating each entry in the string. Also note the "." at the end of the CLASSPATH string.
--
Roger L. Cauvin
rcauvin@homemail.com
http://www.thegym.net/rcauvin

Q: I want to know, can Java be run in DOS?
Not Visual J++ but just Java, can it run in DOS?
If so is the language is the same? I have been told that
java can run in DOS but I just want to verify...

Answer: Java technology relies on files with longer names than the old DOS limits, so if you are running older (pre-Microsoft Windows 95) versions of DOS, Java technology will not work. If you are running a version of DOS (such as the one that comes with Windows 95) that allows long file names, you should not have any problems.

Note that older versions of WinZip do not support long file names, even if the installed version of DOS does. You can get a free upgrade of WinZip with support for long file names from their web site.

source: http://java.sun.com/products/jdk/faq.html#E1

Q: I am just wondering. What is the difference between java and java2?

Answer: Java2 is really just another name for JRE/JDK1.2 and higher. Various things improved dramatically between JDK1.1 and JDK1.2, mainly in the libraries - things like the collection classes suddenly appeared.
The extensions mechanism was also introduced, and I suspect the security policy was firmed up (I've never needed to investigate that, fortunately).

Q: I am curious to know why Sun seems to ship two virtual machines (VMs) with

their software development kit (SDK, aka JDK). I am talking in particular about JDK 2 v1.3 for win32, although this question may apply to other versions.

Here is where the VMs are on my installation:
c:\Program Files\jdk 1.3\bin\java.exe
c:\Program Files\jdk 1.3\jre\bin\java.exe

Incidentally, it also appears that there are lots of libraries in the jre\bin subdirectory that are not present in the other bin directory.

Both files are 21k on disk, and it seems unlikely that either of them are actually a true VM.

Answer: The EXE files aren't the VMs, they're just wrapper that invoke the VM.
The actual VM files are called JVM.DLL, and you'll find them in:
jre\bin\classic (basic, interpreted VM).
jre\bin\hotspot (the real thing).

If you invoke java.exe, you'll get hotspot by default. Use "java -classic" to get the old VM.
The main difference between them is that when you run java then you also get access to the java compiler and some extra "developer" stuff.
When running JRE you get a more streamlined version. Notice that it is possible to download just the JRE, which is a lot smaller than the full JDK.
--
kenny

Q: How to set the default memory limit of java virtual machine in a java application?

Answer 1: java -Xms16m -Xmx32m MainClassName
here: -Xms16m => 16meg initial memory allocation
-Xmx32m => 32meg max memory allocation

Answer 2: Run your Java program with -mx<number> switch like this:

java -mx128m ClassName

This for example will set maximum memory allocation pool to 128MB

Q: Do the classes in java.util.zip handle password-encrypted zip files? I've looked through the API, and I don't see any mention of it.

Answer: No, they don't. But Zip's built-in encryption isn't safe anyway by today's standards.

Q: I have a ZIP file and I want to check the file type of its entries WITHOUT unzipping it. Anyone knows how to do it?

Answer: You can't do this, unless knowing the file suffix is enough information for you. Then you can simply list the file names using the ZipFile class "entries()" method.

If you cannot determine the file type without examining the file itself, then obviously

you must read the file, i.e. unpack the ZIP file.
--
Paul Lutus
www.arachnoid.com

Q: I want to compile a source file(.java) and then run it (.class) from inside another java program. How can I do that?

Answer: Make sure you place tools.jar on your class path.

If using JDK < 1.3, then use

public static int sun.tools.javac.Main.compile(String[] args);

if using JDK >= 1.3 then use

public static int com.sun.tools.javac.Main.compile(String[] args);

then if the result code indicates all is well (==0) the just go ahead and load the resulting class file, probably in a new class loader.
--
Neal M Gafter

Q: How do I make java apllication instalable? I have written a Java application and have the .class files in one location. How do I make it installable? Is it possible to convert it to an executable file?

Answer 1: just create a batchfile or a .sc file (if on unix)
Answer 2: Or check http://installshield.com/
They have special edition for Java which lets to write additional interface for configuring of program parameters during instalation.

Q: I've got a Java application that I need to install on my future customer's computers but this is the first program that I've ever tried to sell to anybody and I want to keep my costs down.

Is there something out there that installs the JRE, if it needs to, and then my program ?

I know of InstallShield, InstallAnywhere and the like but I'm hoping to find something a lot cheaper since I'm just a small time operator (who may not sell even one of his programs).

Answer:
1. ZipCentral is a free Windows app that will create a zip file, or a self-extracting executable. It has the ability to execute a BAT file after unzipping your files. You can download a copy from

http://zipcentral.iscool.net/

2. Sun's Java Web Start (the price is right - free):

http://java.sun.com/products/javawebstart/index.html

JavaTM Web Start -- a new application-deployment technology -- gives you the

power to launch full-featured applications with a single click from your Web browser. You can now download and launch applications, such as a complete spreadsheet program or an Internet chat client, without going through complicated installation procedures

3. Zerog's Now! http://www.zerog.com/downloads_01.html

There are no restrictions on the use of free products: InstallAnywhere Now! and PowerUpdate Now!. Now! 4.01 is free, installs a JRE

Q: I need to know how to run Multiple JVMs on one machine. As far as I know if I run 2 different Java Programs on one machine at the same time, both of these programs use the SAME JVM. Is this statement right?

Answer: NO!. Each invocation of the 'java' command creates a new and separate JVM, at least in those JVMs based on the Sun code, and getting them to cooperate on a shared Java task is not automatic, and definitely non-trivial.

Q: I am totally confused about the differences between the SDK, JDK and IDE products

I am brand new to the world of Java and am most interested in learning the language. However, I am confused about the differences between the SDK, JDK and IDE products. I have gone to the Sun site and even some of the IDE vendor sites and I have yet to find something that tells me what the differences are between the three and which of the three I need to program in Java.

I want to program using some form of IDE but do I need to separately download and install a SDK and/or JDK? Sun needs to improve their documentation for us new to the Java environment.

Answer 1: IDE is an acronym for _I_ntegrated _D_evelopment _E_nvironment. These products are the one stop shops for coding, running and debugging your code. Often these will include GUI based drag and drop form designers and "wizards" for the shells of common forms of code (Application, Applet, etc.) JBuilder is an IDE.

The IDE may stand on its own, or it may act as a front end for a JDK.

JDK is _J_ava _D_esign _K_it. A JDK is a command line based interface to the JVM, plus the classes. You are responsible for your own editors, creating code for GUI elements, and all code. All of the IDE's I have reviewed personally come with JDK or their own vendor's equivalent (JVM and class libraries). Some IDE's are capable of a sort of "upgrading" by downloading the latest JDK from Sun (JBuilder for example).

Answer 2: If you want to write a Java program using any editor and not an IDE then you would want to download the JDK. It will let you compile and run Java programs from the command line (like a DOS window).
JDK stands for Java Development Kit and SDK stands for Standard Development Kit.

Java comes in three versions - Standard, Enterprise, and Micro editions. JDK could be any one of the three. SDK is the standard one - this is the one most people use. If you want an IDE they typically come with a JDK so all you would need to do there is

download the IDE and start using it.

Q: How does a java application stored within a jar file reference/edit/read other files ( like .txt, or data files,) that are also within the jar file?

Answer: Classes located in a JAR archive are loaded via a class loader whose purpose is to load classes form JAR archives. This ClassLoader implements the getResource and getResourceAsStream methods to retrieve files from the JAR file. So you can take any class from the JAR and say ClassName.class.getClassLoader().getResource("fname"); to get the resource and use it.

Q: I want to keep my java GUI always on the top of any other desktop application. Any idea?

I want to keep my java GUI always on the top of any other desktop application. Any idea?

Answer: Spawn a thread that knows about the parent Window, and every X milliseconds, executes the toFront () command of that window. Just remember to execute it using SwingUtilities.invokeLater (), and don't
let your users launch two apps, unless you enjoy screen lockup.

Q: I am beginner in Java and know that Java program runs on Java Virtual Machine. But how does Java works with file systems? They are real. How Java does solve this problem? For example if I want to modify ACL (Access list) on UNIX. There are no ACLs in Windows. No such method in API as well.
Does it mean that file system of Java is quite limited?

Answer: Yes, it is limited in some sense... Java supposed to run everywhere. "Everywhere" actually means that SUN limited file system functionality by common (for most OSs) methods. It is strength of Java and weakness as well.
Since Java 1.2 we got at least some functions like: setLastModified() similar to touch() function in UNIX.
Or listRoots() returns the list of all disks available in the system. It is important for Windows systems (On Unix all partitions are mounted to root "/." and has no names like in Win: A:\, C:\)
I hope in future we will get even more.
--
Luis

Q: Can a java application be run of a CD without installing anything (i.e. runtime, etc) on the target computer?

I would like to put my application and hand it out as a demo, but I want to make it easy to view.

Answer 1: by Dale King  The JRE was made so that it didn't need to be "installed". What I did in one case was to simply put the JRE into a jre folder in the same directory as my application then invoke it from that directory using:

jre\bin\jre.exe -cp MyJar.java MyClass

That was for JDK1.1 and you have to modify it slightly for Java 2. But this did not require any installation of environment variables to be set up. The JRE was smart

enough to know how to get to its system classes relative to where the jre.exe file was located.

Answer 2: you could try a Java to native compiler.

Q: I would like to know whether it is possible to test the memory, so as to avoid the OutOfMemoryError or whether it is possible to increase the amount of memory in the JRM.

Answer: You can get the total and available memory used by the VM by making two calls from the Runtime class:
```
   Runtime runtime = Runtime.getRuntime();
   long free = runtime.freeMemory(); //the available memory
   long total = runtime.totalMemory(); // the total for the JVM
```

The amount returned be totalMemory() isn't that useful unless you specify how much memory your program will have from the beginning (if you don't, the JVM will just keep grabbing more until you run out).
You can set the initial and maximum memory from the command line:
java -Xms64m -Xmx64m name.of.Application
This will start your appplication with 64 megs initial and maximum memory.
--
Corey Wineman

Q: What needs to be done to reduce size of a jar file? What optimization techniques to use on classes inside the jar file? What tools if any?

Answer: A JAR file is a ZIP archive. You can influence its size by choosing the degree of compression you want to have. This is usually defined by a value between 0 (no compression) and 9 (maximum compression). Although JAR tool does not list a -9 switch, you might want to create compressed JARs with any ZIP tool like Winzip or the free Info-ZIP zip command line tool.
The amount of reduction you get totally depends on the nature of your data.
Note that if you use compression in your JAR file, loading goes slower (classes must be decompressed).

Q: Is there any way to run code after the VM has been instructed to exit?

Answer: In 1.3, you can use Runtime.addShutdownHook(Thread hook)

Q: Where can I find Java --> Native code compilers?

Answer: We just published a list of Java products that lets you do
Java --> Native code compilation:
http://javafaq.nu/java/staff/staff.shtml

Q: I have a directory having class files arranged in package hierarcy. How can I make the executable of this whole directory?

Any application available for that in Windows NT environment.

Answer: Make a JAR file out of it and add a manifest file that indicates which main() method of which class must be called.
Double-clicking this JAR file will run your application.

Q: I'm interested in writing a little mp3 player in Java...

I'm interested in writing a little mp3 player in java. I have an entirely different app right now that plays sound (wav files), and I substituted an mp3 file for one of the waves but it didn't work. Can anyone tell me if java even supports mp3 files?

Answer: Go to the "Products & APIs" section of java.sun.com and look for JMF (Java Media Framework). It's a library that also supports reading MP3 files.

Q: Are there any tools out there that will convert a program writen in C to JAVA?

Answer: Yes.

C2J: http://www.novosoft-us.com/NS2B.nsf/w1/C2J
C2J has successfully compiled itself as well as programs such as PGP and YACC. Obviously YMMV.

Based on C2J is a C++ to Java tool: http://sol.pace.edu/~tilevich/c2j.html
--
jim

Q: Can we create DLLs in java??? if yes How???

Answer: Unfortunately it is impossible. DLL stands for Dynamic Linking Library and has definite structure inside. DLL is a part of executable code and helps to make an application for Windows to be more smaller. And more flexible. It is something like classes but compiled (Java class files are byte codes and JDK compiles them during the runtime...).
In Java it is not possible to make an executable code. But with third party software Yes!
See native compilers on our site.
But I didn't hear about creating DLLs.

So my answer is: in Java it is not possible, but with third party applications it is possible theoretically. Although I do not know any compilers that produce DLLs there is no limitations to do that....
--
John

Q: Can someone tell me the difference between the JRE that comes with the J2SDK and the stand-alone JRE?

Can someone tell me the difference between the JRE that comes with the J2SDK and the stand-alone JRE? When should I use which? I read the sun's web page and they said the the JRE stand-alone package is for shipping your application, it doesn't come with a compiler nor debugger, but what does it really mean?

Answer: Exactly that. The Java 2 SDK (aka JDK) is the JRE plus the compiler (javac) and debugger (jdb). The JRE is entirely sufficient to run a Java application (with a couple exceptions in situations where you call into the compiler or some such). It's entirely redistributable with your Java application if you've written an app in Java intended for an audience that may not have the JRE installed.
--
Chris Smith

Q: Does anyone know of a java machine that will run from a 1.44Mb Floppy? I

have an application I want to run from a dos 7.1 floppy disk.

Answer: please check here:
http://www.transvirtual.com/kaffe-features.htm
there is written:

"Efficiency is not just about execution speed, but Kaffe's JIT is quite speedy: it runs Java code only 30% slower than plain C. Such things  as memory consumption of a JIT-enabled, graphical Kaffe are also important.
Here we can execute a full system on a 4 MB DOS system, and the VM and library footprint won't exceed 1 MB. Our complete source tree fits on a single 1.4 MB floppy. This is what we mean by "efficiency".

Good enough for you?

Q: Anybody know a good tool to distribute your made java classes with for windows platforms?

I want to create a setup file that alse verifies existence of java runtimes etc etc and creates a shortcut and that kinda things...I tried looking into Wise installation systems and Installshield asswel, but they are both not really build for this kinda things...

Answer: You should try, InstallAnywhere by ZeroG (http://www.zerog.com)

ZeroG has a free version call, "InstallAnywhere NOW". You should give it a try, its a really easy-to-use java installer. It has all the features you need, such as creating an Icon on your desktop, creating a folder under the Windows "Start" button, it can be set to have the user search for a JVM on their computer, and much much more... (I sound like a commercial, hehehe)
JHig310336

Q: I see all the time in the code examples that some Exception is caught but has empty body. Is it a good practice?

Answer: No, it is not! Of course it lets to run program but does not treat the problem. It should be logged at least for further analysis. This "lazy programming" approach finally fools even the author.
For example, in case of network connection: no Exception report, and of course no connection. If your program full of such Exception handlers you will just sit and guess what has happened.
--
AP. (J.A.)

Q: Heap size limit!! I am running JVM from JDK 1.2 on Solaris 2.7 and I couldn't allocate the max heapsize over 2G when I invoke the JVM.

I have repetive tasks that take 500m memory each to run, so I naturally want to run as many threads as possible. I figured out this 2G (-mx2047m) limit by trial and error but is there any way out of this? My workstation happen to have 2G physical memory, and the file size limit is 2G as well (from ulimit), are there any co-relation among those numbers?

Answer: Yes, there is a relation: both result from limiting addressing space to what you can get with signed 32-bit ints for addresses:
$2^{31} - 1 = 2 * 2^{30} - 1 = 2 * 1 GB - 1 = 2 GB$

One of the interesting features of Java, is you could run the code with 64 bit addresses, and nothing would need to change in either the source code or the class files. The only difference would be you could hold a lot more objects and stack frames before you blew virtual RAM.

Obviously you would need a different JVM or Hotspot.

Java never lets you discover how big references really are inside or how they are implemented, e.g. as pointers or as handles, or even the granularity of the addressibility of your machine.
On a 32-bit OS it is a liitle hard to get 32+ bit memory space. How about you try a 64-bit Solaris 8?
answered by Michiel,
Roedy Green, JAVA GLOSSARY see http://www.mindprod.com/jgloss.html
and Johnny Bravo


Q: How can I format a diskette using Java? And, what's the way to get the diskette's size?

Answer: As far as I know: no. Formatting a disk is strongly system dependant, so you can hardly do this with java.

You *can* however start a new process, have it open a shell or any other kind of command processor your particular operating system uses, and issue the format command.
> And, what's the way to get the
> diskette's size?
Again system dependant.
--
Ansgar W. Konermann

Q: I am a 17 year old in the JDK version 1.1 . Is it worth upgrading and getting a later package, and learning swing?

If so, why?

Answer: Yes.

Java 1.3 provides many performance improvements and library additions which will make your life as a programmer easier.

Swing is "better" than the AWT in that because it is mostly written in Java - it has the advantage of the same look-and-feel across different platforms (which the AWT was crap at). The reference of the other learned poster to a mainframe is the common argument that Swing is slow. This _is_ true, however only if you build large applications and I doubt that at the moment you will.
In the future you also have the option of compiling your Swing classes to native code (viewed as a venerable evil by some some, but a practical solution by others) or performing other oprtimisation tricks that will see your GUI apps not only looking great and being cross-platform, but also performing well.
It's what we call in the trade a no-brainer. Time to get Swinging young chap.
--

pip

📄 Q: Is it possible to create a Jar file that is not unjarable?

Or only unjarable on a certain domain/server? Is the jar.exe JDK-specific (I don't believe so)? Was I just asleep at the command line and imagining the whole thing?

Answer: You could conceivably encrypt a jar with a password, but you wouldn't be able to start the application from that jar. You could have another jar that actually knows how to decrypt it and creates its own decrypting class loader. Of course your startup jar could be hacked allowing someone to figure out how to decrypt the jar. So once again, you can slow down the process and make it more painful but you can't make it impossible.

To make this workable it would probably be a lot easier to encrypt the files within the jar rather than the jar itself, since you need random access to the jar file, but only sequential access to the files within. It is more difficult to write a good random access encryption scheme. This would allow you to unjar the files, but the files would be unintelligible. You might also apply a cipher to the file names in the jar so someone would not know whether a file was a class or a resource file.
--
Dale King

📄 Q: How does the system find the path to JDK 1.2 after I type in "java -version"?

I installed the jdk1.2 on a NT system, also VisualCafe4.1 with jdk1.3 was installed on the same system. After typing in "java -version" in a DOS window, I always see java.exe 1.2 is invoked even though I couldn't find the path to JDK 1.2 from the system environment parameters. How does the system find the path to JDK 1.2 after I type in "java -version"?

The reason I ask this question because I want to invoke jdk1.3 under a DOS window without uninstall jdk1.2. I did add a path to jdk1.3 in system environment and reboot the system, but JDK 1.2's java.exe was still invoked after typing in "java -version" in a DOS window.

Answer: Because when the JDK install kit placed the two programs java.exe and javaw.exe in the WINDIR and that's in the PATH. And these programs read the registry to find the JDK/JRE.

If you placed jdk1.3\bin in the PATH before WINDIR, you should be fine. From your description, I guess you didn't, so you're still getting jdk1.2. Or you can directly execute jdk1.3\bin\java.exe which will work.

Another trick I use to swap JDK's in an out is to place the directory named java\bin into the PATH and just rename the desired JDK to java, like this:

c:\>dir j*

Volume in drive C has no label
Volume Serial Number is 07CF-0B17
Directory of C:\

JDK12~1 2 <DIR> 03-19-00 1:57a jdk1.2.2

```
JDK13~1 0 <DIR> 12-21-00 1:42a jdk1.3.0
JDK11~1 8 <DIR> 06-16-00 2:29p jdk1.1.8
0 file(s) 0 bytes
3 dir(s) 16,252.02 MB free

c:\>PATH
PATH=c:\java\bin;c:\tools;c:\windows;c:\windows\command
c:\>ren jdk1.2.2 java
```

If I wanted to switch to JDK 1.3.0, I simply do:

```
c:\>ren java jdk1.2.2
c:\>ren jdk1.3.0 java
```

and voila! new JVM. The drawback here is you need to know what the current one is, but that's simple using java -version or by process of elimination.
--
Joseph A. Millar

Q: I would like to know if it could be considered as dangerous to change directly the content of a bytecode.

Perhaps you'll think the question is strange... In my case, I'm just trying to replace the content of a known string by another string of the same length.

I 've already tried and it seems to work properly, but I'm not sure that it could be OK with different (all) JVM or for example, if the content of the bytecode could controlled by a checksum or something.
on JVM there's nothing about these kind of control.

Answer: Depends on your definition of "dangerous." If you mean "error-prone", then I would answer with a definite yes, especially if you are modifying the code itself... you need to be aware of all the jump targets so you can update them as necessary, and understand enough about the verifier that you can write provably safe code; otherwise a verifier will reject the class file.
However, a properly written resulting class file is perfectly valid and portable; that's the meaning of Java's much-hyped "binary compatibility".
There are also no checksums to worry about.
--
Chris Smith

Q: what is the difference between "C:\\" and "C:\\." ?

In the following codes. can anyone explain this ?

```
File dir = new File("C:\\" );
String [ ] files = dir .list();

File dir = new File("C:\\." );
String[] files = dir.list();
```

Answer: "." is used to refer to the current directory. For example, using the change directory command "cd ." changes you to the current directory, effectively doing nothing. "c:\\.\Files\image.jpg" is exactly the same as saying

"c:\\Files\image.jpg"

The code you gave should do exactly the same thing in both forms, to my mind - return a list of the files in the root of the c:\ partition.
--
Lloyd Colling
http://members.xoom.com/lcolling/

📋 Q: This is likely a very silly question. I need to create a .cab file, but I have no idea how to do it...

Answer: Microsoft has a tool for it. See
http://msdn.microsoft.com/workshop/management/cab/cab.asp

You can also get a shareware version of a Cabinet Manager (easier to use than the MS tool) from
http://www.micrologognet.com/
--
Jos

or you can download a free *.cab tool at:
http://home.t-online.de/home/lars.hederer/english.htm

📋 Q: Why we can reengineer Java byte code (.class file) back to Java source code? But why binary .exe file we are unable to do it? What is the significant difference?

Answer: AFAIK, Java byte code goes back to _some_ source, not to the original source. So, reverse engineering is limited.

> But why binary .exe file we are unable to do it? What is the significant difference?

(a) There is more than one way to do something using C++.
(b) There are highly optimizing C++ compilers.
(c) You won't get the original source anyway. See (a).

Imagine that your C++ code contains inline functions. The compiler is free do place the body of it replacing a call, or instantiate the function and provide a real call to it. First, there is no way to know when it is going to do one or the other. Second, with some code inlined, how to decide what _was_ an inline function and what wasn't?

So, the answer is: the distances between the levels of abstraction between byte code and Java source and between machine code and C++ source are significantly different. The bigger the distance, the less possible it is to recreate the source code.

Victor Bazarov

in java bytecode all the original variable names are kept. in an exe file smaller symbols are used.

Some .exe decompilers are good enough to convince a jury that the "original" source code was reconstituted. See the Microsoft v. Stac case for an example.

Java is easier, but C is still possible.

Q: I am attempting to write a program that recursively calls a method in order to go down a directory tree, read the information in each directory concerning their files, then write that information to a text file. Are there any built-in methods or programs in Java 1.3 to do this?

Answer: Here is a recursive example for finding a file.

```
static public String Findfile(String dir, String to_find) {
    int i;
    String results;

    String dir_list[]=(new File(dir)).list();
    for (i = 0; i < dir_list.length; i++) {
        File to_test = new File(dir,dir_list[i]);
        if (to_test.isDirectory()) {
            results = Findfile(to_test.getAbsolutePath(),to_find);
            if (results.length() > 0) return results;
        } else {
            if ((to_test.getName()).equalsIgnoreCase(to_find))
                return to_test.getAbsolutePath();
        }
    }
    return "";
}
```

Regards,...Ron

Q: I use the function inside of my class that has the same name in another class of my package. It seems working that program can be confused by this and it seems working better if I use this function within of class like this:

this.XXX();
How does Java distinguish the same methods in different classes?

Answer: Suppose you're inside a method and you'd like to get the reference to the current object. Since that reference is passed secretly by the compiler, there's no identifier for it. However, for this purpose there's a keyword: this. "This" keyword–which can be used only inside a method–produces the reference to the object the method has been called for. You can treat this reference just like any other object reference. Keep in mind that if you're calling a method of your class from within another method of your class, you don't need to use this; you simply call the method. The current this reference is automatically used for the other method. Thus you can say:

```
class Apricot {
void pick() { /* ... */ }
void pit() { pick(); /* ... */ }
}
```
Inside pit( ), you could say this.pick( ) but there's no need to. The compiler does it for you automatically. The "this" keyword is used only for those special cases in which you need to explicitly use the reference to the current object. For example, it's often

used in return statements when you want to return the reference to the current object:
source: "Thinking in Java" http://www.javafaq.nu/java/book/Contents.shtml

Q: My friend claim that garbage collectors do not collect int value since they are not created with new() method..
Is he right?

Answer: Programmers know about the importance of initialization, but often forget the importance of cleanup. After all, who needs to clean up an int? But with libraries, simply "letting go" of an object once you're done with it is not always safe. Of course, Java has the garbage collector to reclaim the memory of objects that are no longer used. Now consider a very unusual case. Suppose your object allocates "special" memory without using new. The garbage collector knows only how to release memory allocated with new, so it won't know how to release the object's "special" memory. To handle this case, Java provides a method called finalize( ) that you can define for your class. Here's how it's supposed to work. When the garbage collector is ready to release the storage used for your object, it will first call finalize( ), and only on the next garbage-collection pass will it reclaim the object's memory. So if you choose to use finalize( ), it gives you the ability to perform some important cleanup at the time of garbage collection.

source: http://www.javafaq.nu/java/book/Chapter04.shtml#Heading169

Q: In my program where I use finalizers to help GC to free the memory faster... But it seems that it is difficult to know when it happens. I tried to find it and see that even on the same machine my program runs differently...

Answer: You are right! Garbage collection happens differently each time because it based not on definite schedule but quite complicate alghorithms that take into consideration many factors such as CPU load, number of variables, memory size and so on.
Even developers of JVMs just guess when some process can start but not exactly. Since we have many JVMs, Java programmers, therefore, should avoid writing code for which program correctness depends upon the timely finalization of objects. For example, if a finalizer of an unreferenced object releases a resource that is needed again later by the program, the resource will not be made available until after the garbage collector has run the object finalizer. If the program needs the resource before the garbage collector has gotten around to finalizing the unreferenced object, the program is out of luck.
--
AP. (J.A.)

Q: Does Garbage Collection hang my program for a while?

Answer: Well, of course it somehow "hung" if you run your program on one CPU. Not in terms that it hungs until some GC-ing is over.
Usually well written GC runs in own thread, alongside to your Java program and does not more time than any other thread.
Your program will not wait until some point is reached in GC but rather wait some amount of time which the GC-ing thread allowed to take.
--
AP. (J.A.)

📑 Q: Could you please tell the advantages and disadvantages of having Garbage Collecting in Java?

Answer:
1. Although the programmer still allocates data structures, they are never explicitly freed. Instead, they are "garbage collected" when no live references to them are detected. This avoids the problem of having a live pointer to a dead object.
So, GC "keeps" eye on amount of memory allocated by program and tries to free memory of unreferenced objects in "good" (when your program does not consume much CPU) time.
You do not need to do free() operation like in C++. GC does it for you. Most of memory leaks happen due to bugs in Java itself rather than bad programming (happens also :-))
In a large application, a good garbage collector is more efficient than malloc/free
2. GC makes heap defragmentation (merges the small pieces of free memory into one big piece) that increases performance on the fly. It is difficult to do such thing easy in most of programs written on C++.
3. Your time! If you have fast enough CPU and good GC you will save a lot of time. Manual tuning of a millions pieces of code like malloc/free will take so much time that increases the cost of project dramatically!
Let say like this if you have slow CPU and small program then C++ with malloc/free is more efficient. If you have big one - rely on GC!
4. Security issue: Java programmers cannot crash the JVM by incorrectly freeing memory

Main disadvantage is that GC adds overhead that can affect performance. In some real time it is critically important that no GC-ing will run in definite periods of time..
--
AP. (J.A.)

📑 Q: Do I need to call Garbage Collector gc() explicitly? If not why then does exist this method?

Answer: Do not afraid, if you do not call gc() it will run anyway! You can be sure that GC-ing happens anyway...
Why does SUN provide us such method?
I see two reasons at least for having it:

1. Time critical applications. If you know that in some moment it is safe to run GC call it. Probably it will run immediately. Not always.
Anyway it can provide better distribution of GC-ing in time increasing GC activity in "safe" time

2. Test applications. It can help you to be sure that your objects will be collected faster than usually.

# Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

☐ IT Professional  ☐ General Computer  ☐ Design and Graphics  ☐ Linux  ☐ Freeware/Shareware

☐ Games  Intranet  Computer Hardware  Web Design  ☐ Computer Software

☐ UNIX  ☐ Web Software  ☐ Windows Software  ☐ Personal Finance  ☐ Programming Software

☐ Servers  ☐ C/C++  ☐ Powerbuilder  ☐ Perl  ☐ XML

☐ Java  ☐ SQL  ☐ Design/Graphics  ☐ Utilities  ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Miscellaneous - II

Q: Is it possible to know if a file changed?

Answer: The are some techniques:

1. Trust the last change date the OS maintains. The file could have had an Oscar Wilde pulled on it, insert a comma and take it out again so the last change date records a change, but it is not really changed.

2. Check the file size. Most chances will trigger a file size change.

3. Compute a digest, e.g. a 64 bit Adlerian checksum. If it is the same, chances are nothing really changed. The odds are quite astronomically small you could have a change and not detect it, but it is theoretically possible.

4. Compare the old and new files with .equals() after reading them into RAM, possibly in chunks.

I talk peripherally about these issues in the automatic file updater student project and the delta creator project in the student projects section of the java glossary.
--
Roedy Green
http://mindprod.com
http://209.153.246.39

Q: Can I access/use COM objects with Java without having to use Microsoft's J++?

I have a database app that has an API that uses COM objects and I would like to do all development in Java and JSP.

Answer 1: Two options I know of. JIntegra from www.linar.com or alternatively the open source product, Jacob from http://users.rcn.com/danadler/jacob/.

Phil

Answer 2: IBM also has a product called something like Bridge2OLE. An then there is always the option of doing it yourself plain vanilla JNI.

Jim S

Q: How can I calculate number of bytes occupied by an object in memory?

I have a object as follows:

```
public class obj {
   private String str = new String("ABCDE");
   private int i;
}
```

If I create ten objects of Obj class, how much memory space do they occupy?

Answer: The Runtime.totalMemory() method should help. However, because memory is being allocated and deallocated all the time, you should probably create a lot of the same objects and then divide the change in memory by the number of objects.

```
class MemoryTest {

   protected static final long COUNT = 100;

   public static void main(String[] arg) {
      long start, end, difference;
      Object[] array = new Object[COUNT];
      long i;

      Runtime.getRuntime.gc(); // let's hope the
      // garbage collector runs
      start = Runtime.getRuntime().totalMemory();
      for (i = 0; i < COUNT; i++) {
         array[i] = new Object();
      }
      Runtime.getRuntime.gc();
      end = Runtime.getRuntime().totalMemory();
      difference = (end - start) / COUNT;
      System.out.println("Approximately " + difference
               + " bytes used by 1 java.lang.Object with default constructor");
      }
}
```

WARNING: Strings are optimized to use as little memory as possible by reusing the same object if the strings are the same. You're going to have to do something special to test these out:
1. Make a program to write a bunch of random, same-length strings to a file (Let's say "strings.txt").
2. Use the code given above.
3. Before you calculate the start, open "strings.txt" with a FileReader or input stream.
4. In the for loop, read the a string from the file and put it in the array.
5. Close the file after the second call to the totalMemory() method.

If you use this code in any other form, don't declare or construct any objects between the calls to the totalMemory() methods. This could alter results.

this tip is from http://developer.java.sun.com/developer/qow/archive/107/index.html site

Q: Can anyone tell me where can I find the specification of the bytecode and the VM??

Answer: See http://java.sun.com/docs/books/vmspec/index.html

Q: My java app is getting rather large - is there any way to put the code in different physical files? In C I would do this and use the #include directive to reference these files. How do you do this with java?

Answer: The typical way to separate a large project is to put each class in its own file. When you get so many classes it's hard to keep them straight, categorize your classes into packages. Right now, for example, my current project, which we've been developing about a month, has about 150 classes in three different packages. There's no way I'd keep all that code in one file. ;)

As long as you keep all of your code in the same package, you don't need to do anything at all for the classes in different files to be able to find each other. Keep your source files in one directory, and they just do.

In case you care, there's an IDE called JCreator that contains a tool to take a single source file with lots of classes and automatically split it into multiple source files with a class per file. It does all of this for you.
--
Chris Smith

P.S. by John: JCreator you can find at http://www.jcreator.com/

Q: Would anyone out there have any recommendations/feedback on any good Java free profiling software?

Answer: If you are looking for a free product, take a look at HPjmeter from hp at http://www.hp.com/products1/unix/java/hpjmeter/index.html
I have found some trouble running it under win98 but it's quite useful.
--
xevi.

Q: If X is in the default package, any class should be able to find it without an import, right?

This one has me completely stumped :-/

Take the following two source files:

X.java:
public class X {
}

```
Y.java:
package test;
public class Y {
    X x;
}
```

X compiles, but Y doesn't. It fails with the error "Cannot resolve symbol X". A similar error is found under jikes.

If X is in the default package, any class should be able to find it without an import, right?

Answer: It's not an error: You don't import X anywhere, so the compiler _must_ assume it's in the same package. Since test.X doesn't exist, it cannot find it. The compiler does _not_ automatically use the default package, it defaults to the _current_ package.

To resolve it, add

import X;

to the Y.java file.
--
Tor Iver Wilhelmsen

Q: Do you guys know if it is possible to make executable files with Kawa, because I only see an option to make jar files (and even that is confusing as hell)?

On the side note, does anyone know how to make a little batch file to run the jar (so that I don't have to write stuff like java -jar run.jar)?

Answer: I suspect it doesn't allow static compilation - that's a much harder job. Executable jar files are the preferred approach in many cases anyway.

But here unusual receipt of malign of exe files pointed by Tom Almy! If you have a C compiler, you can make your own executables (which require the Java runtime be installed) using any Java design environment.

Assuming the jar file Foo.jar with main class Bar, here are the steps to making Foo.exe:

1. Compile the following program runner.c to runner.exe:
```
#include <process.h>
int main(void) { execlp("java.exe", "java.exe", "-cp", "foo.exe", "Bar", NULL); return 0;
}
```
2. Concatenate the runner.exe and foo.jar files:
```
copy /b runner.exe+foo.jar foo.exe
```

You are done!
--
Tom Almy

Q: How to prevent disassembling class ? If I develop a commercial product, I

don't want to give my sources to a JAD user...

Answer: There are three basic options:

A) Use an obfuscator (which will prevent casual disassembly, but may not stop determined reverse-engineers)

B) Use a client-server solution. The important code runs on a server controlled by you, and the user never sees the bytecode at all.

C) Make all your customers sign a legal contract promising not to disassemble the code, or they have to pay a huge fine.

P.S: JAD is extremelly easy to use Java Decompiler program written by Pavel Kouznetsov

Q: Does anyone do this & if so do you have a bat file which swaps between different JVM'es? Do I have to reboot NT once I change the path?

Answer: Yes you can. Just change the PATH environment variable in a DOS in WIN95-98 (command promt in NT) session, and in THAT session (and that session only), the version specified
will be used. To change it permanently you have to reboot.

Or you can consider using development tools like VCafe, JBuilder, VisualAge, etc. They usually have the function to set which JVM/Compiler to use. If you change the variable (in NT) in the Control Panel, System, you do not have to reboot.
Using Batchfiles will not work as a switch, however, since the variables are only valid during that batch session, so you would have to remain in that session and execute your command-line commands there to use the changed variables.

Q: Do I need to include all JAR files in CLASSPATH?

When you say java classpath = /directory1;. helloWorld do all *.jars in /directory1 get included? I have seen mixed behavior (sometimes they got included, sometimes not) with different jdk
installations (NT, Solaris, 1.2.2, and 1.3). I can't find any definitive answer to the question anywhere. Thanks in advance.

Answer: .JAR files are not included when you just specify the path to a directory. You must explicitly call out each .jar and .zip file as part of your classpath. It's a pain but you just gotta do it. I have not worked with 1.3 yet, so maybe they've changed something there, but if you follow this rule it will always work, regardless of the version.
The one exception to this rule is the /jre/lib/ext directory. This special directory is scanned when the JVM boots up and any .jar archives found in it are automatically added to the classpath. This is the only way that .jar files may be added to the classpath without explicitly listing them.

Q: How can I launch external applications from a Java application?

I mean what is the method that i got to invoke when i get an event like mouseclick to launch another application like photoshop or any other one???

Answer: Runtime.getRuntime().exec("notepad.exe");

that will launch notepad from withing a java app.

Q: Is there free tool for automatic drawing the UML diagram from a java source? (LINUX)

Answer: JVision generates UML from java source. The Linux version is free for non-commercial use.
See http://www.object-insight.com

Q: Again about classpath variable

I have a java program that I need to run through DOS, but my computer isn't set up for this, as I normally use textpad.
Anyway, I am a bit confused about what to use as my classpaths. I know that I need them, but do I need one for the jdk1.3 file, and then where do I have to run my program from, in that directory?
Could some one please explain, as I've been to the sun setting up classpaths page, and I didn't really get it...

Answer: What I usually do is this:

set my path to include the java\bin directory (the directory where java.exe and javac.exe are)

in dos the command is:

PATH=%PATH%;c:\java\bin\

then go to the directory where my project is and type the following to compile:

javac *.java

and to run:

java myproj (where myproj.class contains your main method)

or to run with the support of outside classes:

java -classpath QTJava.zip;skinRegion.jar;. myproj

QTJava is required for QuickTime
skinRegion provides extra tools for configuring your windows (and gui!)

generally if you don't need extra classes in your program (outside the ones that ship with java) you won't need to specify a classpath

Hope that helps!
--
Jonah Braun

P.S. by John: From my point of view classpath environment variable cause much troubles! I did not see any Java programmer that has not problem with this variable. I agree that this thing could be done in some better, more clear for understanding, way.
Unfortunately, SUN's documentation quite often is very poor and not friendly to us. Such good and big company could give more and better examples to technologies that they developed!
Often just give at least one example...

Q: I have had a poke around the collections lot and I can't seem to find any implementation of a FIFO stack, I've found a FILO stack but no FIFO.

Anyone one knows of one already implemented in the standard API for JDK 1.4 beta 2?

Answer: That might be because stacks *are* FILO structures. FIFO structures are usually called queues or deques (double ended queues). Anyway, LinkedList will do what you want, just add to one end and remove from the other.
--
Michiel

Q: We are looking for a Java code obfuscator to prevent our code from being decompiled. Can anybody recommend a good tool?

Answer, Part 1: Compilation to optimised native code is more efficient than obfuscation.
If your only deployment platform is Windows, have a look at our
Excelsior JET native compiler:
http://www.excelsior-usa.com/jet.html
Aside from the obfuscators others recommend, there is IBM's JAX obfuscator:
http://www.research.ibm.com/jax/
and you can try JProof at www.jproof.com free!

and Preemptive's DashoPro obfuscator:
http://www.preemptive.com/
--
Dmitry Leskov

Answer, Part 2:
If you are writing an Application, rather than an Applet, you can achieve greater protection by using both an obfuscator and using a custom class loader (Applets don't support custom class loaders without signing). There was an article in Dr. Dobb's Journal a few years ago that showed how to create a class loader (check www.ddj.com for the source code).
With a class loader you can have most of your program files encrypted. A "loader" program is used to load the encrypted files. Most of the code in the loader program can be placed in data arrays, and can also be encrypted.
The encryption key can be based on information extracted from the loader program itself, so that if someone tried to patch the loader program to print out the key, they wouldn't get the right value (it would still be possible for someone to customize the JVM so that it extracts the byte codes after the program was loaded).
The code contained in the data array could also contain additional data arrays containing encrypted code. Using two or more levels of encrypted code makes it difficult (but not impossible) for someone to find the key that will decode the rest of

the program files.
--
by Carl G.

Q: I am looking for a free downloadable Java IDE, is there such software?

Answer: Try out these IDE's:
http://www.borland.com/jbuilder/foundation
http://www.sun.com/forte/ffj/index.html
http://www.netbeans.org
--
Klaus Hartlage

Q: The javac compiler, which I downloaded for free as part of the Java 2 SDK, complains when I give it a file that has more than one public declaration.

It tells me, I have to put each public class in its own file. Is there any way around this?
I got the source code from someone using Codewarrior. Apparently, that compiler isn't so fussy. I would still prefer to use javac (the price is right). But I still have a lot to learn about Java.
Am I missing something simple? Some command line option or something?

Answer: By definition in the Java Language Reference, any source file must contain one and only one public class. You are allowed to include as many other non-public classes as you wish; but one and only one public class of the same name as the source file.

Filename: JavaTips.java

```
public class JavaTipOne{
    public static void main(String[] args){
      // do smth here
    }
}
class JavaTipTwo{
}
```
--
Dave Alger

Q: I want to put debugging code in my java classes that I can disable at a later date easily. Is there a way?

Answer: A simple solution would be to set a boolean flag such as
private static final boolean DEBUG = true;
in the class or classes you wish to debug. Then when you want to turn debugging off you can simply set the flag to false.

Your trace statements would then look like:
```
if(DEBUG){
    System.out.println("Debug statement");
    // other stuff
}
```

--
Rob

Q: I want to put a web site on CD-ROM in 8.3 file name format. The problem I have is that one of the files has the form "janorama.class" and when I change it to "janorama.cla" the javascript is no longer run.

I have tried changing <applet code="Janorama.class" to <applet code="Janorama.cla"
as well as changing the file extension.
Can any one tell me how I can accomplish this.

Answer: Use a JAR file! It has *.jar extension - just 3 letters!

Q: What does one type on the keyboard to produce the end of file character? I'm using forte and I tried ctrl z, but this is not break my code out of the while loop which loops until the EOF Character is encountered. Any ideas?

Answer: There is no EOF character. The different input classes report the EOF condition in different ways. For example InputStream.read() will return the value -1 if you try to read more data than is available.
On the other hand, BufferedReader.readLine() will return a null for an attempt to read more data when there is none left. What input class did you use?

Most operating systems report EOF condition to the program automatically when reading from files, but they have a hard time doing so for keyboard input. The OS usually provides a special character sequence you can type to indicate to the OS you are done, the OS in turn reports EOF to your program.
But the character you type is not an EOF character that you can test for in your code. (You didn't post your code so I have to guess this is what you tried, a common gotcha.)

On DOS and Windows9x, the character to type is control-Z. On Macintosh and Unix (and Linux) the character is control-D.
--
Wayne Pollock

Q: Were significant performance improvements made in the debugger between these versions?

When I start Weblogic under the debugger in JBuilder Foundation 4.0, it takes 9 minutes under 1.3.0, but only 1 minute using 1.3.1.
I couldn't see any mention of this in the release notes for 1.3.1 -- am I missing something?

Answer: 1.3.0 has a bug in Hotspot which makes debugging using the new APIs (which JBuilder uses) crawl. The workaround is to use the old JIT-enabled VM through the -classic option.

I don't know why 1.3.1's release notes wouldn't mention this.
--
Tor Iver Wilhelmsen

Q: I am in a situation where I have to dynamically generate classes. How do I compile and then call these classes? This is what I have figured out so far:

```
// the class will be a sub-class of GeneratedClass;
StringBuffer code = generateClass();

??? compile code ???

Class aClass = Class.forClass(myGeneratedClassName);
GeneratedClass obj = (GeneratedClass) aClass.newInstance();
```

Basically I need the code to fill in between the ???s above.

Answer: I would start by saving it to a file appropriately named. Then call
Runtime.exec("javac filename.java");

that should make a class file which you can then load if its in your classpath.
There might be an easier way, but I am not aware of it.
--
C. Lamont Gilbert

Q: Here is a small query. How to get % of memory usage in Java.

freeMemory() gives the long value of memory .
After certain % usage of CPU I want to take some action, so how to get this % usage
value .

Answer:
```
Runtime rt = Runtime.getRuntime();
long freeMem = rt.freeMemory();
long totMem = rt.totalMemory();
long usedMem = totMem - freeMem;
double percentUsed = (double)usedMem / (double)totMem;
```

Q: Is it possible to make application that goes into tray Icon area?

To put java application in tray Icon area, How can I do?

Answer: That area is a OS specific area. You will need the JNI...
A work around that you could use to stay away from JNI, is maybe to use a VB app
that sits in the tray for you then have your java app check a file called stats.dat (as an
example) every 2 seconds or so to see if the VB app has put some info in their that
your java app may need.

Its shifty but it works.
--
Vincent Panuccio

Q: Jikes is much faster, but are there any other reasons that jikes is better/worse
than javac? Any useful features of either one that make it better?

I've heard javac's incremental compiling does work well. Any truth to this or other
opinions?

Answer: Jikes compiles much faster but the resulting code goes at the same speed
and won't work with some debuggers like Jbuilder 4

Q: Here is a small query. How to get % of memory usage in Java.

freeMemory() gives the long value of memory .
After certain % usage of CPU I want to take some action, so how to get this % usage value .

Answer:
Runtime rt = Runtime.getRuntime();
long freeMem = rt.freeMemory();
long totMem = rt.totalMemory();
long usedMem = totMem - freeMem;
double percentUsed = (double)usedMem / (double)totMem;

Q: I'm trying to use the "exec" method in the java.lang.Runtime class, to control Netscape remotely. But I'm doing something wrong, because when I run the program, all the exec statements are just ignored, no matter what they were supposed to do.

The test program reads something like this:

```
public static void main (String[] args) throws java.io.IOException {
   Runtime R=Runtime.getRuntime();
   // R.exit(0);
   Process p1= R.exec("ls");
   System.out.println("nothing nothing");
   Process p2= R.exec("netscape -remote 'openURL(http://www.uiuc.edu)'");
}
```

the program prints "nothing nothing" and exits. If I uncomment the line R.exit(0) the program does exits...
So, I don't know what is happening.
I'm sure it is just a dummy detail you can tell me just by looking at the program.

Answer: The problem is that Java does not have the concept of a shell environment. That means it doesn't have a PATH environment variable, so it can't find what you are trying to execute.
Instead of "ls", use "/bin/ls". In other words, always include the fully qualified path to the executable.
--
Mitch

Q: Is there any way to change the classpath on the fly?

That is from within an application I want to be able to change the classpath so it includes a new jar file.

Answer: You have to write your own class loader, which is prepared to search the additional jar(s).
Well, you have to use a new *instance* of a class loader - but URLClassLoader should usually be good enough, so you don't need to write your own custom ClassLoader.
--
Laz E. Knights

Q: I'm new to Java programming. I don't understand some things in this code:

import java.awt.event.*;

import java.awt.*;

//MY QUESTION: Why do we need to use "import java.awt.event.*;"? We have already imported all packages from "AWT" by "import java.awt.*;" statement.
//when I tried running by deleting "import java.awt.event.*;", the compiler gave error like "WindowAdapter not found" "ActionListener Not found",

Answer: No you haven't. import foo.*; imports all the *classes* in the foo package, it doesn't say anything about the foo.bar package. Packages aren't *actually* hierarchical in the JVM's eyes - we just organise them in hierarchies to make them easier to deal with.
--
John S.

Q: Java debugging questions...

1. I m trying to use java_g / jdb pair in the windows environment. I couldn't find java_g in JDK 1.2 package. Was it replaced with something else ( java -Xdebug ??? )

2. I don't get any output when I do a Runtime.traceMethodCalls(true ). I was expecting all the function calls to be traced. Does this mean that the JDK 1.2 doesn't support this on windows?

3. To refresh my memory on remote debugging.
a) Compile the code using javac -debug
b) Run the class using java_g
c) Attach the jdb from teh remote machine.

Did I miss something?

Answer: I think they changed this at some point. I currently:

1) Compile everything using: javac -g
2) Run using:
java -Xdebug
-Xrunjdwp:transport=dt_socket,server=y,suspend=n,address=8000
YourClassName
3) Debug using:
jdb -connect hostname:8000

--
Peter

Q: We are developing our first Java application, and we encountered some problems with the Java memory management.

The operating system that we are using is mainly Windows (98, NT, 2K) but we are also planning to use Linux.

The problem with the memory is the following: our java application "eats" memory from the operating system, but the memory is never released. We are sure that our objects get garbage collected, but beside that, the java virtual machine keeps the memory.

We searched the web for documentation, and the ones that we founded are bad news for us!
The docs said that this is the normal operation of the virtual machine!!!

Please, tell me that I'm wrong and that the problem can be solved...

Answer: That's the usual behaviour of the JVM. The only possibility I detected to release memory to the operating system (checked with Windows NT + JRE 1.3) is to call the JVM with "java -Xincgc" and call System.gc() frequently.

If your problem is that you want to launch several applications you should consider running them all in one JVM.

Make sure you are not subclassing java.lang.Thread without actually running the threads -- this has been a known memory-eater.
Also, remember that any object in a Vector, Hashtable, etc. is "rooted" as long as its 'container' is...

There are some tools you can use to determine what kind of objects are being created, and what the heap looks like. A simple version is shipped with the JDK that is invoked with
java -Xrunhprof:<options>

Run java -Xrunhprof:help for details...
--
answers by Joern, Blair Wyman

Q: Does anyone know how I would go about mirroring a certain directory to another one using Java?

I want to do this in real time, so effectively when I'm copying a file into the first directory I want a copy to be created in the second simultaneously.

Answer: There is a Java program that synchronizes directories on a computer or on separate computers over a network.

http://www.kcmultimedia.com/jcase/
JCase is a tool that synchronizes directories on a computer or on separate computers over a network. It is intended for such chores as keeping a common directory current between a laptop and desktop computer. Windows users especially, will find JCase to be a faster, more reliable and more configurable alternative to the Windows Briefcase.
Although Java is a cross-platform language, JCase 1.0 will only be able to synchronize directories in compatible file systems. You will not be able to synchronize a Windows directory with a UNIX directory or vice versa. You will be able to synchronize two Windows directories or two UNIX directories.
--
John

Q: I am trying to do some profiling and need to know that at the end of my program, how many objects of a particular class have been Garbage collected.

How can I do this?

Answer: Well you could use a decent profiler.......

One way is also to add a static int to each object type you want to follow and increment it whenever the constructor is called, that way you get a count of the number of objects made(Which ultimately is near on the same as those collected as long as you don't have references being held all over the place).
If you necessarily need to know how many objects that have been garbage collected rather than created, move the increment to the finalize() method.
Do something like this:

```
class A {
    static int count;
    // ...
    // fields and methods
    // ...
    public void finalize() {
        count++;
    }
}
```

But keep in mind that finalize() might never be called, even if the object is GCd. You only know that finalize() will be called before the memory is reused.
--
Paul, Igor, Roger

Q: Hi, I'm looking into different profilers / memory / thread analyzers for Java and I'm wondering if any of you have had experience with any of them (love em / hate em). I'm looking for something that can profile well "locally" but also profile an application running under a web server servlet / JSP engine (specifically, IPlanet Web Server 4.1, possibly moving to 6.x later).

Experiences, thoughts, or feedback from vendors as well are welcome, and thanks.
--
John

Answer 1: I use JProbe's Profiler, version 2.8. It includes memory and performance profiling. I use it strictly for local profiling. I have found it to be very helpful in identifying both types of bottlenecks. It takes some time to learn how to use it optimally, but once you get the hang of it, the tool can be very helpful. I have not explored other products, primarily because at the moment we are still stuck at JDK 1.1.5. JProbe version 2.8 supports 1.1.7 and 1.1.8 as well as Java 2. JProbe 3.0 is actually out now but no longer supports JDK 1.1.x (at least there sales rep recommended that I not upgrade).

My biggest critique of JProbe is their documentation and have told them as much whenever I speak with a sales rep. If you fiddle with it long enough you can figure out what information is being presented in any given GUI window and how to make sense of it. But the application suffers from the classic weakness among insufficiently documented software: it has at least a sentence or more describing most GUI components, but fails to really put it all together effectively into a document that effectively communicates how to use it (maybe they think you'll buy some training or

tech support that way; I don't know).
--
Brian

Answer 2: Since I've been looking at two other products I thought I'd offer my thoughts on them so far for the benefit of those watching the thread:

1) DevPartner (NuMega): Really handy, debugs down to the windows kernel api level on windows. Excellent summary statistics at the end of a run, but not really good graphical display (that I've found yet at least) during a run. Excellent "drill into your source" capabilities and an oustanding display of "percent of allocations come from where".
Speedy "C-like" interface, and NuMega has excellent Windows products (BoundsChecker, etc) so I'm inclined to like them for that reason.
Downside: Each profiling tool is separate, with no integrated view of the whole thing. RemoteAgent failed to install on IPlanet box on first couple of attempts; need to try it with all applications shut down. This was disheartening. Had one nasty system-bye-bye NT crash bug at one point, so shows its "kernel debugger" roots in that respect; didn't happen a second time though. Nice support for profiling apps from the command line easily.

2) OptimizeIt. (VMGear) Easy and robust integration with IPlanet web server, which we needed in our environment, though the "uninstall" to IPlanet functionality could have been added as well.
Nice graphical display at runtime, some features only available then such as code drill-down (though a snapshot might make them available at other times -- haven't tried it). An API interface for turning it on and off -- cool, but haven't yet exercised it. Possibly able to use in production environment, but need to look into this feature more to see if it's available with IPlanet or simply for use on command-line type apps. Somewhat intuitive docs, though the API documentation could stand more description.
Some ugly Swing UI bugs running under Win98;
haven't tried them on NT -- these can be worked around but are annoying. Set-up of new projects somewhat more intuitive than DevPartner's, though DevPartner allows for easier and seemingly more extensive tweaking (including direct command line editing) once project is already set up.
Would like to see more summary information (total bytes allocated over the course of a run, etc), and jeesh fix the swing bugs. But a good job overall.
--
John Lockwood

# Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

| ☐ IT Professional | ☐ General Computer | ☐ Design and Graphics | ☐ Linux | ☐ Freeware/Shareware |
| ☐ Games | Intranet | Computer Hardware | Web Design | ☐ Computer Software |
| ☐ UNIX | ☐ Web Software | ☐ Windows Software | ☐ Personal Finance | ☐ Programming Software |
| ☐ Servers | ☐ C/C++ | ☐ Powerbuilder | ☐ Perl | ☐ XML |
| ☐ Java | ☐ SQL | ☐ Design/Graphics | ☐ Utilities | ☐ Handhelds |

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

# Networking

Q: I have difficulty understanding what is the difference between sockets and ports. They seem to be doing the same thing, receiving/sending data streams to other computers over network.

Answer: A port is a software address on a computer on the network--for instance, the News server is a piece of software that is normally addressed through port 119, the POP server through port 110, the SMTP server through port 25, and so on. A socket is a communication path to a port. When you want your program to communicate over the network, you have give it a way of addressing the port, and this is done by creating a socket and attaching it to the port.
basically, socket = IP + ports
Sockets provide acces to the port+ip

Q: Could you give me an example how to generate TCP/IP packets? I need to write my first network program.

Answer: There is no need to write such program. Java hides this implementation from you. Just use java.net.Socket or java.net.ServerSocket class and use appropriate constructor. Easy and fast!
If you want even faster, use the method setTcpNoDelay(true) :-)
--
AP (J.A.)

Q: We were trying to write a mail client using sun's javamail. I was wondering if there is a way to set the priority of the message.
Commercial mail clients does this by setting the X-priority:<num> field of the smtp header ( 1 means highest and 5 means lowest - I think the rfc allows much more than this ). Looking at the documentation I could not find any way. I was wondering if any of you have done anything similar.

Answer:  Look at MimeBodyPart::addHeader(String name, String value);

You can add any headers allowed by the RFC spec. :-)

📑 Q: ServerSocket or DatagramSocket? What is better to use in my applications?

Answer: Both of them are good. It depends on task you do.
DatagramSocket is designed for UDP and ServerSocket for TCP. TCP is more reliable in terms that it ensures delivery, UDP not. But UDP packets are much smaller and have no such big headers like TCP packets have.
If it is smaller then there is more probability that it arrives faster. UDP used mostly in areas where you do no mind about retransmission. You just do not need this, because information become obsolete. Sound, for example.
It is better to make small pause or repeat last packet than to play some piece that was send first but received second.
It is up to you to take decision. If you do not mind that can lose small piece of data - use UDP. Otherwise use TCP.
--
AP. (J.A.)

📑 Q: Why Java networking classes has no Ping method? It is number one utility!

Answer: I have no answer to this question. Probably because it uses some native interfaces.
The good news is that solution already exists and it is free.
You can take free implementation from our site here:
http://www.javafaq.nu/java/examples/files/pingicmp.zip
The program is written by programmed by M Isabel Garcia and Oscar Fernandez.

📑 Q: How can I send/receive SMS messages via GSM phone with Java?

Answer: Use Kvanttisms - Java class library. The library contains classes for encoding/decoding SMS messages in PDU format and communicating with a GSM terminal through a serial link.
This library is distributed under Apache Software license, so you will not have problems to use in most of cases.
Take it here:
http://sourceforge.net/projects/kvanttisms/
--
AP. (J.A.)

📑 Q: Is it possible to ping an email address? I need to validate entries made on an order entry form, and as part of the validation process I would not only like to validate the formatting of the email address, put also if the address to authentic.

Answer: No, it is not. If it were possible, spammers could use this to detect valid E-mail addresses en masse. A loop could be written that would detect all active AOL E-mail addresses over a period of days.
You can easily "validate" an E-mail address as to its form, even sometimes ping the domain at the right of the address (but not always), but you cannot verify that the E-mail address is real without actually posting a message to it. This is why so many E-commerce sites do just that.

In the past there were some UNIX network services that listed all valid E-mail addresses on a server, but they have largely been discontinued for the same reason -- abuse.
--
Paul Lutus
www.arachnoid.com

**Q:** I wrote small network program using TCP socket. I see that often it does send immediately the data I am trying to send. I have heard that it is because of "nagling". What does it mean - nagling?

**Answer:** "Nagling" is network transmission algorithm named after John Nagle. This algorithm specified a means of dealing with what he called the small packet problem. This problem arises when we try to send for example one character over the net.
Sending of just one byte causes adding additionally ~40 bytes to packet header. So if we send one page of text with small packets it can cause incredible traffic. Overheading will be 4000 % compare to real data.
John Nagle algorithm avoids such trouble and sends packets when they become big enough. In some applications you want to send one byte anyway, for example if you have heart beat that supervise your connection. To disable "nagling" use setTcpNoDelay(true) from java.net.Socket
--
AP. (J.A.)

**Q:** can someone please post a snippet of Java source code that will issue a ping to a specified IP and either return whether successful or not, or round trip time.

**Answer:** A solution related to that has been posted a few days ago, an alternative way works for me via Runtime.getRuntime():

```
import java.io.*;

import java.net.*;

class Ping {
  public static void main(String[] args) {
    BufferedReader in = null;
    try {
      Runtime r = Runtime.getRuntime();
      Process p = r.exec("ping 62.2.78.245");
      if (p == null) {
        System.out.println("Could not connect");
      }
      in = new BufferedReader(new InputStreamReader(p.getInputStream()));
      String line;
      while ((line = in.readLine()) != null) {
        System.out.println(line);
      }
      in.close();
    } catch (IOException io) {
      System.err.println(io.toString());
    }
  }
}
```

--
Linda

**Q:** I need telnet implemented in Java. I found a few packages on the net. Which is best?

Answer: It is difficult to say which really best one is until you run for years and get all troubles. I like the implementation from Matthias L. Jugel and Marcus Mei/ner because it supports SSH. It is very important! Ordinary telnet connection exposes everything to any person who uses simplest network tools.

SSH provides secure connection. This library is free and available under GNU General Public License.

--
Leo

📋 Q: I am trying socket level programming through firewalls. Could somebody tell what Http tunnelling is and how to achieve that using Java 2?

Answer: As an aside, how do you request your proxy server to fetch a page from the net?

http://developer.java.sun.com/developer/technicalArticles/InnerWorkings/Burrowing/index.html

📋 Q: How can I let dial a phone number with a modem in a Java app.? Is there a way without a System.exec() call and without any M$ classes?

Answer:  You could use javax.comm to do it manually via the serial port and the good old AT command set on the modem. Alternatively, you might look at JTAPI, but that might have its own problems and a lot of overkill.

📋 Q: Does it possible to have two thread running at the same time which reads from the same socket. If a message is received, does both threads then receive it?

Answer: Two threads can read data from the same socket input stream, but they WON'T each get copies of the same data. They'll each get separate parts of the message.

📋 Q: how can I get an IP Adress in the class InetAdress? The constructor is private, so I can't use it. I want to call the method getName () to get the domain name out of an IP Adress.

Answer:  It is not necessary to construct something :-)
Just do it like this:
for example: String hostname = InetAddress.getLocalHost().getHostName();

📋 Q: I'm converting an old java client/server program which is based on raw byte stream heavily into new one which requires utilizing object streams. But if I open input/output object streams on both sides this blocks system and won't proceed...

Hi,
I'm converting an old java client/server program which is based on raw byte stream heavily into new one which requires utilizing object streams. But if I open input/output object streams on both side this blocks system and won't proceed.


ObjectInputStream in = new ObjectInputStream(socket.getInputStream());
ObjectOutputStream out = new
ObjectOutputStream(socket.getOutputStream());

Answer:  Upon opening the ObjectInputStream, the constructor blocks to read a header from the stream (doing who-knows-what). Now, what happens is, both your client and server open the InputStream... and happily wait forever for the other side to send them the header they want. Deadlock guaranteed!
The solution is simple: open the ObjectOutputStream first on at least one side, but better on

both sides (usually, symmetry == good). Problem solved
:)
You are trying to keep two streams going at once independently, right.
That means you need at least two threads at each end.

📄 Q: Are there classes available to read and write to an RS 232 port or does this require using native code?

Answer:  See the Java Communications API.

📄 Q: Is there a System property that'll tell me the (TCP/IP) machine name? If not, how do you go about it?

I need to display all the machine's TCP/IP addresses to the user. This requires a call to InetAddress.getAllByName(), which requires the machine name. You *can't* pass null or "localhost" to this method. Well, you can pass "localhost", but that only gets you the localhost addy itself-
127.0.0.1. Not useful.

Answer:  Try this: Working that out, I tried

```
String hostname = InetAddress.getLocalHost().getHostName();
System.out.println(hostname);
InetAddress[] ads = InetAddress.getAllByName(hostname);
for (int i=0; i<ads.length; i++) {
    System.out.println(ads[i]);
}
```
by Michiel

📄 Q: Hi, I am new to JTapi (java telephony). I am trying to run a sample code provided with the JTAPI specification. However I get an error message such as :

Can't get Provider:
javax.telephony.JtapiPeerUnavailableException: JtapiPeer: DefaultJtapiPeer could not be instantiated.

Answer:
As with many Java packages (most notably JDBC) java.telephony is only an API -there is nothing behind it.

You have to pay someone for an implementation.

The point is that if you use the API then it will run with anyone's implementation.

📖 Q: Is there any way to connect through the proxy?

I'm connected to the web behind a proxy. When i try to connect to servers outside they time out and give an exception.
Is there any way to connect through the proxy?

Answer: Try this at the command prompt:

java -Dhttp.proxySet=true -Dhttp.proxyHost=<ProxyHostNameHere>
-Dhttp.proxyPort=<ProxyPortNameHere> <myclass>

<ProxyHostNameHere> is to be replaced by the proxy host.
<ProxyPortNameHere> is to be replaced by a proxy port number (i.e. 8080)
<myclass> is the java class that you are running.

You can also set these function through the System class. Check the API for setting properties.
Hope that works.
--
Dan

📖 Q: Socket - My problem is the main class that spawned these threads cannot kill the thread if it's blocked in a read.

I'm making a program that launches some threads that connect to the net and do some stuff. I'm using a blocking read (TCP/IP BufferedReader.readLine) because I've been recommended by a few people that it was the better way to do it.
My problem is the main class that spawned these threads cannot kill the thread if it's blocked in a read.
Is there a thread function I could use? I tried use BufferedReader.ready() to make my read not blocking
(BufferedReader.ready() returns true only if the the stream is readable otherwise loop so there is no actual blocking) but my problem with that was that BufferedReader.ready() doesn't throw an exception if the other side disconnects so I'm left looping infinitely. I'm really stuck here so if anyone can give me a strategy (I don't need it spelled out to me just general "I used this function type help) I'd really appreciate it.

Answer 1: I've found the only way to unblock a (Socket) read-blocking thread is to close() the socket from another thread. That'll throw an IOException in the read() call.
--
Michiel

Answer 2: Another method that seems to work is to set a socket timeout to something moderately short (like 10 seconds). The blocked read will throw an exception when the timeout occurs. Then the thread can inspect the state of a variable (set by another thread calling an appropriate method) to see if it should terminate.
This approach also makes it trivial to implement a longer inactivity timeout if desired and gives the network thread an opportunity to do some maintenance work if it wants.
--
Peter

📖 Q: I get the impression that only read operations from a Sockets InputStream throw an Exception if the opposite Socket has closed the socket. Writing to the Socket's outputStream works fine...

Is there a way to detect if what I write into a Socket's outputstream is actually still being received by the other side?
Or will I have to check that I actually get a response via the InputStream, and if not, try to open a new Socket and resend the request again?

Answer: You can try calling flush() after writing the output, but there's no guarantee that you'll get an immediate exception. The underlying TCP/IP software may go through timeout and retry logic before giving up.
That's probably why you'll usually find out about a broken socket when waiting to read. You should rewrite client so that it sends a request and waits for a response. If it doesn't get a response, it should try to make a new connection and do the request/response thing again.
If it doesn't work the second time, it should give up for the time being. Perhaps it really makes sense that one has to verify 'by hand' that requests go through.
--
Duane Morse

Q: ... I imagine, I'll have to cut my file into small datagrams.

I just made a little chat program (client and server) and I would like to add the possibility to transfer files.
The client and the server are communicating with TCP/IP using a socket.
I imagine, I'll have to cut my file into small datagrams. Is it necessary ? Has someone an idea or a link to source code of such a function ??

Answer: No. If you were using UDP/IP (DatagramSocket and friends) then it would need to be split up. The point of TCP is to avoid this need. Simple send the entire file in a stream.

Q: I would like to send a large amount of data through a Java socket. I currently use the following (pseudocode):

out = BufferedOutputStream(socketOS, 8092)
loop{
data = byte[100000]
out.write(data)
}

My network connection can send 1 MB in about 5 seconds via FTP, etc., but the above process takes over a minute to send 1 MB.
Is there a way to even come close to my connection's capabilities?

Answer: Once of the main problems with this buffered stream, is that its write method is synchronized. As we all know, synchronized methods can take up to 6 times longer than normal methods to execute. For this reason might I suggest taking the source for BufferedOutputStream, taking out the synchronization, and recompiling as a new class? I would guess this would increase time considerably. Give it a try!
More: one of the unoptimized points is the memory-allocation...
data =byte[100000]... in the loop.
This means, each transferred byte is allocated in memory. That's very bad (very slow)!!!!
--
Chris Shorrock http://www.tantalus.com

Q: Tomcat and packages...
I'm a bit confused about how packages should be organized in Java.
If I work for mycorp, should my classes go into a package named com.mycorp.myapp?
If I do, it seems I must recreate this directory structure in Tomcat in order to use the classes:

webapps/myapp/WEB-INF/classes/com/mycorp/myapp ?

Answer: Correct. When you state e.g. package org.foo.cool; in a java source file the compiled file must reside in a directory structure that mirrors this:
org/foo/cool (your source files should/must also reside in a similar directory structure)

The CLASSPATH must be set to the top directory of the packages.
To start an application whose main method is in the org.foo.cool.CoolApp class place the sources in e.g.
/home/foo/javaapps/org/foo/cool/ and start the app with
java -classpath /home/foo/java org.foo.cool.CoolApp
(actually java -classpath /home/foo/java org/foo/cool/CoolApp also works)
--
Nils O. Selasdal

Q: Where can get the more JSP source?

Answer: try

http://www.jspin.com/

http://jsptags.com/

http://www.jsptut.com/Getfamiliar.html

Q: our project is currently serializing a lot of stuff. The aim was/is to be able to start up quicker when you (only?) had to de-serialize, instead of initializing all those objects.

My question: is this really a start up time winner or the opposite?

Answer: My experience with serialization is limited, precisely because it took so much time to deserialize - and so I abandoned using it.

Well, the only way to tell for sure is to try it both ways and see which one is faster.

If I had to guess, I'd say object initialization is faster. But I don't know what you code does to initialize its objects.
Does it just set some fields to default values? Or does it set its fields to values returned from a website accessed over a 9600 baud modem? It makes a difference.

If you want your program to start up fast, the way to do that is:

1) Write your program in the most straightforward way, not thinking about performance until it's done and feature complete.

2) Check to see if it's fast enough. If so, stop.

3) If not, use a performance-measuring tool to see where it's spending its start up time; optimize those routines.

Any other methodology is invalid, in that it is as likely to slow your app as speed it. (I'd guess that's what you did.)
--
Marshall Spight

Q: I've tried in every possible way I could find to determine my actual internet IP address with Java. It just doesn't work. My computer is connected through a router that connects to my ISP. This causes the InetAddress.get* methods to return only an intranet ip address.

InetAddress.getAllByName("<router ip address>") doesn't work either, it returns the intranet address of the router.

Answer: If you can't see "your" IP address using getAllByName(), then it is not "your" IP.

If the following code cannot resolve your Internet IP address, then you don't have a direct connection to the Internet:

```
// NetTest.java

import java.net.*;

class NetTest {

    public static void main(String args[]){
        try {
            String hostName = InetAddress.getLocalHost().getHostName();
            InetAddress[] ipList = InetAddress.getAllByName(hostName);
            System.out.println(hostName + ":");
            for(int i = 0;i < ipList.length;i++) {
                System.out.println(ipList[i].getHostAddress());
            }
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
}
```

My example run:

pl-alpha:
192.168.0.2
228.232.218.126

The first is my fixed intranet address; the second is my temporary log-on address.

If the NetTest (getAllByName) is returning only private IPs for you (192.168.*, 10.*) it sounds like the router is providing NAT and Masquerading so that the PC doesn't really have a real IP address--it's like you're behind a firewall. There's no way for clients on the internet to reach your server.

On the other hand, if you can configure your router to forward packets on a particular port to your intranet address, then you can just bind to the local address and the rest will come out ok.
--

Paul Lutus

[www.arachnoid.com](www.arachnoid.com)

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

| | | | | |
|---|---|---|---|---|
| ☐ IT Professional | ☐ General Computer | ☐ Design and Graphics | ☐ Linux | ☐ Freeware/Shareware |
| ☐ Games | Intranet | Computer Hardware | Web Design | ☐ Computer Software |
| ☐ UNIX | ☐ Web Software | ☐ Windows Software | ☐ Personal Finance | ☐ Programming Software |
| ☐ Servers | ☐ C/C++ | ☐ Powerbuilder | ☐ Perl | ☐ XML |
| ☐ Java | ☐ SQL | ☐ Design/Graphics | ☐ Utilities | ☐ Handhelds |

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

# Operational Systems & Java

Q: Is anyone aware of any way to determine OS type, version...

Answer: There is in the java.lang.System
try first:

```
Properties p = System.getProperties();
System.out.println(p);
```

to see what properties you can get(CPU among others)..
you then can:

```
String prop = System.getProperty("aproperty(the CPU among other)");
```

Q: Does anyone know if there is an easy way to invoke UNIX shell scripts from a Java application.

Answer:  There is!
execute a shell and then write the name of the shell script on the shells stdin stream.

```
Process child = rtime.exec("/bin/bash");
BufferedWriter outWriter = new BufferedWriter(new
OutputStreamWriter(child.getOutputStream()));
outWriter.writeLine("filename");
outWriter.flush();
```

Q: We need to authenticate a user on the local Linux system before he/she/it can log on to the application and begin taking over the world. Since the Linux system uses shadow passwords, we must have a method to retrieve the password and authenticate the user in the user shadow database. Once authenticated, this user can work with the application.
How it can be done in Java?

Answer: It can be done, it is already done!
Here if you are interested in interfacing C with Java in Linux (the JNI Solution):

http://cscene.org/CS4/CS4-04.html
--
John

📰 Q: Is there a way to differentiate the enter key on the main keyboard from the enter key on the keypad?

Answer: I don't think so, they're both represented by the character code 10.

📰 Q: Is there any way to manage Systray Icons with Java? I didn't even find anything in the Microsoft packages...

Answer:  You must create a C/C++ program that call directly the
NotifyIcon API to display/manage the Icon in the Systray. Then
call the program from Java with JNI.

📰 Q: Currently I'm running two operating systems on one machine, Windows 2000 and Windows ME...

Currently I'm running two operating systems on one machine, Windows 2000 and Windows ME.
Windows ME runs on default, Windows 2000 is on D drive and ME is on C drive. I tried to add the JDK directory to the classpath but there isn't a autoexec.bat on the D directory, but there is one for C.
Should I just create a autoexec.bat for D?

Answer:  Go to Settings/Control Panel/System/Advanced/Environment Variables... and
edit your CLASSPATH variable if exists or add a new one.

📰 Q: I need to be able to run a shell script from java, by doing a host call to unix. I am currently trying 'Runtime.exec' to do this. It says it runs it, but doesn't actually successfully complete the command (which is a file delete).

I need to be able to run a shell script from java, by doing a host call to unix. I am currently trying 'Runtime.exec' to do this. It says it runs it, but doesn't actually successfully complete the command (which is a file delete).
A few questions I have about this:

1. Can I trace it or something to see why it isn't working?
2. How can I get the 'return code' from the shell script?
3. Will the java procedure wait for the shell script to execute, or does
it run a seperate thread concurrently?

Answer:  > 1. Can I trace it or something to see why it isn't working?

Runtime.exec() returns a Process object that you can get information from.

If the script is written with sh or some derivate thereof you can do 'set -x' in the script to trace its behaviour. For other scripting languages there is likely a similar mechanism. To see the trace in your Java program, read from the error or output streams of the process (Process.getErrorStream() or Process.getOutputStream()). It's possible that the script is already printing a message to its error stream that you aren't seeing, indicating why it isn't working.

Answer: > 2. How can I get the 'return code' from the shell script?

Call Process.exitValue().

Answer: > 3. Will the java procedure wait for the shell script to execute, or does it run a seperate thread concurrently?

The program runs concurrently in a separate *process*, not a thread within the JVM. To wait for the process to finish, call Process.waitFor().

📑 Q: Are not-initializable classes and methods like System.out.println() and Math.random() "synchronized" ?

Answer: I think they are synchronized. Simple observation: did you ever see once that printout was broken into two pices by another printouts? I mean for example if you do:

In 1st thread: System.out.println("1234567890");
And in 2nd thread: System.out.println("something else here");

it never will be broken like:

12345
something else here
67890

Even if Sun didn't write about it explicitly, we can see that it is synchronized or at least behaves like synchronized that is the same for our real life.

📑 Q: I have a little problem with the JVM (1.3) under Linux (SUSE 7.1). Every time I try to run a java application that includes a GUI, while the JVM starts I get a few warning messages saying that the JVM can't locate some fonts. The application continues to run but non of the fonts included in my code are used, instead I get the same font in every part of the GUI.

Answer: I get the same response. This is because the fonts specified in the code are not resident in the place where the JVM looks for fonts. They may actually be on your Linux machine somewhere - there are hundreds of fonts available to X windows in Linux.

There are only 5 fonts which are guaranteed to be available in every java environment:
Serif, SansSerif, Dialog, DialogInput, and Monspaced.
There are other fonts available to you on your system, of course. But the problem with using any fonts other than the guaranteed five is that you cannot be assured your user will have the fonts you have specified in you code, on his/her system.

📑 Q: I want to know, if I have more than one CPU on my machine, is it possible to bind a JVM to a particular CPU?

Answer: That's operating-system dependent..... Java has no control over this. For example, under Solaris, look at pbind(1M)...

📑 Q: How can I pass a string to the command line(DOS) ? Also i want to capture

the output given by the command line in a string.

Answer:  Try this out:

```
// works for DOS
String cmds[] = new String[2];
cmds[0] = "dir"; // replace with "ls" on UNIX
cmds[1] = "c:"; // replace with "/" on UNIX

// execute the command
Process pro = Runtime.getRuntime().exec(cmds);
// wait until it's done executing
pro.waitFor();

// what did the process output from the Input pipe back to
// this process (okay, who named this stuff)?
InputStream out = pro.getInputStream();

// output it (really slowly)
int i;

while ((i = out.read()) != -1) System.out.println((char) i);
```

📋 Q: How can I take a program that runs in a DOS shell and send the text that comes from the shell program into a Java program where it can analyzed, etc.?

Answer:  From a command line, use a pipe (with the "|" symbol):

c:\> dosprogram | java JavaProgram

In the Java program, read the text from System.in:

```
public static void main(String[] args) throws IOException {
    int nLines = 0;
    BufferedReader in =
    new BufferedReader(
    new InputStreamReader( System.in));
    for (;;) {
        String line = in.readLine();
        if (line == null)
            break;
        nLines++;
        System.out.println(nLines + ":" + line);
    }
}
```

📋  Q: If there is a way to run a java program by just typing the name in UNIX. I mean instead of typing for example "java Main" just type "Main" and run the program. And how to implement that in a makefile?

Answer:  Write a script that runs the program and put it in your path. For instance:

```
#!/bin/sh
java BlahBlah
```

Call this whatever you want, mv it to your /usr/local/bin directory, then just type it at the command line and BlahBlah will be run.

Q: Are there any Java libraries for executing Linux commands?

Answer:  Try java.lang.Runtime.exec(). E.g.

Runtime.getRuntime().exec("xterm");

Note if you want to use shell builtins or shell features like redirection you need a shell, e.g.:

Runtime.getRuntime().exec(new String[] {"/bin/sh",
                    "-c", "ls / 2>&1 | tee ls.log"});

Q: I'd like to know how to know which operating system java application is running on.

Answer:  You could try using the system Properties.
e.g.

Properties prop = System.getProperties();

String osString = prop.getProperty( "os.name" );

Q: I would like to know how my Java program can catch when someone sends a "kill" to my app in Unix or does a Ctrl-C in windows?

In Unix there is atexit() function that handles this type of situation. Is this possible in Java ?

Answer:  Starting with 1.3 there is Runtime.addShutdownHook(). This is for cleanup only.
from API: "A shutdown hook is simply an initialized but unstarted thread. When the virtual machine begins its shutdown sequence it will start all registered shutdown hooks in some unspecified order and let them run concurrently. When all the hooks have finished it will then run all uninvoked finalizers if finalization-on-exit has been enabled.
Finally, the virtual machine will halt. Note that daemon threads will continue to run during the shutdown sequence, as will non-daemon threads if shutdown was initiated by invoking the exit method. Once the shutdown sequence has begun it can be stopped only by invoking the halt method, which forcibly terminates the virtual machine.
Once the shutdown sequence has begun it is impossible to register a new shutdown hook or de-register a previously-registered hook. Attempting either of these operations will cause an IllegalStateException to be thrown.
Shutdown hooks should also finish their work quickly. When a program invokes exit the expectation is that the virtual machine will promptly shut down and exit. When the virtual machine is terminated due to user logoff or system shutdown the underlying operating system may only allow a fixed amount of time in which to shut down and exit. It is therefore inadvisable to attempt any user interaction or to perform a

long-running computation in a shutdown hook."
more read here: http://java.sun.com/j2se/1.3/docs/api/index.html

Q: Is it possible for a minimized window in MS Windows task bar to start blinking? In any IRC client a minimized window starts blinking when a new message appears in it. Is it possible in Java?

Answer: I doubt very much that it is possible with pure java; you would have to use some native code to achieve that. Maybe what you could try though, is to play around with the icon used in the title bar of the frame. I haven't tried this myself - and in any case I'm running linux, but it might be possible to have two icons and to switch them around at some rate using the Frame.setIconImage(Image) method. This will only affect the icon itself - not the whole window, but it's better than nothing.
--
Allen Wallis

Q: How do I find the list of all system properties?

Answer:  do smth like this:

```
Enumeration list = System.getProperties().propertyNames();
   while(list.hasMoreElements()){
      System.out.println((String) list.nextElement());
   }
}
```

Q: Is it cls-like command in DOS (CMD) window? I would like to clean the screen the user sees?

Answer:  Unfortunately there is no command as "cls", but try this instead:

```
for    (int i=0; i<25; i++)  System.out.println();
```

It will print 25 empty lines and they will move current printouts up

Q: Why my program does not give the address of the local machine on one PC and give on another?

Answer:  As long as you have TCP/IP installed, you should at least get 127.0.0.1

Q: Can applet corrupt my registry file?

I have written a java applet for displaying my genealogy. It has worked fine at home. My brother tested it on my web site with no problems. BUT, the third person I asked to take a look at it reported, that though it worked fine, when he exited the page, his system locked up and he had to reboot. After the reboot he said his registry was corrupted and had to be restored.

Has anyone seen anything like this? if so, do you have any suggestions as to probable cause and fix?

Answer:  Any time you don't shutdown properly, your registry can be corrupted, even if the program that did the freezing was not even using the registry.

The registry is one of the stupidest ideas ever conceived in computer science. It puts

ALL your eggs in one basket.
by Roedy Green
For the JAVA GLOSSARY see http://mindprod.com/jgloss.html


📖  Q: Does anybody know how to find out the complete path of a system default browser (if there's such) from a Java stand alone app?

Answer: Under Windows you can 'invoke' the default browser calling 'start filename.html', under Unix is a lot more complicated because nobody can assure you that a browser exists and the user is under X....
--
Davide

📖  Q: ...The problem is that, after the file is uploaded to the server, every end of line is placed with ^M character...

I have written a servlet which uploads a file from client(Win32) to server (Unix). I have used DataInputStream to read the file in bytes and FileOutputStream to write the file to the location in the server.

The problem is that, after the file is uploaded to the server, every end of line is placed with ^M character. This problem is faced only when I upload to Unix Server from Windows O/S.

Answer: This is normal, I think. A line ends with a single return symbol in Unix and ends with a return symbol and a switching line symbol in Windows. The editor in Unix shows "^M" when it meets the switching line symbol.
So to fix this bug, you can remove the switching line symbol in the end of lines when you upload files from Windows to Unix.
Or Use a BufferedReader to read the source file (line by line) and a BufferedWriter to write the destination file. That way the source line separators will be discarded and the correct separators (according to the target OS) will be used.
If you only have an InputStream to start with, use InputStreamReader to convert it to a reader.
--
Jorge

📖  Q: I have a server written in Java that I would like to start up as a NT service...does anyone know how to do this?

Answer: Check this site:
http://www.kcmultimedia.com/smaster/

I read there:
"ServiceInstaller is a FREE utility that makes it easy to install programs as Windows NT Services, including pure Java applications, without manually editing the registry!"

📖  Q: Does any one know how to kill or stop a process through java?

Answer: It must be a process you created, and there are rare occasions when this will not work.

Process p = Runtime.getRuntime().exec(path);

// later ...

p.destroy();

--
Paul Lutus

Q: I'd like to determine the free disk space in a platform independent way. So far, I've found no Java way to do so...

The best I've been able to come up with is running the UNIX "df" utility (which is available for a number of non-UNIX platforms, too, such as Win32 and OS/2).

Answer: Samizdat Productions Releases JConfig 2.1.1 JConfig is a class library that extends the core Java API. It lets you work with files, web browsers, processes, file types, and other system-level items in a much more advanced manner than that provided by the standard Java class libraries. A list of JConfig's features is given below.

JConfig is free for most freeware and educational projects, and it now comes with the complete Java and C++ source code!

** Download JConfig here:
http://www.tolstoy.com/samizdat/jconfig.html
Here's a partial list of JConfig's features, by category:
    Files:
Enumerate the user's disk drives, and obtain extended information on files, directories, volumes, and filesystems: their icons, creation dates, version information, mount points, and more...
    Web Browsers:
Launch a file or URL in the user's Web browser...
    Video Monitors:
Enumerate and get information on the user's video monitors: bit depth, bounds, and more...
    External Processes:
Create external processes, send basic commands to external processes, obtain the PSN or HWND of a process you created, and enumerate the currently running processes...
    File Types:
Find applications associated with a given file type, find applications by name, and convert between Windows file extensions and Mac creator/file type codes...
--
boruvek

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

☐ IT Professional  ☐ General Computer  ☐ Design and Graphics  ☐ Linux  ☐ Freeware/Shareware

☐ Games  Intranet  Computer Hardware  Web Design  ☐ Computer Software

☐ UNIX  ☐ Web Software  ☐ Windows Software  ☐ Personal Finance  ☐ Programming Software

☐ Servers  ☐ C/C++  ☐ Powerbuilder  ☐ Perl  ☐ XML

☐ Java  ☐ SQL  ☐ Design/Graphics  ☐ Utilities  ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++, SQL, XML and another languages programming topics YOU want to hear about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR interests and start getting the stuff you want today. The best free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Servlets & Servers

Q: The first thing is that i don't know how to do to create Servlets ... no more precisely, I don't know how to compile it !!

It says that it doesn't find the packages (javax I think) ...
Plz, could someone tell me exactly what I should put on my environment variables (Class_path, path, java_home etc...)

Answer: Well, I also started to learn about servlets. I recently downloaded JDK1.3.0 and Apache Tomcat 3.1.0 .
I set the following paths.

CLASSPATH = %TOMCAT_HOME%\webapps\examples\WEB-INF\classes\work;
  (my working directory)
TOMCAT_HOME = C:\JDK\tomcat;
  (or wherever you installed tomcat)
PATH=%TOMCAT_HOME%\bin ;
  appended tomcat bin path to the path statement)

I also copied the servlet.jar found in the tomcat bin directory to both:

C:\jdk\jre\lib\ext\
C:\Program Files\JavaSoft\JRE\1.3\lib\ext\

I hope this helps....

Gregory Pedder

Q: I do not see constructors in servlets? Am I missing something? How can servlet run without constructor?

Answer: There is no need to use constructor in servlets for you because servlet are similar to applet in a way it is running. The creation of servlet instance is a Server responsibility. You just need to use init() method.

This method called first when servlet is loaded into servlet container...
--
AP (J.A.)

Q: Where can I find online books about servlets?

Answer: I have on the site free books link collection. Doing search for "servlet" I got 5 books. Please check here:
http://www.javafaq.nu/indexu/search.php4?keyword=servlet
--
John

Q: When can I use System.exit() in servlets?

Answer: Never! Depends on server you run on... Security exceptions on good server or full shut down for simpler one.
--
John

Q: Why Servlets are better than CGI scripts?

Answer: Generally, every new client's request to CGI script starting a new process. With servlet web server starts just a new thread that is much cheaper in terms of time for starting up, memory and CPU consumption.
JVM uses just one process.
This answer is right for "standard" CGI. Now appear new modules for Perl that uses the same approach as servlet does. Anyway, writing large programs on Java is much easier than on Perl, at least for me.
Java has also very rich API.
--
John

Q: I was writing and testing some servlet program. Now I decided to remove it from our web server.
But it is still there causing confusion to another developers I am working with.
How can I unload my servlet explicitly?

Answer: It depends on server you use. Often you can unregister your servlet somehow.
Easier way to make your servlet empty (remove all code), compile it and copy class file to server.
Many servers reload automatically a new class if they see it was changed. You will still have your servlet on the server but it will do nothing and nobody will complain...
--
Peter

Q: Any simple server I can download to run my java servlets...

Answer:
Try Resin, www.caucho.com
Try tomcat from http://jakarta.apache.org - pure Java,
servlets 2.2, good stuff! Tomcat is good. It's not fast, but it's very easy to setup and good solution for development. Since most heay-duty servers do implement SUN specification, one can migrate application in no time.
--

DG

Jigsaw is also good but doesn't support the Servlet 2.2 spec

<URL:http://jigsaw.w3.org/>

Both of these are open source with BSD-like license.

There's also Jetty http://jetty.mortbay.com/ Artistic license, Servlet 2.2 spec compliant), GNU Paperclips http://www.tapsellferrier.co.uk/gnupaperclips/ GPL, claims to be moving towards Servlet 2.3 spec compliance) and also vqServer  http://www.vqsoft.com/vq/server/  which is free but not open source.

Finally many of the commercial servers can be downloaded free for trial periods.
--
Simon Brooke http://www.jasmine.org.uk/~simon

Q: Can someone point me to some tutorial/example/text book that covers the subject of receiving on a server side a file upload from a browser sent as with content-type set to multipart/form-data.
Thanks.

Answer: There are free Java classes that will do this for you. Here's one I wrote:

http://users.boone.net/wbrameld/multipartformdata/
If you still want to write your own, you can use mine as an example.

or here:
http://www.servlets.com/cos/index.html
There you can download package which contains also examples, how to do it.
--
Walter Brameld

Q: How can I get access to Cookie set at the Client?

Answer: The following code should access a cookie on a client. It reads in all the cookies on the machine. And checks there name for whichever one you are looking for.

```
Cookie[] cookies = request.getCookies();
for(int i=0; i < cookies.length; i++) {
    Cookie thisCookie = cookies[i];
    if (thisCookie.getName().equals("Cookiename")) {
        // Do whatever you want with the cookie....
    } else {
        // cookie doesn't exist...
    }
}
```
The Cookie class is in package javax.servlet.http.Cookie

Q: I'd like to provide to the showDocument() method of an applet the URL of a CGI program...

I'd like to provide to the showDocument() method of an applet the URL of a CGI program with including a certain number of (URL-encoded) parameters to this URL (which is the same as doing a GET HTTP request).
What is the maximum size I can give to this URL ?

Answer:  If I remember exactly it something around 240 for many servers. Maybe less, but not more!!!
1000000%
I read it  last year in "Java Servlet Programming" from O'Reily.

Q: I am experimenting a Java server application. This program has worked well

It did start on the Red Hat Linux 6.0 server, but it does not open the socket, in other words, it cannot communicate with the client applet on the Linux.  On this Linux server I have installed every components and all of them were running at the experiment time.  Why does this server application communicate with the client applet only on the Linux?  Does anyone give me a suggestion?

Answer:  Take a look at your port number. If it is under 1024, it is a protected port number and non-privileged users cannot touch it on Linux or any other Unix-system.

Q:Where Can I find a server to try my servlets?

I am creating a client/server application. I don't run my own server
and my ISP won't allow me to install and run applications from their
server.
Does anyone know of anywhere (preferably FREE) that will allow
me to use server side Java? Any help is GREATLY appreciated.

Answer:  http://www.mycgiserver.com/

Q: Hi, I am using servlets. I need to store an object NOT a string in a cookie. Is that possible? The helpfile says BASE64 encoding is suggested for use with binary values. How can I do that???

Answer: You could serialize the object into a ByteArrayOutputStream and then Base64 encode the resulting byte []. Keep in mind the size limitations of a cookie and the overhead of transporting it back and forth between the browser and the server. Limitations are:
     *  at most 300 cookies

     * at most 4096 bytes per cookie (as measured by the characters that comprise the cookie non-terminal in the syntax description of the Set-Cookie2 header, and as received in the Set-Cookie2 header)

     * at most 20 cookies per unique host or domain name
For more details please refer to RFC 2965.

Q: Hi, I want to send a POST request, but I can't find such functionality in the servlet API, how can I do this? Must I implement this with a socket connection to port 80?

Answer:  A servlet can do anything a standalone Java application can do. It doesn't need anything beyond what the java.net package already provides. You can use an

httpURLConnection to POST to a server program like a servlet or CGI script:

```
// Create a string with the parms you want to post  and convert it to a byte array. You may need to
// pass the values through java.net.URLEncoder.encodeURL()
// if they have embedded blanks or special characters

String parms = "a=10" + "&b=20" + "&c=30";
byte[] bytes = parms.getBytes();

// Create a URL pointing to the servlet or CGI script and open an HttpURLConnection on that URL

URL url = new URL(TARGET_URL);
HttpURLConnection con = (HttpURLConnection) url.openConnection();

// Indicate that you will be doing input and output, that the method is POST, and that the content
// length is the length of the byte array

con.setDoOutput(true);
con.setDoInput(true);
con.setRequestMethod("POST");
con.setRequestProperty("Content-length", String.valueOf(bytes.length));

// Write the parameters to the URL output stream

OutputStream out = con.getOutputStream();
out.write(bytes);
out.flush();
// Read the response
BufferedReader in = new BufferedReader(
        new InputStreamReader(con.getInputStream()));
  while (true) {
     String line = in.readLine();
     if (line == null) break;
     System.out.println(line);
  }
in.close();
out.close();
con.disconnect();


--
Phil Hanna
Author of Instant Java Servlets
http://www.philhanna.com
```

Q: I am writing an application, using Java Servlets, which requires me to set and read cookies. It works okay, but fails if I set the domain of the cookie to something other than the current server.

I am writing an application, using Java Servlets, which requires me to set and read

cookies.
Using the servlet API, and the javax.servlet.http.Cookie class.
I created a new cookie, and added it to the http response, using its addCookie() method. It works okay, but fails if I use the setDomain method, on the newly created cookie, to set the domain of the cookie to something other than the current server.

Answer:  I suspect that is not a legal operation for any browser to accept a cookie that has a domain inconsistent with the source of the cookie.
by William Brogden

Q: I am working on weblogic server 5.1 with MsSQLSERVER7 i am able to load the driver but it says unable to get socket connection

I am working on weblogic server 5.1 with MsSQLSERVER7 i am able to load the driver but it says unable to get socket connection.It says connect to the MSSQLSERVER's host and port no.
How do I get these name and value.

Answer: The MS Sql Server's host is usually the name or ip of the server that run SQL Server, if you know the IP (ping <nameoftheserver>), put the IP in it, it will be faster, for the Port number, sincerely I don't
remember the standard port, but look into the SQL Server documentation and you will find it.
by Davide

Q: Whenever I compile my servlets it always says "can't find package javax.*" even though I downloaded the JSDK. Where the JSDK files so it'll find that package?

Answer:  There are no classes in the javax.* package. There are classes in javax.servlet.* and javax.servlet.http.*, but neither are really related to javax.* -- importing javax.* won't affect them. You should import the packages that you really want to use!

Q: I have a list of html links on a web page, and I want to be able to call a servlet based on what the user clicked on...

I used:

<a href=/servlet/SomeServlet">

for each link, and I want the servlet to display information based on what the user clicked on. If the user click on an applet link, I want the Servlet to print, "You just clicked on an
applet", etc.

My question is, how do I send information to a servlet, based on what html link the user clicked on? I know i can use getParamaterValue() for getting information off of forms, but I'm not sure how to do this with html tags.

Answer:  Change the link to:

<A href="/servlet/SomeServlet?click=apple">link</A>

In the servlet, use request.getParameter("click") to retrieve the value.

Give each link a unique "click" value and that will tell you what was clicked.
B Russell

📋 Q: I'm looking for a Java HTTP server framework that can be used and modified for a project I'm going to be working on.

My boss asked me what the equivalent in Java was that IIS was in the Windows world or Apache was in the UNIX. I've looked at Jigsaw, but am wondering if anyone out there knows of other resources...open source would be great. Thanks in advance for any input.

Answer: There are other pure Java web servers, like the Apache PicoServer, Jetty (http://www.jetty.org ???). Perhaps you could take one of those to write a http block for the Apache Server Framework avalon (http://java.apache.org/framework ) Another tip is to search sourceforge for web servers.
http://www.sourceforge.net
--
Glen Shelly

📋 Q: I am currently running Microsoft's IIS server. I have the Java Virtual machine installed on this system. What else will I need to run my servlet. Do need the Apache web server instead? Will IIS support Java servlet?

Answer: You will need a Servlet Engine to run your servlets.
You can use either Allaire's JRun (http://www.jrun.com) or Tomcat (http://jakarta.apache.org).
Both of them work with IIS.
--
Madhusudhanan Challur

📋 Q: How can I avoid browser caching?

In an applet, I call the function showDocument to send a GET message to one of my servlets, which sends me back a PDF file dynamically generated by this servlet.
My problem is : the browser (in this case IE) always sends me back a PDF file with the same content.

Answer 1: There are a few possibilities to avoid this problem.
A simple workaround is to add a random number URL variable with the request, such as "ranx" with a value of the milliseconds since midnight. This will make the request unique and, hence, avoid browser caching.
by Duane Morse

Answer 2: There are two other response setHeader attributes you can set:

response.setHeader("pragma", "no-cache");
response.setHeader("Cache-Control", "no-cache");
response.setDateHeader("Expires", 0 )

Just in case.
Doug Schwartz

Answer 3: // for Answer 1 and 2 please go to yesterday's tip.
When you generate the PDF file, make sure you set the header to tell the browser that the file will expire at a certain time. The browser should not cache the response past the given time. The Java code looks something like this:

```
import java.text.SimpleDateFormat;
import java.util.Date;

SimpleDateFormat dateFormat
= new SimpleDateFormat("EEE, dd MMM yyyy - HH:mm:ss z");

response.setHeader("Expires", dateFormat.format(new Date()));
```

The format of the date is very particular.
---
Scott Gartner

Q: Wich free software can I use to debug Servlets?

Answer: What you need to do is use Jakarta-Tomcat to run your servlets. Then, bring in all the tomcat jars into your "required libraries".
Then set up your "project run" to actually run the Tomcat class. If you run it in debug mode, you should be able to put breakpoints in your servlets.

We did this at a project I was on with weblogic and EJBs, and Jbuilder 4 does it with tomcat, so I'm assuming you can do this as well.\
--
Trever M. Shick

Q: Is tomcat an EJB container?

Answer: No. It's only a servlet/JSP container. However, you can certainly use it with an EJB container, and you can get another free EJB container such as JBoss.
--
Chris Smith

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

# Threads

Q: Can anyone please explain the difference between the two types: green threads and native threads?

Answer: Green threads is thread mechanism implemented in JVM itself.
It is blind and can run on any OS, so actually all threads are run in one native thread and scheduling is up to JVM. This is disadvantageously for SMP systems, since only one processor can serve Java application.

Native threads is a mechanism based on OS threading mechanism. This allows to use features of hardware and OS. For example,there is IBM's JDK for AIX that supports native threads. The perfomance of applications can be highly imploved by this.

Q: I use myThread.isAlive() method to test that thread is still alive. But sometime even if it indicates that some thread is still running I am getting errors when try to use that thread. What is wrong?

Answer: the method is Alive makes you sure that thread actually "was" not "is" alive at the moment when you used it. I think it could be better if SUN rename it to wasAlive() and did not confuse people. This method just checks once when you use it and does not keep eye on what happens after.
Probably the thread will exit right after that.
The method activeCount() does the same: it counts all running threads. This information becomes obsolete immediately next moment!
enumerate() method is another example.
All those methods are useful only for statistic collection, nothing else. Avoid using them for other purposes!
To know when the thread exits use join() method.
--
AP (J.A.)

Q: What are Thread Groups useful for? I do not see big reason to have them in Java API...

Answer: From the beginning Thread Groups were added for security reasons. Assumed, that not trusted code from third parties will run in dedicated Thread Group. Of course it can not affect the rest of program. Those threads from external companies were not allowed to start, stop and suspend your threads. Then this model is changed and we have no such protection now.
Now the Thread Groups will let you a little bit easier manage your threads. You can manage many threads simultaneously rather than each separately. For example you can set maximum priority level, call suspend(), interrupt() on all threads in your group. Also giving good names will help you to easily track and debug your program.

M.

Q: What was wrong with stop() method for threads? Why did SUN deprecate it? I really need it...

Answer: stop() method was very cruel. It stopped the thread without giving any possibility to save the data.
It was not possible to predict when it will be applied. Even using finally method could not help - stop() interrupted it as well.
Really it is still possible to use it. Simple solution:

use boolean variable, for example:
boolean letStop;
and add two functions - prepareForStop() and doMyStop().

Method prepareForStop() must save all your data and make sure that you run all your code in thread, then set letStop to true.
Method doMyStop() will call stop() if only letStop value is true, otherwise do nothing (or inform you and you call prepareForStop()).
Keep in mind that stop() is deprecated.
--
AP (J.A.)

Q: I tried to use destroy() method on my thread. But nothing happens! I double checked my code and sure it is Ok. What is problem?

Answer: The problem is that SUN did not implement this method. So, it exists, but does not destroy anything. Why?
Sun says that if you use such powerful method your program will crash/hang later, maybe even sooner. I believe to it because they left stop() method even, although in deprecated form, but not destroy().
--
E

Q: I know that exist Win32 and POSIX threads. Are Java threads different on different OSs? How can it affect my program?

Answer: Although Java designed to run on different OSs it does not mean that your program will run in the same way, especially if you have threads.
The problem is that Java runs on some OS, not Java OS. And underlying OS implementation is different on different OSs. UNIX uses POSIX threads - Portable Operating System Interface. This is standard for UNIX and approved by IEEE. Microsoft as usually made it "slightly" different that causes standard incompatibility -

Win32 threads.
And implementation is different. For example, on Windows you have just limited number of priority levels, I do not remember how many, but it was something ~10-20. In UNIX - thousands!
So, if your program uses priority comparison of threads, let say priority 23453 and priority 23454 then it will be no difference on Windows. Be aware about it.
--
AP (J.S.)

Q: I move from C to Java now. My friend says that I still do my programming in an old way since do not use threads. But I do not see any reason to use them in my program. Could you advice where I should use them and why?

Answer: You are right! Having threads functionality in Java does not mean that you MUST use them.
A lot of programs work better without any threads. For example, editors, calculators. Some programs run better and more reliable if they are multithreaded, for example servers, programs with repetitive, independent tasks. The final decision of course will be done by you. Multithreaded programs much more difficult to debug, avoid dead lock situation. Because threads are running often independently you can not predict exactly what happens in your program right now. Often multithreaded programs behave differently each time you start them and it is difficult to repeat problem exactly when you debug your program. So, if you have simple logic, possibly no need to run multiple tasks - try to avoid the thread usage.
--
JA

Q: I know that it is possible to create threads by two different ways. What is essential difference? What is better and when?

Answer: that's right! We can create a new thread by subclassing Thread:
public class MyBestThread extends Thread {...

Also we can do it by using Runnable interface. The difference is that by implementing Runnable we change just run() method keeping the possibility to subclass something more useful...
The difference between these two methods is really small and often makes no sense.
--
AP (J.A.)

Q: Why suspend() and resume() methods were deprecated?

Answer: These two functions were included into API with purpose to let do Garbage Collecting (GC) and debugging.
For these two task the usage of suspend() and resume() is meaningful. But for all other things they are real disaster because suspended thread can work as a controlling thread and hold a lock. Suspending of this thread can cause hanging of program and really can be used effectively. That's why they were deprecated.
But you can still use them if you sure that do right thing. Nobody knows how long SUN is going to keep deprecated methods and classes. Quite possible that they never will be removed.
--
AP (J.A.)

📋 Q: Is it true that ThreadDeath exception was "secret" and accidentally exposed to public?

Answer: Yes it is true. The original implementation was not intended to show it to us. But due to some mistakes of SUN programmers it got out and now it is a part of API. Warning: "An application should catch instances of this class only if it must clean up after being terminated asynchronously. If ThreadDeath is caught by a method, it is important that it be rethrown so that the thread actually dies." from API reference for java.lang.ThreadDeath.
--
AP (J.A.)

📋 Q: I know how to start thread - to run run() method... But seems there is no exit() method. How do I stop my thread? Will it run forever?

Answer: Nothing can be run forever :-). In Java there is no exit() method. It was made intentionally. The idea was that thread will exit when run method returns. So, when you reach return in your run method - your thread will exit and die.
If you need to keep your thread working you do not let it reach the return :-)
--
AP (J.A.)

📋 Q: When should I use a daemon thread? Why would I use one instead of a regular thread? What is the purpose of daemon threads?

Answer: Any Java thread can be a daemon thread. Daemon threads are service providers for other threads or objects running in the same process as the daemon thread. For example, the HotJava browser has a daemon thread, named Background Image Reader, that reads images from the file system or the network for any object or thread that needs an image.
Daemon threads are typically independent threads within an application that provide services for other objects within that same application. The run() method for a daemon thread is typically an infinite loop that waits for a service request.

When the only remaining threads in a process are daemon threads, the interpreter exits. This makes sense because when there are only daemon threads remaining, there is no other thread for which a daemon thread can provide a service.

To specify that a thread is a daemon thread call the setDaemon() method with a boolean parameter that is true. To determine if a thread is a daemon thread use the accessor method isDaemon().

📋 Q: Hi, Is there somebody who can tell me why my thread sleeps longer then I told him to do...

I have a thread that has to sleep for 60000 millesec. But every 4, 5 minutes it sleeps for 61000 millesec.? I have to built an application that get the time every minute, but with this sleep I can't trust the java threads.

So can somebody tell me what is going wrong???

Answer: Really JDK never give you warranty that will wake your thread after XXX ms.
You can be sure only

that your thread will not be waked up before!

For good timing you should take another, better for real time perfomance, VM.
For example PERC from Nemonics.com or something else...

📘 I have created a program with a main method that instantiates and starts three threads, the first two of which are daemons. Why daemons does die when normal thread die?

Answer:  Because of nature of daemon threads. They are alive if exists at least one "normal user's" thread. Otherwise they die immediately

📘 Q: Does anyone know if there is a way for two threads to find each other if they are started in two different JVM?

In other words, I start a thread in one JVM, then I want to PipeWrite to another Thread that is doing a PipeReader. Any help would be appreciated!

Answer: Use of Piped streams is only supported inside the same JVM. If you want IPC, you have to use sockets or xfer data using files. I would recommend the sockets approach, it has the added advantage of working not only between process, but across the network as well.

Use of other IPC mechanisms, like shared memory, pipes, mail boxes, etc, are not supported by the JVM core. You could always roll your own native code, but why bother when you have access to sockets?

RMI is another possibility but it all depends on what you want to do. I'll let the RMI experts talk about this option.
And CORBA, if you want to get real fancy.
--
Joe
first published on comp.lang.java.* newsgroup

📘 Q: I would like to ask a question about garbage collection of Thread Object.

When will a Thread Object be garbaged collected?
When the reference count to it becomes zero, or when it enters the "Dead" state, i.e. the run() member function terminates?

Answer, part1: Since Thread is also an Object, it will only garbage collected when the reference count is zero.  You may think it is quite non-sense. the thread is useless when it enter "dead" state. why not garbage collect it?
That's because the thread object itself may contain some other useful information even the thread dead , e.g. the result of the execution of the thread. Thus, it is not sensible to do garbage collect when the reference count is not zero.
--
Anthon
P.S. Important ad! Except when object A holds a reference only to object B and object B holds a reference only to object A. Both reference counts are non-zero, but both objects are eligible for garbage collection. Which is why few, if any, modern VMs use reference counting to determine eligibility for garbage collection.
Jim

Answer, Part 2: You can consider a Thread object as a normal Object for garbage collection purposes if you consider this one rule: A running thread will not be garbage collected.
That is, a normal running thread is a root object, so the Garbage Collector will not attempt to mark it for collection. When the thread is not running, though, normal Garbage Collection rules apply (i.e. total # references == 0 then collect).

To get a thread with different behavior,
see the Thread.setDaemon(boolean bState) method.
--Brian

Q: I was until recently using the MS version of the Java API and I was using Thread.stop() when I switched to SUN I see it is deprecated.

I understand why, in fact it was causing a problem that has been solved by taking it out.
However, I do need to halt the thread from running without making it sleep.
How can I do this?

Answer: One simple way to kill a thread is to have a boolean stop variable with a method to set it to true, like so:

```
endThread(){
   stop = true;
}

run(){
    while(!stop){
          //all run code goes here
    }
}
```

If the rest of your code is okay with the last loop finishing before run() ends and the Thread dies, this works great as it is simple and can't cause any bad states. If not you can add more conditional statements just before you would affect something your stopped thread shouldn't.
After a certain point adding conditionals would become too inefficient and I'm sure there's a solution to that but I don't want to figure it out right now.
--
Ben

Q: How to overcome failures in threads?

How can I monitor multiple threads? I need to find what my threads do.
I have found thread.isalive() but I have a pool of threads and they are always alive.
What I need is more specific info.

I have also found thread.getclass() but I didn't understand how to use it.

For example, if I write:
```
public class test implements Runnable .....{
    private int data;
    public int getData() { return data; }
```

}

Is it possible to use:

test test1 = (test)Thread.currentThread().getClass(); ?

I have multiple threads and I need to create a model which can handle the problems with threads. For example if one thread fails I need to interrupt that thread and complete its work.

Answer: Try to use such kind of inspecting pattern:

A. you have an inspector class Inspector which extends Thread and has a method:
public void
handleProblem(ObjectContainingAllInformationsAboutTheProblem o) {
   // here you get the information about the problem
   // from the object o and use them to handle this
   // problem.
}

B. You have a Pool of thread objects MyThread each of them implementing two methods :
public void inspect(Inspector i) {
   if (thereIsAProblem) {
     i.handleProblem(new
     ObjectContainingAllInformationsAboutTheProblem(...))
   }
}

C. Your main program creates an instance of Inspector and informs it about the thread pool (gives it a reference to the pool) and starts Inspectors thread.

D. In its run() method, Inspector will periodically inspect all MyThread objects calling a method such as the following :
private void inspectAll() {
   // threadPool is the reference to the TreadPool...
   for (int i=0 ; i < threadPool.numberOfThreads ; i++) {
     threadPool.getThread(i).inspect(this);
   }
}

Well, I hope it helps you :-)

Anyway you can extends the inspectors interface to gather more information and to be able to monitor your threads in the way you want and when you want (if you don't inspect the Threads you don't get any information)
--
Michel

Q: Anyone know how I can stop a thread without using the deprecated stop() method?

Normally I would just have a "while(alive) {...}" loop, and some "stop()

{alive = false;}" method.
However, this time all the thread does is call one method, which it sits in until it finishes, and I want to stop it in certain situations. The method isn't one I've written, so I can't alter that.

Answer: I always construct my threads so they can be killed at any time. An outline of the code (this is just a sketch) is the following.

```
Mumble mumble = new Mumble(args);
Thread t = new Thread(mumble);
t.setDaemon(true);
t.start();


private class Mumble implements Runnable {
    private Thingy args = null;

    public Mumble(Thingy args) {
        this.args = args;
    }

    public void run() {
        try {
            process();
        }
        catch (InterruptedException e) {
            // cleanup processing and fall through to method end
        }
    }

    private void process() {
        while(true) {
            // do something with args
        }
    }
}
```

Note that calling interrupt() on a Thread doesn't just instantly generate an InterruptedException.

The interrupt does happen instantly because the thread calling it is executing, not the thread executing the process() (unless you are on a multi-processor machine in which case this may not be true). The point of having interrupt() is to be able to stop something like a long running method to complete.

The only very difficult problem lies in blocking I/O operations, which do not throw InterruptedException and probably should. Blocking I/O will block until either the underlying stream is closed or the operation is complete... so be careful about using large blocking I/O operations for cancellable I/O.
--
Allan Wax

Q: I want to have a program that does something every 5 seconds. Is there a

method called pause(5000) or wait(5000)?

The point is that I don't want to create a thread simply for this purpose. Thread.sleep() can have the current thread sleep/idle for certain period of time, but can the job be done without threading?

Answer: You can use Thread.sleep( 5000 ) without creating a new Thread object because the method is static. You can look at the example below:

```
public class TestSleep {
    public static void main ( String args[] ) {
        System.out.println( "Starting" );
        for ( int i = 1; i < 101; i++ ) {
            System.out.print( i + "\t" );
            if ( i % 10 == 0 ) {
                System.out.print( "\n" );
                try {
                    Thread.sleep(1000);
                }
                catch( java.lang.InterruptedException Ie ) {
                    Ie.printStackTrace();
                }
            }
        }
    }
}
```

--
Mikkel Bundgaard

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Sound & Multimedia

### Do you know that Version 1.3 of the JavaTM 2 Platform
includes  a powerful new API for capturing, processing, and
playing back audio and MIDI (Musical Intstrument Digital Interface) data.
You can read here more: http://java.sun.com/j2se/1.3/docs/guide/sound/index.html

I'm doing a small console java app, and want to know what the function is to make the pc speaker beep....

Answer: Toolkit.getDefaultToolkit().beep();

Q: I would like to burn a CD from within my code, that way I won't have to waste so much time making illegal copies of music. Is there a convenient way to do this in Java?

Answer: Unfortunately Java doesn't provide any API for this :-)

Q: I use System.out.println("\007") to make a beep. It works fine with java.exe version 2.2.2_007, but doesn't work when I use javaw.

It's said that java and javaw are exactly the same but apparently they have some differences.
How I can use System.out.println("\007") to make a beep with javaw?

Answer: System.out.println sends a character to the standard output stream, which is redirected to nowhere when you use javaw. Same would happen if you ran your application under UNIX and redirected standard out to /dev/null. I'd guess that your code wouldn't work at all on a Macintosh.

To solve this problem, do what you need more directly, for example using java.awt.Toolkit.beep instead of a non-portable kludge with sending special ASCII codes to an output stream.

Chris Smith

📖 Q: Just wondering if Java has classes and methods supporting joysticks? Is there any sample code working with joysticks some where?

Answer: Sure it does. It's part of the Java 3D API. There is an InputDevice interface. It is described as follows:

"InputDevice is the interface through which Java 3D and Java 3D application programs communicate with a device driver. All input devices that Java 3D uses must implement the InputDevice interface and be registered with Java 3D via a call to PhysicalEnvironment.addInputDevice(InputDevice). An input device transfers information to the Java 3D implementation and Java 3D applications by writing transform information to sensors that the device driver has created and manages. The driver can update its sensor information each time the pollAndProcessInput method is called."

The trick is finding a driver. :)

http://sourceforge.net/projects/j3djoystick/
http://www.j3d.org/utilities/sensors.html

Jim S.

📖 Q: I've coded this class so I can call Util.playSound("beep.wav") in my application (Win 2000/jdk1.4.0) to play the beep file contained in sound/beep.wav in my .jar file.

About 1/3 of the time, this just doesn't produce any sound at all. What am I doing wrong here?

```
class Util {
    public static void playSound(String name) {
      AudioClip clip =
myapp.app.newAudioClip(myapp.app.getClass().getResource("sound/"+name));
      clip.play();
   }
}
```

Answer: I've create several Java 1.1 Applets and have noticed that short AudioClips are often heard only part of the time on Windows-based platforms. I traced the problem to the poor system clock resolution (they are only accurate to about 55 milliseconds) and the way AudioClips are buffered and mixed (I looked at the way the unsupported sun.audio.* classes are implemented).
I was hoping that the in newer versions of Java (e.g., 1.4, or in versions using newAudioClip) this problem would have been fixed. Perhaps your problem resides somewhere else. One possible work-around is to make your AudioClip files a little longer (e.g. pad them with 55 ms of "silence" on both ends).
And for 1.3 and later, you should be using javax.sound instead
--
Carl G.

# Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

☐ IT Professional   ☐ General Computer   ☐ Design and Graphics   ☐ Linux   ☐ Freeware/Shareware

☐ Games   Intranet   Computer Hardware   Web Design   ☐ Computer Software

☐ UNIX   ☐ Web Software   ☐ Windows Software   ☐ Personal Finance   ☐ Programming Software

☐ Servers   ☐ C/C++   ☐ Powerbuilder   ☐ Perl   ☐ XML

☐ Java   ☐ SQL   ☐ Design/Graphics   ☐ Utilities   ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu


# String, text, numbers, I/O I part


Q: I need to detect whether a user enters a capitalized letter. I'm using the following code so far:

```
// The keyboard
BufferedReader br =
new BufferedReader(new InputStreamReader(System.in));

String input = null;
do{
  try {
    input = br.readLine();
  } catch (IOException ioe) {
  System.out.println("IO error");
  System.exit(1);
}

//check input for uppercase characters (ignore the
//fact that it might be numbers, assume valid input)
...
```

I have searched the API, in particular, the string function for a way to do this but no luck.

Answer: Try this:
```
if(input.equals(input.toLowerCase)) {
  // no upper case letters
  // (untested code)
}
```

Additionally, if you need to test individual characters, char comparisons are perfectly fine.

```
char blah = 'a';
if(blah >= 'A' && blah <= 'Z'){} // if the char is uppercase then...
```
--
Jeff

P.S If you want that code works for all languages, not only English use please do it
like this:
```
if (Character.isUpperCase(blah)) { ... }
```
--
Dr. Harald M. Muller
Senior Consultant - OO, Java, SW-Architecture
Cortex Brainware, WWW: http://www.cortex-brainware.de

Q: An idea for validating the phone number field on a form..,

Does anyone have an idea for validating the phone number field on a form. I am
looking for something that will basically check this input mask

*111*111*1111*

Where the 1's are number's and the *'s are either - . spaces, or any
other character like (). Please advise.

Answer 1:  You could use a regular expression package. For example, Jakarta ORO:
http://jakarta.apache.org/oro/

Answer 2: i'm thinking regular expressions. See:
http://www.cacas.org/java/gnu/regexp/
http://www.crocodile.org/~sts/Rex/

Q: Could someone show me a basic File I/O example? I just can't figure out
streams. I'm willing to accept basic mockery in exchange...

Could someone show me a basic File I/O example? I just can't figure out streams.
I'm willing to accept basic mockery in exchange...

Answer:
```
import java.io.*;

public class FileIO {

    public static void main(String[] args) throws Exception {
        if(args.length!=1){
            System.out.println("Invalid parameters!!!");
            System.exit(0);
        }
        File fl = new File(args[0]);
        FileReader fileReader = new FileReader(fl);
        BufferedReader bufferedReader = new BufferedReader(fileReader);
        String currentLine;
        while( (currentLine = bufferedReader.readLine()) != null ){
            System.out.println(currentLine);
        }
    }
```

}

📄 Q: Does anybody know a convenient way to pause the dos program execution until user hits enter? In C I used getc. Does Java have an equivalent of "cin>>"?

Answer:
```
try {
   System.in.read()
} catch (Exception e) {
}
```

Have fun!
--
Bary

📄 Q: I've got a (simple) menu on a new application and am trying to put in the works behind the cut, copy & paste menu options - does anyone know how I can do this - what's the code or can you point me in the right direction?

Answer: Look at java.awt.datatransfer package. It contains much of the tools necessary to implement cut. copy, paste.

Can anyone please explain clearly how BufferedReader works and how to use it to get input from a keyboard?

📄 Q: Can anyone please explain clearly how BufferedReader works and how to use it to get input from a keyboard?

Answer:  BufferedReader is a filter reader class.
That is, it wraps another reader and reading from it is like reading from the reader it wraps, except that it changes something. In the case of BufferedReader, it reads in large chunks and then you can retrieve its data in smaller bits. To use it to read from System.in, you first need a reader to wrap. You can bridge from an input stream (which System.in is) to a reader by using an InputStreamReader.
Then wrap that in a BufferedReader as follows:

```
   BufferedReader input = new BufferedReader(new InputStreamReader(System.in));
```

Now you can call methods of BufferedReader to read from standard input. Generally, you create a BufferedReader to be able to call the readLine() method. That isn't BufferedReader's main intended use -- the main intended use is performance -- but you don't generally care too awfully much about performance of reads from the console. So call readLine to get a line of input, which will be null on end of stream (user presses Ctrl-D on UNIX or a file was redirected in and is done).
answered by Chris Smith

📄 Q: How do I encode the value of a variable of type long (or int) into bytes? And how do I restore the original value of the long (or int) variable back

How do I encode the value of a variable of type long (or int) into bytes such that the number of bytes used will always be the same, say 4 bytes?

Answer:
```
int in; ...
byte b1 = (byte)(in & 0xff);
```

```
byte b2 = (byte)((in >> 8) & 0xff);
byte b3 = (byte)((in >> 16) & 0xff);
byte b4 = (byte)(in >>> 24);
```

: How do I restore the original value of the long (or int) variable back
: from the bytes that i have just created then?

Answer:
```
int in = (b1 & 0xff) | ((b2 << 8) & 0xff00) |
((b3 << 24) >>> 8) | (b4 << 24);
```
by Tim Tyler

📄 Q: I would like to know if there was a method that replace a substring (in a string) by an other one.

For ex: in the string: "toto=yes, tata=no", I want to replace "yes" by "no", and "no" by "N/A",

Answer: There isn't a method to do it, since Strings are immutable, so you can't directly replace elements. (If you're doing a lot of string manipulation, shift the string to a StringBuffer.)
However, it's not hard to create a new string that performs the replacement, along the lines of:
```
int p = source.indexOf (target);
String result = source.substring (0, p) +
replacement + source.substring (p + target.length ())
```

In your example, beware of the order of replacement, since the second no will replace the first one.

Matt Humphrey

📄 Q: I have a String containing multiple logical lines separated with "\r\n". I want to tokenize that string to individual lines.
However, StringTokenizer doesn't work the way I would like it to work with "empty" lines.

Example:
```
String str = "first line\r\n" +
"\r\n" +
"third line\r\n");
```

I would like StringTokenizer to tokenize the string to the following three strings:
"first line"
""

"third line".

I have set "\r\n" to delim, but StringTokenizer.nextToken doesn't seem to return the second, empty line.
I also tried setting the returnDelims in StringTokenizer constructor to false, but then nextToken returns the "\r" and "\n" as seperate strings. It should return "", in order to make rest of my code to work OK.
How should I solve this problem?

Answer: There are several ways to get around this problem with utilities that I have written. The easiest way is to use a split function.
String[] stuff = StringHelper.split("first\r\n\r\nthird", "\r\n");
stuff will be a string array with 3 elements, including the empty line
in the middle. You can get StringHelper from:
http://ostermiller.org/utils/StringHelper.html

The second way is to use a StringTokenizer that returns empty tokens.
I have written a StringTokenizer that does such at:
http://ostermiller.org/utils/StringTokenizer.html
The only problem that you would run into is that StringTokenizer is a character tokenizer. So you would get extra empty tokens in between the \r and the \n.
--
Stephen

Q: I'm looking for a solution for parsing a String object. I need to parse a math expression and calculate it.

At first, I thought of StringTokenizer, but when I looked through the API document, multiple delimiters might not be allowed.
What is the best way to do this operation?

Answer: multiple delimiters _are_ allowed.
StringTokenizer st = new StringTokenizer(your_string, "+-/* ")
and so on.

Q: Are there any public domain matrix/linear algebra java packages out there?

I'm mainly interested in solving systems of equations and finding matrix inverses.

Answer: http://math.nist.gov/javanumerics/
The JavaNumerics page provides a focal point for information on numerical computing in Java.

Q: I am using the sinusoidal wave calculated by feeding a in function sin(a) values from zero to pi (3.142).I am wondering how is the Math.sin implemented in respect to effective calculating in practice:

Should I prepare a table with pre-calculated sine-values, or does java already have that table prepared for me?
If I 'pre-calculate' that table once, it'll make about 1024 sin-calculations plus all the lookups from the table (r, g, b + a values (4 * 255 values *4 transitions)).
Otherwise it's just the Math.sin-function call for the aforementioned RGB-values.

Answer: Most likely the JVM utilizes the floating point unit (FPU) of your microprocessor. The FPU is hard to beat these days in software. You could try a Taylor sequence, and Chebyshev approximations, but I doubt that you will be faster. On Intel, it is a hardware instruction. It is implemented internally as a polynomial approximation. You are best to treat it as a black box.
Profile it.
If speed is a problem, pre-calculating a table is an option. It's the old trade-off memory vs. performance.
--
Thomas Weidenfeller

📖 Q: Does anyone know how to write multi-line string in Java?

Answer: Something like this:

```
String a =
    "This is a
         multiline string.";
```

It is really pain to print HTML or XML from the Java program. Perl offer something like this:

```
print <<END_HTML;
  <html>
     <body>
        <h1>this is html.</h1>
     </body>
</html>
END_HTML;
```

Answer: Try this:
String a = "This is a\nmultiline string"

A "\n" stands for a line feed. Take a look at the Java language specification (downloadable on Sun's site), it has a section about strings.

Answer2: You mean like this?

```
String a = "<html>" +
              " <body>" +
              " <h1>this is html.</h1>" +
              " </body>" +
              "</html>";
```

📖 I cannot find the method to convert a binary number to an int. I used Integer.toBinaryString to get a decimal to binary but I don't know how to to convert it back.

Answer: Try using Integer.parseInt(String s, int radix) with radix = 2
that should do your job.

📖 Q: How do I launch a native Document by its Associated MIME Type? For example, I would like to ask the 'operating system' what application is associated with .DOC and then launch it.

Answer: On WinNt,

```
String docName = "c:\\someyourdir\\nameofdoc.doc";
Runtime.getRuntime().exec("cmd.exe /c "+docName);
```

📖 Q: How do I indicate Unicode characters that cannot be represented in ASCII, such as ö?

Answer: from "Java Tutorial
(http://java.sun.com/docs/books/tutorial/i18n/text/convertintro.html)

"To indicate Unicode characters that cannot be represented in ASCII, such as o, we used the \uXXXX escape sequence. Each X in the escape sequence is a hexadecimal digit. The following example shows how to indicate the o character with an escape sequence:

String str = "\u00F6";
char c = '\u00F6';
Character letter = new Character ('\u00F6'); "

Q: When I tried to read one string representing boolean value and convert it into boolean it didn't work. Finally I found that Java API has a bug!

I wrote the program that uses redaing ini file settings for initialization. All settings in a file are strings. I am converting them to appropriate type during reading. When I tried to read one string representing boolean value and convert it into boolean it didn't work. Finally I found that Java API has a bag:

boolean x = true;
getBoolean(x);
will show false!!!!

Why Java has method that doesn't work? Is it bug in Java or I am stupid?

Answer: neither statement is true! It is not a bug and you are Ok! Just please read more carefully JavaDoc next time.
It is written there for getBoolean ():
"Returns is true if and only if the system property named by the argument exists and is equal to the string "true".
(Beginning with Java 1.0.2, the test of this string is case insensitive.)
A system property is accessible through getProperty, a method defined by the System class."

So you didn't use this method properly...
Use instead:

public static Boolean valueOf(String s)

This method returns the boolean value represented by the specified String. A new Boolean object is constructed. This Boolean contains the value true if the string argument is not null and is equal, ignoring case, to the string "true".

example:
boolean x= true;
(Boolean.valueOf(x)).booleanValue()
gives you proper boolean (not Boolean!) value

Q: Is there any Java API allowing creating easily PDF files (Adobe Acrobat type) including images?

Answer: No, just text
Etymon™ PJ is a developer toolkit for parsing, modifying, and creating PDF documents.
http://www.etymon.com/pj/index.html

Q: I'm working on a java project and looking for a better API that can generate PDF, work with Excel, Word documents... Where can I find it?

Answer:

**iTest**
This library contains classes that generate documents in the Portable Document Format(PDF) and/or HTML

**XML->PDF**
FOP is the world's first print formatter driven by XSL formatting objects. It is a Java application that reads a formatting object tree and then turns it into a PDF document. The formatting object tree, can be in the form of an XML document (output by an XSLT engine like XT or Xalan) or can be passed in memory as a DOM Document or (in the case of XT) SAX events.

**JPageLayout from Sitraka**
JClass PageLayout provides sophisticated and easy-to-use APIs for adding text, images, and tables to any document. Output directly to the Java AWT Printer, Acrobat PDF, HTML, PostScript Level 2, or PCL 5. Customize almost every aspect of your report styles and print output for professional results.

Q: How can I extract text from PDF file?

Answer: I have used the Acrobat Viewer Bean (http://www.adobe.com/products/acrviewer/main.html) to extract text from PDFs. This bean is quite buggy but it was OK for this task. One drawback is that it depends on AWT even if you don't do anything GUIish with it, so if you want to use it on a Server with no X-Windows running you'll have to install the Pure Java AWT to get things running.
--
Patrick

Q: I'm looking for a rich text editor that I can embed within a web page, and allow users to enter rich text that I can in turn store as HTML.

I've seen similar applets through web based e-mail clients. I'd appreciate it if someone could point me in the right direction!

Answer:
Try Swing, by Robinson, Manning Publication. You could probably adjust the code to fit into the applet style.
It is here

http://javafaq.nu/java/free-swing-book/free-swing-book-chapter20.shtml

John

Q: I want to have a while loop execute for a maximum of n seconds or until it receives something in an input stream. How would I do this?

Answer: I think you could do it this way:
*******************************************
InputStream Input=null;
int n=10; /*Number of seconds to wait*/
/*initialize you input stream*/

```
...
/*Now start you while loop*/

long lStart = (new Date()).getTime();
long lCurrent = (new Date()).getTime();

while((lCurrent-lStart < n*1000) && (Input.available()==0)){
    Thread.currentThread.sleep(100); /* This will give JVM time to other threads */
    lCurrent = (new Date()).getTime();
}
*******************************************
```

You could simply count number of steps inside the loop keeping in mind that each step takes 100ms but I think using dates would be a bit more precise method.
Alex Shlega

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

# String, text, numbers, I/O II part

Q: How do I parse it to date and make a date difference?? How many days in between the String from and to?

I am always confusing with date and time.

I have 2 String, from and to
String from = "01.01.2001" //dd.MM.yyyy
String to = "01.03.2001" //dd.MM.yyyy

How do I parse it to date and make a date difference?? How many days in between the String from and to?

Answer:
```
import java.text.SimpleDateFormat;
import java.util.Date;
public class Tmp {
  public static void main( String argv[] ) throws Exception {
    long DAY = 24L * 60L * 60L * 1000L;
    SimpleDateFormat df = new SimpleDateFormat( "MM.dd.yyyy" );
    Date d1 = df.parse( "01.01.2001" );
    Date d2 = df.parse( "01.03.2001" );
    System.out.println( "The number days between:" );
    System.out.println( d1 );
    System.out.println( "and:" );
    System.out.println( d2 );
    System.out.println( "is: " + (( d2.getTime() - d1.getTime() ) / DAY ));
  }
}
```

But the calculation of the difference in times may not be suitable due to timezone issues and such. I believe there must be a better way?

Michael B. Allen

📧 Q: I am looking for some code to compare current date and a date entered by user. I want to return years between dates. I can't find a method to do this with.

Answer: try GregorianCalendar class which is found in java.util.*

```
Calendar fdate= new GregorianCalendar(1999,1,25);
Calendar ldate= new GregorianCalendar(2000,1,25);

then use get methods and compare
fyear= fdate.get(Calendar.YEAR);
lyear= ldate.get(Calendar.YEAR);
```

📧 Q: If I have a string with a hex value (e.g. "7FFFFF"), is there a way to obtain an int value from this (e.g. 8388607)?

Answer: See Integer.parseInt (String s, int radix). The radix for a hexadecimal number is 16.

📧 Q: How do I convert this double to the following String:

```
double d = 19.969332079021637;
String s = "19.97";
```

Answer: double d = 19.969332079021637;
```
DecimalFormat df = new DecimalFormat("##,###.##");
String s = df.format(d);
```

or

Use NumberFormat in the java.text package

```
...
double d = 19.969332079021637
NumberFormat format = NumberFormat.getInstance();
format.setMaximumFractionDigits(2);
format.setMinimumFractionDigits(2);
String s = format.format(d); // s="19.97"
```

📧 Q: How do I convert a String to an integer. For example: String strSEQ = "SEQ7";

How do I pull "7" from the string and convert it to an integer?

Answer: I'd do something like this:

```
StringBuffer digits = new StringBuffer();
char c;
for (int i=0; i<strSEQ.length(); i++) {
   c = strSEQ.charAt(i);
   if (Character.isDigit(c))
      digits.append(c);

}
int intValue = Integer.parseInt(digits.toString());
```

--
Michiel

Q: How can I display other languages such as chinese, french, japanses etc. inside applet or frame?

Answer: You have to install the approriate font, and add it to a font.properties file. Further information can be found here:

http://forum.java.sun.com/read/16805306/q_CUP8NP-1rgAAYsA#LR
http://java.sun.com/products/jdk/1.1/docs/guide/intl/index.html
http://developer.java.sun.com/developer/qow/archive/65/index.html

Q: I need to determine the current year, at runtime, in the form of a four digit number, preferably in "int" form. How can I do this?

Answer: Hey, what makes you think that the year is a four digit number? It won't always be, you know. That's the kind of sloppy thinking that got us into that whole y2k mess, and we all know what a huge issue THAT turned out to be!

If you write your program assuming that the year is 4 digits, it will cease to function properly in less that 8000 years! Try not to have such a "short term" viewpoint.

Here's the deprecated solution:

int year = new java.util.Date().getYear() + 1900;

This was deprecated because it is too convenient! We programmers often get paid per line of code, you know.
Actually it was deprecated because it didn't have an implicit way to deal with time zones and it contains that silly Y2K problem.


The SUPERIOR way to do is as follows:

java.util.Calendar cal = Calendar.getInstance();
int year = cal.get(Calendar.YEAR);

--
Marshall

Q: I have a string like 23,N/A,,N/A,87, then I used the StringTokenizer t = new StringTokenizer( s, ",");

to get the token. I seems I only got 4 tokens back, not 5. How doesStringTokenizer treat the ,, in this case? I had tried String.equals("") or String.equals(null) to pick the EMPTY string between ,, , it did not work.
Anyone can give a clue?

Answer: There is another constructor, with a third parameter. Using that one, you can get the`boundary markers, such as the comma, returned to you as well. That would allow you to handle your empty strings.
--
Michele.

📑  Q: How to know in Java if a given font has or not a fixed width ?

For example to list all the "fixed width" fonts that are available on the user's platform.

Answer: I'm not sure whether there's an easier solution... if there is none, you might try to get the FontMetrics objects from a specific Font. Then iterate over every character and call charWidth(char ch) on each. It should return the same value for each character if it's monospaced, different values for each one otherwise.

This might be a little "hacky", but if you really need an idea...

Karsten

📑  Q: I want to create a Date object by just specifying the Year, Month and Day.

e.g. the user can enter 1998 as Year, 11 as Month of Nov, 15 as the day.
With these three information, how can i create a Date object (or similar object that stores the data)?

Answer: The correct way to do this is something like this:

Calendar cal = Calendar.getInstance();
cal.set(1998, Calendar.December, 15);
Date date = cal.getTime();

📑  Q: Let's say I have a text string with the text: Hello! Then how do I add quotes to it so it will be: "Hello!"

Answer:
answer is String s = "Hello!"

"\"" + s + "\""

and if the result is to become the new value of s,

s = "\"" + s + "\""

📑  Q: I want to send Image through network. I made OutputStream to send Image...

OutputStream out = socket.getOutputStream();
but I can't send Image directly.
How can I send Image?

Answer: Unfortunately, the java.awt.Image class is abstract and is not serializable, otherwise this would be easy.

From the Java docs: "The abstract class Image is the superclass of all classes that represent graphical images. The image must be obtained in a platform-specific manner."

Consider using the Image.getSource method, getting the producer, getting the bytes, then send [width][height][bytes...].
--
Mark Watson, Java consulting, Open Source and Content: www.markwatson.com
Commercial software products: www.knowledgebooks.com

📋 Q: I want to get the code behind an HTML page.

Answer:
```
URL url = new URL("http://www.sun.com");
BufferedReader reader;
reader = new BufferedReader(new InputStreamReader(url.openStream()));
String str;
while ((str = reader.readLine())!=null){
   // process str
}
```

📋 Q: I am wondering if JDK supports to open a file in the exclusive mode?

Answer: No, file locking is not supported in current VMs. You can implement lockfiles, however, using the File.createNewFile() method. Since JDK 1.3: "Atomically creates a new, empty file named by this abstract pathname if and only if a file with this name does not yet exist. The check for the ' existence of the file and the creation of the file if it does not exist are a single operation that is atomic with respect to all other filesystem activities that might affect the file. This method, in combination with the deleteOnExit() method, can therefore serve as the basis for a simple but reliable cooperative file-locking protocol."

📋 Q: How do I write an image to stdout? This is for a cgi program that will return an image to a browser.

For example, if a user click on a link to http://somewhere.edu/cgi/getpix.cgi the browser will receive image data from the cgi program and will display the image.

Answer: Send "Content-type: image/gif" - and then send the data.

Alternatively, send an HTML page, with a pointer to the (possibly-freshly-written) image on the server.
--
Tim Tyler Try my latest game - it rockz - http://rockz.co.uk/

📋 Q: Does anyone know how I can get around the fact that a Message obj (from the JavaMail API) is not serializable?

I would like to save a message obj to the hd, then read the file (is 'file' the wrong word here?) back to my e-mail client and read it as a Message obj.

Answer: The obvious answer is: implement serializable yourself in a subclass! But! That won't work. The serialization mechanism will throw an exception at runtime. Even if it didn't, the class may be nonserializable for a good reason, for example it might contain a socket, which can't be reinitialized from a stream.

What you need is a subclass that implements the Externalizable interface, and then your own code that saves as serializable objects those attributes of the object that you need to reconstitute it later. I'm not familiar with this particular object, but I imagine it has several get() methods that return serializable objects like Strings or ints. What you have to do is recover those objects and put them into the output stream yourself, and reverse the process on unserialization. The fields that are not serializable you ignore.

You have to provide a no-argument constructor that reconstructs anything you aren't going to serialize (for example, opening a new socket). Read about the Externalizable interface for more details.
--
Frank LaRosa

Q: Is it possible to send Vector class in socket communication. If it is, what function can make it.

Answer: Make sure the objects contained in the Vector are Serializable. Look at java.io.ObjectOutputStream for serializing the Vector by wrapping the output stream from socket with the ObjectOutputStream...

Q: I wrote a little Java Tool that imports a 10 MB Text file. My Problem is that this text file has no line breaks, that means the whole file contains one record, which are all records put to one together.

With the readLine () method it takes more than an hour to complete my instruction.

How can I accelerate this process, I only need to insert a break line after each "record" (after x characters -> "\n").

Answer: If you will tell us what you are trying to accomplish, we may be able to offer a suggestion.

If you are trying to reformat the file to have line breaks, you will have to read the entire file once and write out a new file with the line breaks. It seems obvious that the original file was meant to be random-accessed and has fixed-length records, in which case it is just fine as it is -- if it is read appropriately.
--
Paul Lutus
www.arachnoid.com

Q: How can you tell if an integer is odd or even?

I know an even number is divisible by 2 but I'm thinking then how can I detect if a resulting number after dividing 2 ints has a remainder?

Answer: Basically the method is simple, if a variable contains an odd number I want to return the value 'zero', and on the other hand if the variable contains an even number I want to return the value 'one'.

Check out the % (modulus) operator, it computes the remainder.
```
public int modMethod(int div){
    if(div%2==0){
        return 1;
    } else{
        return 0;
    }
}
```

Q: I'm looking for an algorithm that would compress a large integer down to a not-that-long string. Actually, it's not an integer, but a series of integers that could have hundreds of digits in total.

Answer: Use classes ZipInputStream and ZipOutputStream. They are meant for

exactly this purpose. And they meet the requirement of portability between languages (because ZIP compression is ubiquitous).
--
Paul

**Q: How can I round a number to specified precision?**

I have a double field that I would like to round to 2 places of precision, however, it seems like the documentation on the round function only rounds to closest integers. So that I would not be able say .3658585859 = .37 as I would like.

Answer: can you scale the number up and then down again when you are finished?
e.g.  0.3658585859 * 100 = 36.58585859
round(36.58585859) = 37
37 / 100 = 0.37

**Q: I understand that bitwise operations change the 0/1 bits of a number. Question is why?**

I suppose it's interesting that you can manipulate numbers this way, but I can't think of a practical use for doing that.

Can anyone help me understand when are bitwise operations used and why you would use them?

Answer: Bitwise manipulation is often used where memory consumption is critical, and a piece of information may be encoded in less that one byte, for instance. In communication software and protocols, information may be interpreted as a stream of bits where the information is encoded at the bit-level, and you use bitwise manipulation to extract the pieces of information encoded in the bytes. There are other situations where bitwise manipulation is used, as well.
by Greger Ohlson

**Q: Why cannot I cast from double to java.lang.Object?**

I'm trying to build a vector, however, one of the objects that I'm passing to the vector is of type double. How do I cast the double as an object so that I may insert the value into a vector? Does this make sense? Here is the following snippet of code I was trying to use:

myVector.add (1, (Object)myDouble);

Of course when I try to compile I get the following message:

Invalid cast from double to java.lang.Object

Could someone please explain why? I realize that Object is the mother of all objects and therefore ANY reference data type "is an" Object. So therefore I shouldn't have to cast the double, right? Help, I'm a beginner!

Answer: A double is not a reference type, but a primitive one. Hence, it doesn't inherit from Object (or anything else, for that matter). To put primitives (byte, short, int, long, float, double, boolean, char) into something that requires an Object, use Java's wrapper classes.

The wrapper classes are Double, Integer, Long, Boolean, etc., and are basically an object "wrapped" around a primitive type. You make a Double object by:
Double d = new Double (myDouble);

and to get the actual value back,

double z = d.doubleValue();

It works the same way for all the rest of the primitive/wrapper pairs.
by Trevor Hill

Q: What is the easiest way to break an int into two bytes?

Answer: Maybe you have big bytes :) if you are going to break an int into two bytes
A Java int is four bytes large. Do something like this:

```
int i = 4344234;
byte b1 = (byte)(i & 0xff);
byte b2 = (byte)((i >> 8) & 0xff);
byte b3 = (byte)((i >> 16) & 0xff);
byte b4 = (byte)((i >> 24) & 0xff);

--
Marco
```

Q: is there a mod (x, y) function that returns the remainder when x is divided by y? Something equivalent to fmod(x,y) in C?

Answer: a = x%y;

Q: I'm having trouble figuring out how to convert characters to their ASCII value in java. Is there a class like NumberFormat that will do it?

Answer: I can't see any problem here:

```
char ch = 'A'; // character 'A'
int i = (int)ch; // ASCII value for 'A' (=>65)
```
Yes. And just be aware that ASCII only runs from 0 through 127. Anything higher needs to be addressed differently, since Java is using Unicode values.

Q: How to do "Press any key to continue"? I want to do it at Console.

Answer: // ReadConsole.java

```
import java.io.*;

public class ReadConsole {

   public static void main(String args[]) throws IOException {
      System.out.print("Press Enter to continue: ");
      System.in.read();
   }
}
```

You cannot have "press any key" from the console, for various system-dependent reasons. You need to press Enter.


--
Paul Lutus
www.arachnoid.com

Q: Just wondering how people generally convert BufferedOutputStream into a BufferedInputStream to be read from....

This seems really stupid, but I can't find a way to do it in the API....

Answer: if you want to just take what's coming in and send it out then do something like this

BufferedInputStream in = new BufferedInputStream(some inputstream);
BufferedOutputStream out = new BufferedOutputStream(some outputstream);

int i = 0;

while((i = in.read()) != -1){
out.write(i);
}

of course you will have to handle exceptions, but that should be the general way to do it.
--
michael

Q: I have heard that String concatenation operator + affects performance of program if it used much. Is it true?

Answer: Yes, it affects your program performance if you do a lot of "+" operations with strings:
A new StringBuffer must be created, then two arguments are added to it with append(), and the final result must be converted back with a toString().
Your time and space is wasted...
In case if you are appending more than one String, try to use a StringBuffer directly.
--
Alexandre P. (J.A.)

☐ IT Professional  ☐ General Computer  ☐ Design and Graphics  ☐ Linux  ☐ Freeware/Shareware

☐ Games  Intranet  Computer Hardware  ☐ Web Design  ☐ Computer Software

☐ UNIX  ☐ Web Software  ☐ Windows Software  ☐ Personal Finance  ☐ Programming Software

☐ Servers  ☐ C/C++  ☐ Powerbuilder  ☐ Perl  ☐ XML

☐ Java  ☐ SQL  ☐ Design/Graphics  ☐ Utilities  ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## About this e-book

This book is my collection of numerous advices, tips I have collected on numerous Java fora, newsgroups and my site. Here also many answers from my letters to people that asked me questions by e-mail.

This book is a result of my daily work during last year. All answers can be found somewhere on Internet, but it takes much time to find them. Having all of them in one place, in the form of e-book, saves you hundreds hours.

The answers are in form of advices, tips. Often they will guide you how to solve problem but will not give exact code implementation. This book for "researcher" kind of people, who can manage to overcome obstacles without asking always full receipt.
That's why this book, I think, will be very useful not only for beginners but for advanced programmers as well.

Many questions in this book are not frequently asked and furthermore, easily answered by Java Gurus.

They will show you that Java has many advantages that are not so obvious during your first study of it.

I bet that, if you even passed Java Certification Exam, you can not answer most of them stright away! This kind of questions demands not only understanding of JLS (Java Language Specification) but more broad knowledge as well, often from another languages. Part of questions are touching a few languages simultuoneously and do comparison of pros and cons.

Thus, they will give you more knowledge, better perspective to programming and computer science especialy. In modern live we have to be better prepared for fast changing world. Computer world changes even faster! This is most dynamically developing area of human knowledge and get updated fast - most difficult problem. I often feel that it is like ice cake riding . Small delay and you are in the cold water!

As I mention above, this book is collection of tips that are given by many people on different Internet fora. I had no possibility to contact to all of them about right to include their advices into this book. Since they were published on public forums and are redistributed by many sites (that give access to newsgroups and their archives) I believe it will not be a problem if I

will collect some of them into one book.

I kept the names and signs under tips that often points to their sites. I think it will be even beneficial for them to be included into this book.

Anyway, if somebody of you recognize the tip as yours and do not agree to be included into my book, please send me e-mail and I will exclude your tip from this book.

The price for this book is a price of my job on collection and distribution this book (unfortunately big traffic costs money today).

I will be glad to hear your comments and wishes on improving this book! As I mentioned on main page of this e-book, English is not native language for me and this book can (probably does) be in some extent grammarless. Send me your correction of my mistakes!

John Andersson

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## About Author

John Andersson is my pseudonym which I use in my newsletters and on my website JavaFAQ.nu.
It will be probably long explanation why I, as well as another people, use pseudonyms. So, I skip it for "clarity" :-)

I am planning to continue my job in the future and collect more tips for you!

### Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

| | | | | |
|---|---|---|---|---|
| ☐ IT Professional | ☐ General Computer | ☐ Design and Graphics | ☐ Linux | ☐ Freeware/Shareware |
| ☐ Games | Intranet | Computer Hardware | Web Design | ☐ Computer Software |
| ☐ UNIX | ☐ Web Software | ☐ Windows Software | ☐ Personal Finance | ☐ Programming Software |
| ☐ Servers | ☐ C/C++ | ☐ Powerbuilder | ☐ Perl | ☐ XML |
| ☐ Java | ☐ SQL | ☐ Design/Graphics | ☐ Utilities | ☐ Handhelds |

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

All questions please mailto:info@javafaq.nu

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

Java Specialist's Tips (Advanced!) from Dr. Heinz M. Kabutz.

Java Specialist's Tips are not simple short advices, they are well written articles! Each article exploring one area and gives you examples.

Dr. Heinz Kabutz is a Java guru living in South Africa. He consults, holds courses, programs, and - writes a weekly newsletter in which he shares some rather unconventional insights about Java.

Easy style, right proportion of humor and Java lets you read them at once, in one breath!
You will appreciate very much job that was done by Dr. Heinz M. Kabutz. The site of The Java™ Specialists Newsletter is here!

John Andersson

# Contents of Java Specialist's Tips

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📰 Anonymous Inner Classes

Author: Dr. Heinz M. Kabutz

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the second "The Java(tm) Specialists' Newsletter", a low-volume newsletter that is aimed at in-the-trenches Java programmers and those interested in what the trenches look like nowadays. According to the latest Computing SA, the e-commerce industry in South Africa is struggling to take off because it is almost impossible to find Java programmers with any real experience. Germany seems to be experiencing a similar trend so they want to import 30'000 Indians. Aren't we working in an exciting segment of the IT market? I'm so glad I was too jung to try cash in on Y2K bugs and instead invested my time learning more of Java.

You are either on my mailing list because you begged to be added or because you did not beg to be removed, in either case, I hope you are enjoying reading this information and that it will be of benefit to you. Please let me know if there are topics that you would like discussed or if you would like to share some of your own experiences through this newsletter. This newsletter is distributed free of charge to anyone interested, so please forward it to friends and colleagues who might find the information in this newsletter useful.

In the last newsletter I mentioned in the non-kosher section that Java makes provision for more than one GUI thread to be in the system and that I did not know if/when this feature was used. So far I have not found out what the purpose of this is or if it is still used, although I suspect it's a leftover from the Awful Windows Toolkit. Do we have a Swing expert in the house?

## Anonymous Inner Classes

A feature that we did not have in the early days of Java, when all you needed to run the VM was a 80386 with 4 MB RAM and a correctly set CLASSPATH, was the concept of the inner class, specifically the anonymous inner class (heck, we didn't need that newly-fangled fancy stuff then - we were real programmers ;-). Inner classes have made it possible to code with less lines and in the process to obscure the code to the unenlightened but make it wonderfully elegant to those who understand. A good friend and colleague told me "One good Java programmer is better than ten

bad Java programmers" which I wholeheartedly agreed with. He then went on to say "And five bad Java programmers are better than ten bad Java programmers". Inner classes have widened this divide. The typical way of using anonymous inner classes is for writing GUI event handlers, e.g.

```java
button.addActionListener(new ActionListener() {
  // This is how you define an anonymous inner class
  public void actionPerformed(ActionEvent e) {
    System.out.println("The button was pressed!");
  }
});
```

The amazing thing is that we are actually defining a new class(!) while calling another method. You can virtually make new classes in all sorts of places in Java. All of this is old hat to most of you, but last week I found a new application of anonymous inner classes that I had not thought of before. But first a diversion:

A lot of books seem to suggest that if you want to create a Thread in the middle of your code you do it as shown in this code. The example is typically known as an "idiot counter", patented by Microsoft ;-), in which they make the user dream that progress is happening by changing the screen output in some way..... (I have used this technique quite successfully in the past)

```java
new Thread(new Runnable() {
  public void run() {
    try {
      while (true) {
        sleep(1000); System.out.print(".");
      }
    }
    catch(InterruptedException ex) {}
  }
}).start();
```

If we look at the definition of Thread we see that it takes a Runnable as a parameter so it sort-of makes sense to create an anonymous inner class from Runnable and stick that into the parameter. However, looking more carefully inside Thread we notice that the run() method defined in Thread calls the run() method defined in Runnable, if a Runnable has been passed into the Thread constructor. Instead, it would be more efficient to do the following, because we would have one less object on the heap and one less method call per Thread creation:

```java
new Thread() {
  public void run() {
    try {
      while (true) {
        sleep(1000); System.out.print(".");
      }
    }
    catch(InterruptedException ex) {}
  }
}).start();
```

This way Thread itself is made into an anonymous inner class and we override the run() method so instead of Thread.run() having to check that a Runnable exists we can just execute the code in run(). It is a very small difference and I don't know why I have not seen it used before, but I suspect one of the early Java 1.1 lemmings jumped over the cliff holding a placard containing the Runnable as a parameter and all the other lemmings followed the sign ;-) To me the difference is more conceptual than actual and leads me on to the next application of inner classes:

Anyway, I now get to the real reason for another newsletter, which is an application of anonymous inner classes which I find quite useful. In Java 1.0, when we had to pass an array as a parameter to a method we did it as follows:

```
  String[] temp_names = new String[3];
  temp_names[0] = "Heinz";
  temp_names[1] = "John";
  temp_names[2] = "Anton";
  universityRegistration.addNames(temp_names);
```

or, alternatively

```
  String[] temp_names = { "Heinz", "John", "Anton" };
  universityRegistration.addNames(temp_names);
```

In Java 1.1, SUN sneaked in a new construct so we would not need to have a temporary variable, which according to the Refactoring folklore is bad. Since JDK 1.1 we could thus say:

```
  universityRegistration.addNames(
    new String[] { "Heinz", "John", "Anton" });
```

If you wanted to pass in a Collection instead of an array it would look as follows:

```
  Collection temp_names = new Vector(3);
  temp_names.add("Heinz");
  temp_names.add("John");
  temp_names.add("Anton");
  universityRegistration.addNames(temp_names);
```

The ability to avoid local temporary variables with arrays was always a strong deciding factor in defining interfaces to my classes because I could get away with one line of code instead of five, and the less lines of code the better. I would therefore rather define addNames(String[] names) than addNames(Collection names) even if it meant I would have to convert backwards and forwards between arrays and collections. However, with anonymous inner classes we can get the same effect seen above but with collections:

```
  universityRegistration.addNames(new Vector(3)
    {{ add("Heinz"); add("John"); add("Anton"); }});
```

How does work? Very simple, say we wanted to extend Vector with our own Vector, called MyVector, that contains the three elements "Heinz", "John", "Anton":

```
  public class MyVector extends Vector {
    public MyVector() {
      super(3); // to initialise it with a size of 3
      add("Heinz"); add("John"); add("Anton");
    }
  }
```

The call to the super constructor always happens first, so we could re-write MyVector as follows without changing the functionality in any way:

```
  public class MyVector extends Vector {
    { // initializer block
      add("Heinz"); add("John"); add("Anton");
    }
    public MyVector() {
      super(3); // to initialise it with a size of 3
    }
  }
```

If we want to make an instance of an anonymous inner class we can pass the parameters directly to the super class via the parameter list of the constructor of the anonymous class. In addition, any init block denoted by {} is done AFTER the call to the super class constructor is completed, so the class MyVector could look like this:

```
  Vector myVector =
```

```
    new Vector(3) { // defining anonymous inner class
    {
        add("Heinz"); add("John"); add("Anton");
    }
};
```

From here the step to addNames(new Vector(3) {{ add("Heinz"); add("John"); add("Anton"); }}); is quite simple, I've just removed the comment and bunched the curly brackets together.

The thing that amazes me most about this "discovery" is that it is such an obvious thing to do if you have to pass a collection into a method and you know the size and elements of the collection, but in 3.5 years of Java I'd never thought of it, and I bet it does not occur anywhere in the 570'000 lines of JDK 1.3 code or the mega-project I worked on in South Africa.

---

*Warning Advanced:*
*When you access private data members of the outer class, the JDK compiler creates package-access member functions in the outer class for the inner class to access the private members. This firstly leaves a security hole, although all data members in Java are public to the initiated ;-), and secondly used to impede performance before hotspot came along. So, if you are writing code that has to run reasonably well on non-hotspot code it would be better for performance reasons to make data members accessed by inner classes package-access.*
---

As always, I welcome your comments and thank you for the positive feedback from last week.

With regards

Heinz

---

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

### 📋 Blocking Queue

**Author: Dr. Heinz M. Kabutz**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 16th issue of "The Java(tm) Specialists' Newsletter", written in a dreary-weathered-Germany. Since I'm a summer person, I really like living in South Africa where we have 9 months of summer and 3 months of sort-of-winter. It's quite difficult to explain to my 2-year old son the concepts of snow, snow-man, snow-ball, etc. Probably as difficult as explaining to a German child the concepts of cloudless-sky, beach, BSE-free meat, etc.

Next week I will again not be able to post the newsletter due to international travelling (Mauritius), but the week after that I will demonstrate how it is possible to write type-safe enum types in Java using inner classes and how it is possible to "switch" on their object references. Switch statements should never be used, but it is nevertheless fascinating to watch how the Java language constructs can be abused...

## Blocking Queues for inter-thread communication

This week I want to speak about a very useful construct that we use for inter-thread communication, called a blocking queue. Quite often in threaded applications we have a producer-consumer situation where some threads want to pop jobs onto a queue, and some other worker threads want to remove jobs from the queue and then execute them. It is quite useful in such circumstances to write a queue which blocks on pop when there is nothing on the queue. Otherwise the consumers would have to poll, and polling is not very good because it wastes CPU cycles.

I have written a very simple version of the BlockingQueue, a more advanced version would include alarms that are generated when the queue reaches a certain length.

---
*Warning Advanced:*
*When I write pieces of code which are synchronized, I usually avoid synchronizing on "this" or marking the whole method as synchronized. When you synchronize on "this" inside the class, it might happen that other code outside of your control also synchronize on the handle to your object,*

*or worse, call notify on your handle. This would severely mess up your well-written BlockingQueue code. I therefore as a habit always use private data members as locks inside a class, in this case I use the private queue data member.*

*Another disadvantage of indiscriminately synchronizing on "this" is that it is very easy to then lock out parts of your class which do not necessarily have to be locked out from each other. For example, I might have a list of listeners in my BlockingQueue which are notified when the list gets too long. Adding and removing such listeners from the BlockingQueue should be synchronized, but you do not have to synchronize in respect of the push and pop operations, otherwise you limit concurrency.*

---

```java
//: BlockingQueue.java
import java.util.*;
public class BlockingQueue {
  /**
    It makes logical sense to use a linked list for a FIFO queue,
    although an ArrayList is usually more efficient for a short
    queue (on most VMs).
   */
  private final LinkedList queue = new LinkedList();
  /**
    This method pushes an object onto the end of the queue, and
    then notifies one of the waiting threads.
   */
  public void push(Object o) {
    synchronized(queue) {
      queue.add(o);
      queue.notify();
    }
  }
  /**
    The pop operation blocks until either an object is returned
    or the thread is interrupted, in which case it throws an
    InterruptedException.
   */
  public Object pop() throws InterruptedException {
    synchronized(queue) {
      while(queue.isEmpty()){
        queue.wait();
      }
      return queue.removeFirst();
    }
  }
  /** Return the number of elements currently in the queue. */
  public int size() {
    return queue.size();
  }
}
```

Now we've got a nice little test case that uses the blocking queue for 10 worker threads which will each pull as many tasks as possible from the queue. To end the test, we put one poison pill onto the queue for each of the worker threads, which, when executed, interrupts the current thread (evil laughter).

```java
//: BlockingQueueTest.java
public class BlockingQueueTest {
  private final BlockingQueue bq = new BlockingQueue();
  /**
     The Worker thread is not very robust.  If a RuntimeException
     occurse in the run method, the thread will stop.
   */
  private class Worker extends Thread {
    public Worker(String name) { super(name); start(); }
    public void run() {
      try {
        while (!isInterrupted()) {
          ((Runnable)bq.pop()).run();
        }
      } catch(InterruptedException ex) {}
      System.out.println(getName() + " finished");
    }
  }
  public BlockingQueueTest() {
    // We create 10 threads as workers
    Thread[] workers = new Thread[10];
    for (int i=0; i<workers.length; i++)
      workers[i] = new Worker("Worker Thread " + i);
    // We then push 100 commands onto the queue
    for (int i=0; i<100; i++) {
      final String msg = "Task " + i + " completed";
      bq.push(new Runnable() {
        public void run() {
          System.out.println(msg);
          // Sleep a random amount of time, up to 1 second
          try { Thread.sleep((long)(Math.random()*1000)); }
          catch(InterruptedException ex) { }
        }
      });
    }
    // We then push one "poison pill" onto the queue for each
    // worker thread, which will only be processed once the other
    // tasks are completed.
    for (int i=0; i<workers.length; i++) {
      bq.push(new Runnable() {
        public void run() {
          Thread.currentThread().interrupt();
        }
      });
    }
    // Lastly we join ourself to each of the Worker threads, so
    // that we only continue once all the worker threads are
    // finished.
    for (int i=0; i<workers.length; i++) {
      try {
        workers[i].join();
      } catch(InterruptedException ex) {}
```

```
      }
      System.out.println("BlockingQueueTest finished");
   }
   public static void main(String[] args) throws Exception{
      new BlockingQueueTest();
   }
}
```

The concepts in the newsletter can be expanded quite a bit. They could, for example, be used as a basis for implementing a ThreadPool, or otherwise you can implement an "ActiveQueue" which performs callbacks to listeners each time an event is pushed onto the queue via a Thread running inside the ActiveQueue.

It is also possible to use PipedInputStream and PipedOutputStream to send messages between threads, but then you have to set up a whole protocol, and if you want to exchange objects you have to use ObjectOutputStream which will be alot slower than just passing handles.

Until next week, and please remember to forward this newsletter in its entirety to as many Java users as you know.

Heinz

---

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

### Boolean comparisons

**Author: Dr. Heinz M. Kabutz**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 8th issue of "The Java(tm) Specialists' Newsletter", where we look at "in-the-field" tips and tricks used by Java professionals. My intention with this newsletter is to have some fun spreading knowledge of advanced Java concepts, and in the process learning new things, so please keep on forwarding your ideas, comments and criticisms.

A special welcome to those of you who found out about this newsletter through Bruce Eckel's website, and a very hearty "thank you" to Bruce, the author of the best Java learning book of all time!

# Comparing boolean variables with true or false

Java was based on ideas taken from C/C++, but the designers of the language tightened up the language significantly. In C/C++, a zero (0) means false and non-zero (e.g. 1) means true. This caused a lot of bugs that were not found at compile time, especially with assignment vs. comparison.

Here's a snippet of C++ code, which has the same syntax as Java

```
int i = someNumber;
if (i == 4) {
   /* do something */
}
```

In C/C++ it is extremely easy to write

```
int i = someNumber;
if (i = 4) { // note we are doing assignment, not comparison!
```

```
    /* do something */
  }
```

This will have the effect of assigning 4 to "i", regardless of what "someNumber" was equal to, the result will be non-zero, i.e. true, and after this code, "i" would equal 4.

In Java, we now have the "boolean" type which is either "true" or "false". The parameter to the "if" statement has to be "boolean", rather than an int as in C/C++. The code assigning 4 to i would not compile in Java. This is great, because we find a lot of bugs at compile time.

---
*Warning Advanced:*
*Memory optimisation insider trick: Incidentally, when you have a boolean data member,*
*according to my experiments, it is counted as 4 bytes! A boolean array uses one byte per*
*boolean.*
---

This is all very fundamental, so why am I writing this in an advanced Java newsletter? When we assign one boolean to another, the returned value is that of the value we are assigning to. This means that the bug mentioned above is still possible if we are doing boolean comparisons! I have often seen code such as the following, even from experienced, otherwise good, Java programmers (I counted over 150 occurance in the project I'm currently working in):

```java
boolean pentiumTMcpu = Utils.isCpuAPentium();
if (pentiumTMcpu == true) {
  /* work out incorrect salary using double */
}
```

This will compile fine in Java, but so will the following, which assigns true to pentiumTMcpu and will always work out the salary using the Pentium bug (younger readers would not remember):

```java
boolean pentiumTMcpu = Utils.isCpuAPentium();
if (pentiumTMcpu = true) {
  /* this code will always be executed */
}
```

Instead, it would be a lot safer to write

```java
boolean pentiumTMcpu = Utils.isCpuAPentium();
if (pentiumTMcpu) {
  /* work out incorrect salary using double */
}
```

It is very easy for a bug to slip in during maintenance so we should always think about how we can reduce the possibility of bugs being introduced by less experienced, less expensive, less intelligent, less careful future programmers who have to maintain the code we write. (I'm kidding about the less expensive.)

There is a technique used alot in Microsoft's C++ libraries in which they would have written the comparison as:

```java
boolean pentiumTMcpu = Utils.isCpuAPentium();
if (true == pentiumTMcpu) {
  /* work out incorrect salary using double */
}
```

This is also quite safe, since you cannot assign anything to true, and so you would get a compiler error if you only had one "=". I personally don't like that style, but it is just as safe as writing "if (pentiumTMcpu)", except that you might confuse someone who comes to Java having never written C/C++.

That's the end of this newsletter. I am trying to keep these newsletters short, but sometimes get

carried away with my story telling, please forgive me. I generally have topics lined up one month in advance, but if there is something you would like to contribute, please send me an email.

Next week I will look at a topic I call "depth-first-polymorphism", brought to my attention by Dr. Jung, who wrote an earlier newsletter.

Until next week

Heinz

---

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📋 Circular Array List

**Author: Dr. Heinz M. Kabutz**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

**Please let me know if you have trouble reading this newsletter in HTML format.** This new format is meant to make it more pleasant to read the code examples, so please give me feedback on whether it **sucks** or **rulez**.

---

Welcome to the 27th issue of "The Java(tm) Specialists' Newsletter". Last week I was talking to a friend about keyboards and he mentioned what a pity it was that we had grown up on QWERTY, when Dvorak is sooo much better for the fingers. I'm sure you've heard the story that QWERTY was designed to slow you down? He also mentioned that some students he studied with removed all the keys from their keyboards and arranged them in Dvorak order. "What a good idea!", I thought, promptly removing all the keys from my notebook and dropping them down in Dvorak order. There are two problems, however:

1. I touch type QWERTY, so until I find the time to learn Dvorak I want to continue QWERTYing. Some Java applications (JBuilder4, JProbe, etc.) have the extremely annoying habit of changing my input locale to the bottom selection, in my case Dvorak.

2. Three of the keys are different sizes to the rest of the keys, so in order to remember what they are supposed to be, I stuck stickers on them and the places where they are supposed to be, and wrote on them the appropriate Dvorak keys. Whenever someone sees my notebook now, they ask: "Why do you have stickers on your keys?" They don't see that ALL the keys are in the wrong place! Sometimes we are like that too when it comes to Java. We say: "Java is slow!" In the meantime, we use the wrong collections, we create too many objects, we use += with Strings and print new java.util.Date() to the console in a tight loop. When people make statements like "Java is a memory hog" they are demonstrating that they are not looking at the whole picture.

One of the advantages in writing this newsletter is that I get some excellent feedback on some of the things I write about. In newsletter 24 I spoke about self-tuning FIFO queues, the purpose of which was to demonstrate some ideas on self-adapting code, rather than propose a good FIFO implementation. Ecce Jezuch from

Poland suggested that I write a circular Array list instead of this self-adapting silly nonsense code, and while we were debating what it would take to give up his summer weather to write such a construct, I received sample code of a partial implementation from [Allard Siemelink](), an independent Java contractor in the Netherlands. It is quite simple writing a basic circular array FIFO queue, but I didn't think a basic solution would be acceptable to my readers.

## Circular Array List

In this newsletter I present an implementation of java.util.List, which I call the CircularArrayList. It can be used exactly as an ArrayList, i.e. you can remove elements from anywhere in the list, you can add elements to the list, get the size, convert it to an array with the toArray method etc. Some of the methods I've left as an exercise to the reader.

So, the question I didn't answer in newsletter 24 was why java.util.ArrayList is so bad as a FIFO queue when the list contains a lot of elements. I assumed that anybody who had subscribed to an **Advanced** Java(tm) newsletter had at least read the source code of that class and already knew the answer. I think my assumption is probably correct, but here is the source of trouble anyway:

```java
public Object remove(int index) {
  RangeCheck(index);

  modCount++;
  Object oldValue = elementData[index];

  int numMoved = size - index - 1;
  if (numMoved > 0)
    System.arraycopy(elementData, index+1, elementData, index,
                     numMoved);
  elementData[--size] = null; // Let gc do its work
  return oldValue;
}
```

In a FIFO queue, we are adding to the back and removing from the front, so if we call remove(0) on a list of 100 elements, numMoved will equal 99, which will cause a call to System.arraycopy(elementData, 1, elementData, 0, 99);. This is the reason why the ArrayList deteriorates as a FIFO queue when it reaches a certain length.

What follows it a CircularArrayList that looks like the ArrayList except that it has a head and a tail that show the first and last elements in the Object element array. In the ArrayList, the head would always be 0. I've written some comments before each method so you can figure out what it does. I recommend that you read the source code of java.util.ArrayList before reading this code.

```java
// CircularArrayList.java
import java.util.*;
import java.io.*;

public class CircularArrayList extends AbstractList
  implements List, Serializable
{
  private Object[] elementData;
  // head points to the first logical element in the array, and
  // tail points to the element following the last.  This means
  // that the list is empty when head == tail.  It also means
  // that the elementData array has to have an extra space in it.
  private int head=0, tail=0;
  // Strictly speaking, we don't need to keep a handle to size,
  // as it can be calculated programmatically, but keeping it
  // makes the algorithms faster.
  private int size=0;
```

```java
      public CircularArrayList() {
        this(10);
      }

      public CircularArrayList(int size) {
        elementData = new Object[size];
      }

      public CircularArrayList(Collection c) {
        tail = c.size();
        elementData = new Object[c.size()];
        c.toArray(elementData);
      }

      // The convert() method takes a logical index (as if head was
      // always 0) and calculates the index within elementData
      private int convert(int index) {
        return (index + head) % elementData.length;
      }

      public boolean isEmpty() {
        return head == tail; // or size == 0
      }

      // We use this method to ensure that the capacity of the
      // list will suffice for the number of elements we want to
      // insert.  If it is too small, we make a new, bigger array
      // and copy the old elements in.
      public void ensureCapacity(int minCapacity) {
        int oldCapacity = elementData.length;
        if (minCapacity > oldCapacity) {
          int newCapacity = (oldCapacity * 3)/2 + 1;
          if (newCapacity < minCapacity)
              newCapacity = minCapacity;
          Object newData[] = new Object[newCapacity];
          toArray(newData);
          tail = size;
          head = 0;
          elementData = newData;
        }
      }

      public int size() {
        // the size can also be worked out each time as:
        // (tail + elementData.length - head) % elementData.length
        return size;
      }

      public boolean contains(Object elem) {
        return indexOf(elem) >= 0;
      }
```

```java
    public int indexOf(Object elem) {
      if (elem == null) {
        for (int i = 0; i < size; i++)
          if (elementData[convert(i)]==null)
            return i;
      } else {
        for (int i = 0; i < size; i++)
          if (elem.equals(elementData[convert(i)]))
            return i;
      }
      return -1;
    }

    public int lastIndexOf(Object elem) {
      if (elem == null) {
        for (int i = size-1; i >= 0; i--)
          if (elementData[convert(i)]==null)
            return i;
      } else {
        for (int i = size-1; i >= 0; i--)
          if (elem.equals(elementData[convert(i)]))
            return i;
      }
      return -1;
    }

    public Object[] toArray() {
      return toArray(new Object[size]);
    }

    public Object[] toArray(Object a[]) {
      if (a.length < size)
        a = (Object[])java.lang.reflect.Array.newInstance(
          a.getClass().getComponentType(), size);
      if (head < tail) {
        System.arraycopy(elementData, head, a, 0, tail-head);
      } else {
        System.arraycopy(elementData, head, a, 0,
          elementData.length-head);
        System.arraycopy(elementData, 0, a, elementData.length-head,
          tail);
      }
      if (a.length > size)
        a[size] = null;
      return a;
    }

    private void rangeCheck(int index) {
      if (index >= size || index < 0)
        throw new IndexOutOfBoundsException(
          "Index: "+index+", Size: "+size);
```

```java
    }

    public Object get(int index) {
      rangeCheck(index);
      return elementData[convert(index)];
    }

    public Object set(int index, Object element) {
      modCount++;
      rangeCheck(index);
      Object oldValue = elementData[convert(index)];
      elementData[convert(index)] = element;
      return oldValue;
    }

    public boolean add(Object o) {
      modCount++;
      // We have to have at least one empty space
      ensureCapacity(size + 1 + 1);
      elementData[tail] = o;
      tail = (tail+1)%elementData.length;
      size++;
      return true;
    }

    // This method is the main reason we re-wrote the class.
    // It is optimized for removing first and last elements
    // but also allows you to remove in the middle of the list.
    public Object remove(int index) {
      modCount++;
      rangeCheck(index);
      int pos = convert(index);
      // an interesting application of try/finally is to avoid
      // having to use local variables
      try {
        return elementData[pos];
      } finally {
        elementData[pos] = null; // Let gc do its work
        // optimized for FIFO access, i.e. adding to back and
        // removing from front
        if (pos == head) {
          head = (head+1)%elementData.length;
        } else if (pos == tail) {
          tail = (tail-1+elementData.length)%elementData.length;
        } else {
          if (pos > head && pos > tail) { // tail/head/pos
            System.arraycopy(elementData, head, elementData, head+1,
              pos-head);
            head = (head+1)%elementData.length;
          } else {
            System.arraycopy(elementData, pos+1, elementData, pos,
              tail-pos-1);
```

```java
            tail = (tail-1+elementData.length)%elementData.length;
        }
    }
    size--;
  }
}

public void clear() {
  modCount++;
  // Let gc do its work
  for (int i=head; i!=tail; i=(i+1)%elementData.length)
    elementData[i] = null;
  head = tail = size = 0;
}

public boolean addAll(Collection c) {
  modCount++;
  int numNew = c.size();
  // We have to have at least one empty space
  ensureCapacity(size + numNew + 1);
  Iterator e = c.iterator();
  for (int i=0; i < numNew; i++) {
    elementData[tail] = e.next();
    tail = (tail+1)%elementData.length;
    size++;
  }
  return numNew != 0;
}

public void add(int index, Object element) {
  throw new UnsupportedOperationException(
    "This method left as an exercise to the reader ;-)");
}

public boolean addAll(int index, Collection c) {
  throw new UnsupportedOperationException(
    "This method left as an exercise to the reader ;-)");
}

private synchronized void writeObject(ObjectOutputStream s)
    throws IOException {
  s.writeInt(size);
  for (int i=head; i!=tail; i = (i+1)%elementData.length)
    s.writeObject(elementData[i]);
}

private synchronized void readObject(ObjectInputStream s)
    throws IOException, ClassNotFoundException {
  // Read in size of list and allocate array
  head = 0;
  size = tail = s.readInt();
  elementData = new Object[tail];
```

```java
      // Read in all elements in the proper order.
      for (int i=0; i < tail; i++)
        elementData[i] = s.readObject();
    }
  }
```

There are some bugs in the code above. Please don't ask me what they are; obviously I don't know what the bugs are, else they would not be there anymore. If you do find a bug, please tell me so that I can send an errata next week. I've tried to test the CircularArrayList against an ordinary ArrayList for correctness, and against an ArrayList and a LinkedList for performance.

```java
// ArrayListTest.java
import java.util.List;
import java.util.ArrayList;
import java.util.LinkedList;

public class ArrayListTest {
  public static String testList(List list) {
    StringBuffer result = new StringBuffer(1024);
    list.add("ABCD");
    list.add("EFGH");
    list.add("IJKL");
    list.addAll(list);
    result.append(list);
    result.append(list.contains("IJKL"));
    result.append(list.containsAll(new ArrayList()
      {{add("ABCD");add("EFGH");}}));
    result.append(list.equals(new ArrayList(list)));
    for (int i=0; i<6; i++)
      result.append(list.get(i));
    result.append(list.indexOf("EFGH"));
    result.append(list.isEmpty());
    result.append(list.lastIndexOf("EFGH"));
    for (int i=0; i<3; i++) result.append(list.remove(3));
    for (int i=0; i<3; i++) result.append(list.remove(0));
    for (int i=0; i<6; i++) list.add(Integer.toString(i));
    for (int i=0; i<6; i++) result.append(list.get(i));
    Object[] els = list.toArray();
    for (int i=0; i<els.length; i++) result.append(els[i]);
    String[] strs = (String[])list.toArray(new String[0]);
    for (int i=0; i<strs.length; i++) result.append(strs[i]);
    for (int i=0; i<32; i++) {
      list.add(Integer.toHexString(i));
      result.append(list.remove(0));
    }
    result.append(list);
    return result.toString();
  }
  public static void testPerformance(List list, int length) {
    Object job = new Object();
    int iterations = 0;
    for (int j=0; j<length; j++) list.add(job);
    long time = -System.currentTimeMillis();
    while(time + System.currentTimeMillis() < 2000) {
```

```
          iterations++;
          for (int j=0; j<100; j++) {
            list.remove(0);
            list.add(job);
          }
        }
        time += System.currentTimeMillis();
        System.out.println(list.getClass() + " managed " +
          iterations + " iterations in " + time + "ms");
    }
    public static void testCorrectness() {
      String al = testList(new ArrayList(6));
      String cal = testList(new CircularArrayList(6));
      if (al.equals(cal)) System.out.println("Correctness Passed");
      else {
        System.out.println("Expected:");
        System.out.println(al);
        System.out.println("But got:");
        System.out.println(cal);
      }
    }
    public static void testPerformance(int length) {
      System.out.println("Performance with queue length = " + length);
      testPerformance(new ArrayList(), length);
      testPerformance(new LinkedList(), length);
      testPerformance(new CircularArrayList(), length);
    }
    public static void main(String[] args) {
      testCorrectness();
      testPerformance(1);
      testPerformance(10);
      testPerformance(100);
      testPerformance(1000);
      testPerformance(10000);
      testPerformance(100000);
    }
  }
```

I've tried this out on my trusty notebook with the stickers on the keys. I again count iterations rather than amount of time it takes, as that gives me far more accurate figures. The ArrayList beats our CircularArrayList if there is only one element on the list, since ArrayList in that case does not have to do any copying of elements with System.arraycopy(). The CircularArrayList has a few overheads more than the ArrayList, which make it marginally slower in that one case. Besides that one case, CircularArrayList beats the stuffing out of the other lists for FIFO queue behaviour, AND it can behave as an ordinary ArrayList as well. This makes me wonder why Sun Microsystems didn't implement ArrayList as a CircularArrayList in the first place? I wish I knew the answer to that.

```
Correctness Passed
Performance with queue length = 1
class java.util.ArrayList managed 145246 iterations in 2003ms
class java.util.LinkedList managed 42385 iterations in 2003ms
class CircularArrayList managed 51058 iterations in 2013ms
Performance with queue length = 10
class java.util.ArrayList managed 51236 iterations in 2003ms
class java.util.LinkedList managed 41188 iterations in 2003ms
```

```
class CircularArrayList managed 54124 iterations in 2003ms
Performance with queue length = 100
class java.util.ArrayList managed 27019 iterations in 2003ms
class java.util.LinkedList managed 39529 iterations in 2003ms
class CircularArrayList managed 52938 iterations in 2004ms
Performance with queue length = 1000
class java.util.ArrayList managed 5300 iterations in 2003ms
class java.util.LinkedList managed 39132 iterations in 2003ms
class CircularArrayList managed 50893 iterations in 2003ms
Performance with queue length = 10000
class java.util.ArrayList managed 535 iterations in 2003ms
class java.util.LinkedList managed 13102 iterations in 2003ms
class CircularArrayList managed 52030 iterations in 2003ms
Performance with queue length = 100000
class java.util.ArrayList managed 9 iterations in 2063ms
class java.util.LinkedList managed 8002 iterations in 2003ms
class CircularArrayList managed 50803 iterations in 2003ms
```

There you go, I hope you bothered to read and try to understand my code. Please remember to forward this newsletter to interested parties.

Heinz

---

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

### 📧 Class names don't identify a class

## Author: Dr. Heinz M. Kabutz

## Copyright 2001 Maxkab Solutions CC - South Africa
## Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 18th issue of "The Java(tm) Specialists' Newsletter", now sent to 38 countries on all continents of our globe. Please remember to forward the newsletter to others who might be interested.

BEFORE YOU FILE THIS NEWSLETTER ...

I have a question for you:

South Africa is a beautiful country with a relatively stable economy, compared to other developing countries. The South- Western part of South Africa, called the Western Cape, is an especially great place to live, mainly because of the stunning environment, weather, etc. For some strange reason, we are losing a lot of programmers to the land of mist and the land of abundance. There are those in the private sector willing to invest significant money to realise their dream of the Western Cape becoming another success story in the computer world.

Now the question: If you used to live in South Africa, what would entice you to come back to the Western Cape? If you are already living in South Africa, what would prevent you from leaving? Lastly, if you are an IT professional living outside of South Africa, what "things" would attact you to move to the Western Cape?

I'm very interested to hear your opinions on this question, so don't delay, answer today :-)

# Class names don't identify a class

This week I want to introduce the concepts of having Class objects in the VM with the exact same name, but being completely different. I had the opportunity to question Dr. Jung, who has been using this for a while now, and at long last, my small brain has made "click" and I understand (I think). In two weeks time, he is going to write the newsletter (he wrote the piece on dynamic proxies) and he will demonstrate how you can build up a sibling hierarchy of class loaders which you can use to automatically redeploy classes and find dependencies, etc. I didn't quite follow all of his explanations, so I'm looking forward to read what he has to say about this topic. In the meantime, to prepare us all, I've written a simple example to demonstrate how it works.

In JDK 1.2, SUN added a new approach to class loading which allows you to identify classes not only by the class name, but also by the context in which it was loaded. We can set the ClassLoader for a Thread, which we can then use to load classes. By having a different classloader, you are effectively constructing a new instance of Class, which is then completely different from other instances of Class. This was, I think, also possible in JDK 1.1, but it is much easier to make an instance of a ClassLoader in JDK 1.2 as there is now a URLClassLoader which you can point to a directory where it will load the classes from.

For example, say we have two directories, a1 and a2. In each of these directories we have a class A:

```
//: a1/A.java
public class A {
  public String toString() { return "This is the first class"; }
}
//: a2/A.java
public class A {
  public String toString() { return "This is the second class"; }
}
```

As you will agree, the two classes are completely different. They can have different method definitions, data members or access control. The normal way of using these two classes is to choose at compile/run time which one you wish to use. For example, we may have a class NormalTest below:

```
//: NormalTest.java
public class NormalTest {
  public static void main(String[] args) {
    System.out.println(new A());
  }
}
```

To compile this class, we have to specify the directory where A resides, either a1 or a2. Since the signature is the same, we can compile with one class and run with the other class if we want.

```
javac -classpath .;a1 NormalTest.java
java -classpath .;a2 NormalTest
```

would result in "This is the second class" being displayed on the console.

What happens if we want to have instances of both A classes in use at the same time? Normally we cannot do that, but if we use ClassLoaders we can.

```
//: Loader.java
import java.net.*;
public class Loader {
  public static void main(String[] args) throws Exception {
    ClassLoader a1 = new URLClassLoader(
      new URL[] {new URL("file:a1/")}, null);
    ClassLoader a2 = new URLClassLoader(
      new URL[] {new URL("file:a2/")}, null);
    Class c1 = a1.loadClass("A");
    Class c2 = a2.loadClass("A");
    System.out.println("c1.toString(): " + c1);
    System.out.println("c2.toString(): " + c2);
    System.out.println("c1.equals(c2): " + c1.equals(c2));
    System.out.println("c1.newInstance(): " + c1.newInstance());
    System.out.println("c2.newInstance(): " + c2.newInstance());
  }
}
```

The two classes, both called "A", are loaded with different ClassLoader objects, and are thus, as far as the VM is concerned, different classes altogether. We can print the names of the classes, compare the two classes

(even though their names are the same, the classes are not equal, you should therefore never compare just the class names if you want to compare classes, rather use the Class.equals() method), and make instances of them by calling the newInstance() method.

The output if we run Loader is:

```
c1.toString(): class A
c2.toString(): class A
c1.equals(c2): false
c1.newInstance(): This is the first class
c2.newInstance(): This is the second class
```

We can also let these two "A" classes have a common superclass. For example, say we have a superclass called Parent.java located in the root directory:

```java
//: Parent.java
public class Parent {
  public String toString() {
    return "Thanks for caring... but what do you want??? ";
  }
}
```

And our A.java classes are now written as:

```java
//: a1/A.java
public class A extends Parent {
  public String toString() {
    return super.toString() + "This is the first class";
  }
}
//: a2/A.java
public class A extends Parent {
  public String toString() {
    return super.toString() + "This is the second class";
  }
}
```

We then need to have a common parent ClassLoader which we use to load the Parent.class file. We have to specify the location to start looking for Parent.class, and the locations are searched in a hierarchical fasion. Note that the compile-time Parent class is loaded with a different ClassLoader to the one loaded with the URLClassLoader called "parent" so they refer to a different class altogether, which means we cannot type-cast an instance of "A" to a Parent. Also, if we load the class "Parent" without using the classloader, we will see that it is not equal to the superclass of "c1".

```java
//: Loader.java
import java.net.*;
public class Loader {
  public static void main(String[] args) throws Exception {
    ClassLoader parent = new URLClassLoader(
      new URL[] {new URL("file:./")}, null);
    ClassLoader a1 = new URLClassLoader(
      new URL[] {new URL("file:a1/")}, parent);
    ClassLoader a2 = new URLClassLoader(
      new URL[] {new URL("file:a2/")}, parent);
    Class c1 = a1.loadClass("A");
    Class c2 = a2.loadClass("A");
    System.out.println(c1.newInstance());
    System.out.println(c2.newInstance());
    System.out.println(
```

```
      c1.getSuperclass().equals(c2.getSuperclass()));
    System.out.println(
      Class.forName("Parent").equals(c1.getSuperclass()));
    try {
      Parent p = (Parent)c1.newInstance();
    } catch(ClassCastException ex) {
      ex.printStackTrace();
    }
    System.out.println("We expected to get ClassCastException");
  }
}
Thanks for caring... but what do you want??? This is the first class
Thanks for caring... but what do you want??? This is the second class
true
false
java.lang.ClassCastException: A
        at Loader.main(Loader.java:18)
We expected to get ClassCastException
```

Note that the super classes of both "A" classes are equal.

Where is this all this useful? It is very useful when you have an application server into which you want to deploy business objects written by different people. It is entirely feasible that you have two developers with different versions of classes deploying their applications onto the same server. You don't necessarily want to start a new VM for each deployment, and so with ClassLoaders it is possible to have lots of classes with the same name running in the same memory space but not conflicting with one another. They would also share the common JDK classes with one another, so we would not have to have a java.util.ArrayList class loaded for each of the ClassLoaders.

Until next week, and please remember to forward this newsletter in its entirety to as many Java users as you know who would be interested.

Heinz

---

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong!
They read our tips every week!   To subscribe to The Java FAQ Daily send empty e-mail to:
javafaq-tips-subscribe@topica.com  or visit at:
http://www.topica.com/lists/javafaq-tips/

Deadlocks in Java

**Author: Dr. Heinz M. Kabutz**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either <u>subscribe via email</u> or <u>subscribe via the web</u>. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to "The Java(tm) Specialists' Newsletter", a low-volume newsletter exploring experiences of using Java "in the field" (not only java.lang.reflect.Field, but that also). The reason I have sent this email to you is because you were either on a Java course presented by me, or you are a colleague, or I thought the topic might simply interest you.

Just to clear things up, as I always tell my students, my company's name is Maximum Solutions and our logo is "The Java(tm) Specialists", hence the title for the newsletter. This does not mean we are good at Java, it simply means that this is all we do (at the moment). Perhaps in future I will change my logo to "The XML Specialists", or, heaven forbid "The VB Specialists", but then I would have to be really hard-up ;-) The logo is therefore not a display of my arrogance (those of you who know me better would assure you that arrogance is definitely not one of my characteristics, neither is a bad haircut nor rolled up jeans) but is supposed to give me focus in the IT industry.

A warning I feel I have to give you is that the tricks I have discovered in Java are not always "kosher" and are used solely at your own risk. Whenever I mention something that is non-kosher I will also attach a warning label so the more experienced of you can use it and the rest not. Always make sure that your unit tests run before and after a change to a new JDK version, the days of JDK 1.1.x were extremely interesting because each version screwed up the South Africa Locale in a different way. For the past 3.5 years I've been programming exclusively in Java and I can say that it is a fascinating language. Once you dig below the surface of Java it becomes increasingly more fun and interesting to work with, and you end up wanting to exploit the limits of the JDK (last week I made a piece of code I had written with someone else 2 years ago 1000x faster). The JDK 1.3.0 source code contained in the src.jar file when you install the JDK contains a lot of Java code, 574253 lines of Java code to be exact, which is slightly less than the Java program written in Stellenbosch in South Africa where I gained most of my Java programming experience.

# Deadlocks in Java

While travelling overseas I had the priviledge of spending some time helping a Java program in dire straits. It was a graphical interface that was supposed to emulate a C++ program running on a server, bla bla bla. Anyway, this application would occasionaly have a thread deadlock with the result that the screen would stop refreshing. Imagine the despair when facing an unfamiliar source base of several 10'000 lines of Java code with a deadlock SOMEWHERE in there?!? The code was well written, but even though, the task was scary. Luckily I had heard somewhere about the secret CTRL+BREAK trick that SUN smuggled into the JDK without telling anybody. If you press CTRL+BREAK while your Java program is running in a command prompt you get a stack trace of exactly what each thread is doing and by looking at the code you easily find which thread is waiting for which lock!

When you get a deadlock in your program you want to be able to reproduce it reliably as quickly as possible, so we got half the team to just concentrate on finding deadlocks that they could reproduce. In the meantime, the other half would look at the stack traces and figure out why the code was deadlocking. In the process of looking we discovered something which I had heard about second-hand but had not encountered myself. Last week I heard of the same problem occuring at another company which took a lot of effort to clear up, so if your program has any GUI in it, especially Swing, it might pay to listen to this:

*In Swing, all GUI components have to be changed from within the Swing thread.*

This means that you cannot execute jLabel1.setText("blabla") from within any thread besides the Swing thread.

If you have change any GUI from another thread you should rather say:

```java
SwingUtilities.invokeLater(new Runnable() {
  public void run() {
    jLabel1.setText("blabla");
  }
}
```

This will do the change that you requested as soon as it can, which is usually within a few microseconds. There is another call invokeAndWait which I have hardly ever used except in race-conditions.

I was under the impression that failure to use invokeLater would cause some refresh problems of a kind where some parts of the screen get refreshed and others not, I did not realise it could cause a thread deadlock (or maybe that is what everyone was talking about - refresh problem - hmpf - more like a system meltdown). Luckily you don't always have to call invokeLater whenever you change a GUI component because in some cases you already are in the Swing GUI thread, for example if a button is pressed then the ActionListener.actionPerformed method will be called from the Swing thread. However, if you provide a callback class to another class that is not part of the AWT/Swing group you will never know what thread it is coming from, so it is safest to invokeLater.

Remember that any work you do in invokeLater is going to hold up the Swing thread from repainting the window, so please don't do big database queries inside invokeLater but rather only call invokeLater for the parts of the code that are genuinely graphics related. It might pay off to bunch all the GUI related lines in your method by refactoring your code a bit. For information on refactoring look at the book with that title by Martin Fowler.

---
*Warning Advanced:*
*A small optimisation is to have a class that figures out if the current thread is the Swing thread (SwingUtilities.isEventDispatchThread()) and if it is not calls SwingUtilities.invokeLater(Runnable), otherwise it calls the runnable code directly. This way you can finish everything you need to do without interference by other threads. Reason for the Non-Kosher label is that it seems you can have several event dispatch threads, I don't know under what circumstances so if you know, please enlighten me. However, it almost seems like only one of them can be active at any one time.*
---

Anyway, this is the end of the newsletter for tonight, so please let me know your comments and feedback and remember to please let me know if I'm wasting your time with this stuff then I'll not send any more to you.

With regards

Heinz

**Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.**

☐ IT Professional ☐ General Computer ☐ Design and Graphics ☐ Linux ☐ Freeware/Shareware
☐ Games     Intranet     Computer Hardware     Web Design     ☐ Computer Software
☐ UNIX ☐ Web Software ☐ Windows Software ☐ Personal Finance ☐ Programming Software
☐ Servers ☐ C/C++ ☐ Powerbuilder ☐ Perl ☐ XML
☐ Java ☐ SQL ☐ Design/Graphics ☐ Utilities ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Depth-first Polymorphism

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 9th issue of "The Java(tm) Specialists' Newsletter", where we look at "in-the-veld" tips and tricks used by Java professionals. I want to thank all of you who respond to these letters, your comments make the late nights writing these newsletters worthwhile :) And those of you who in their day-to-day communication are imitating the style of these newsletters - you know who you are ! - keep trying ;-)

I must apologize that this newsletter is quite long. Unfortunately I am under severe time pressure at the moment so I did not have time to make it shorter.

I was told after last week's newsletter that Java was not based only on C++ and Smalltalk, but also on Lisp ?!? (I did not know that, no wonder it's so terribly slow. I'm surprised the JDK doesn't come with the Emacs editor.) Anyway, I have to thank Michael Wolber from Infor AG in Germany for pointing that out and for his excellent contribution:

**Greenspun's Tenth Rule of Programming:** *"any sufficiently complicated C or Fortran program contains an ad hoc informally-specified bug-ridden slow implementation of half of Common Lisp."*

And Java?

## Depth-first Polymorphism (or Customised Polyseme)

Consider the following class:

```java
public class Polyseme {
  public static class Top {
    public void f(Object o) {
      System.out.println("Top.f(Object)");
```

```
    }
    public void f(String s) {
       System.out.println("Top.f(String)");
    }
  }
  public static void main(String[] args) {
    Top top = new Top();
    top.f(new java.util.Vector());
    top.f("hello");
    top.f((Object)"bye");
  }
}
```

Java looks for the method with the "narrowest" matching class for the parameter objects. Therefore, the output from running this class is:

```
Top.f(Object)
Top.f(String)
Top.f(Object)
```

In Java, the virtual machine tries to find a matching method for your parameters, starting at the top of the hierarchy and moving down. Say we have the following classes:

```
public class BreadthFirst {
  public static class Top {
    public void f(Object o) {
       System.out.println("Top.f(Object)");
    }
  }
  public static class Middle extends Top {
    public void f(String s) {
       System.out.println("Middle.f(String)");
    }
  }
  public static void main(String[] args) {
    Top top = new Middle();
    top.f(new java.util.Vector());
    top.f("hello");
    top.f((Object)"bye");
  }
}
```

The virtual machine will thus start at Top and check if there are any methods which would accept String.class or Object.class, and indeed, Top.f(Object) would handle all those parameters. The output is therefore the following:

```
Top.f(Object)
Top.f(Object)
Top.f(Object)
```

We could "fix" this by overriding f(Object) and using instanceof to call the correct f() method (brrr - I'd rather get stuck on the N2 than do that [for those not living in Cape Town, the N2 is notoriously dangerous, you either get shot at or in or with if your car breaks down])

```
public class BreadthFirstFix {
  public static class Top {
    public void f(Object o) {
       System.out.println("Top.f(Object)");
```

```
      }
    }
    public static class Middle extends Top {
      public void f(Object o) {
        if (o instanceof String)
          f((String)o);
        else
          super.f(o);
      }
      public void f(String s) {
        System.out.println("Middle.f(String)");
      }
    }
    public static void main(String[] args) {
      Top top = new Middle();
      top.f(new java.util.Vector());
      top.f("hello");
      top.f((Object)"bye");
    }
  }
```

The output would now look as we would expect:

```
  Top.f(Object)
  Middle.f(String)
  Middle.f(String)
```

This might have the correct effect, but it does mean that we have to have such a silly "instanceof" in all the subclasses. If we are designing a OO framework we want to have our clients subclass our classes without having to do acrobatics to achieve this.

Christoph Jung mentioned this problem with Java to me a few weeks ago and we thought of some code you could put at the highest level class that uses reflection to start at the lowest class and then tries to match the method to the type before moving up the hierarchy. I call this "depth-first-polymorphism".

```
  import java.lang.reflect.*;
  public class DepthFirst {
    public static class Top {
      private Method getPolymorphicMethod(Object param) {
        try {
          Class cl = getClass();  // the bottom-most class
          // we start at the bottom and work our way up
          Class[] paramTypes = {param.getClass()};
          while(!cl.equals(Top.class)) {
            try {
              // this way we find the actual method
              return cl.getDeclaredMethod("f", paramTypes);
            } catch(NoSuchMethodException ex) {}
            cl = cl.getSuperclass();
          }
          return null;
        }
        catch(RuntimeException ex) { throw ex; }
        catch(Exception ex) { return null; }
      }
```

```java
      public void f(Object object) {
        Method downPolymorphic = getPolymorphicMethod(object);
        if (downPolymorphic == null) {
          System.out.println("Top.f(Object)");
        } else {
          try {
            downPolymorphic.invoke(this, new Object[] {object});
          }
          catch(RuntimeException ex) { throw ex; }
          catch(Exception ex) {
            throw new RuntimeException(ex.toString());
          }
        }
      }
    }
    public static class Middle extends Top {
      public void f(String s) {
        System.out.println("Middle.f(String)");
      }
    }
    public static class Bottom extends Middle {
      public void f(Integer i) {
        System.out.println("Bottom.f(Integer)");
      }
    }
    public static class RockBottom extends Bottom {
      public void f(String s) {
        System.out.println("RockBottom.f(String)");
      }
    }
    public static void main(String[] args) {
      Top top = new RockBottom();
      top.f(new java.util.Vector());
      top.f("hello");
      top.f(new Integer(42));
      top = new Bottom();
      top.f(new java.util.Vector());
      top.f("hello");
      top.f(new Integer(42));
    }
  }
```

The answer is this time:

```
Top.f(Object)
RockBottom.f(String
Bottom.f(Integer)
Top.f(Object)
Middle.f(String)
Bottom.f(Integer)
```

When should you use this technique? Only if you have a lot of specific type handlers as subclasses of a common superclass where it would make sense to add such a depth-first invoker. You can probably extract this functionality and put it in a separate class. If you use this commercially please

do the exception handling correctly, I didn't bother in my example, in preparation for when I change my logo to "The C# Specialists".

Thanks for your comments, I always appreciate your feedback.

Regards

Heinz

---

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

- [ ] IT Professional
- [ ] General Computer
- [ ] Design and Graphics
- [ ] Linux
- [ ] Freeware/Shareware
- [ ] Games
- [ ] Intranet
- [ ] Computer Hardware
- [ ] Web Design
- [ ] Computer Software
- [ ] UNIX
- [ ] Web Software
- [ ] Windows Software
- [ ] Personal Finance
- [ ] Programming Software
- [ ] Servers
- [ ] C/C++
- [ ] Powerbuilder
- [ ] Perl
- [ ] XML
- [ ] Java
- [ ] SQL
- [ ] Design/Graphics
- [ ] Utilities
- [ ] Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📑 Determining Memory Usage in Java

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

A special thanks to Chris Preimesberger of DevX for featuring our SoftHashMap issue of this newsletter on their website.

---

Welcome to the 29th issue of "The Java(tm) Specialists' Newsletter". I could start off with a witty comment about how the newsletter is going to hit the **big three** at the next issue, but I might step on the toes of my old friend (haha) John Green who is turning 30 today - happy birthday! At least I'm not *that* old yet :-) By the time you read the next newsletter, or maybe this newsletter, I will probably be father the second time round.

This week I am showing you one of my most dear trade secrets. Please be very careful who you show this newsletter to, only send it to friends and people on your local JUG. If this gets into the wrong hands, project troubleshooters like me will be out of a job.

One of the fun parts in Java is guessing how much memory is being used by your object. We are conditioned to ignore memory altogether when programming in Java and that can easily land us in trouble. Java does not have a construct like C/C++ that tells us how much space an object is taking, at least until this newsletter...

*Warning:* The results in this newsletter were derived experimentally rather than looking at the innards or the JVM. Please try out the experiments if you are running on a non-WinNT machine and tell me if you get different results.

## Memory Usage in Java

In Java, memory is allocated in various places such as the stack, heap, etc. In this newsletter I'm only going to look at objects which are stored on the heap. Please don't take me to task for not mentioning the others, they might appear in a future newsletter.

Say I have a class Foo, how much memory will one instance of that class take? The amount of memory can be determined by looking at the data members of the class and all the superclasses' data members. The algorithm I use works as follows:

1. The class takes up at least 8 bytes. So, if you say **new** Object(); you will allocate 8 bytes on the heap.
2. Each data member takes up 4 bytes, except for long and double which take up 8 bytes. Even if the data

member is a byte, it will still take up 4 bytes! In addition, the amount of memory used is increased in 8 byte blocks. So, if you have a class that contains one byte it will take up 8 bytes for the class and 8 bytes for the data, totalling 16 bytes (groan!).

3. Arrays are a bit more clever, at least smaller primitives get packed. I'll deal with these later.

In order to be able to test many different types of objects, I have written a MemoryTestBench class that takes an ObjectFactory which is able to create the type of object that you want to test. The MemoryTestBench can either tell you how many bytes are used by that object or it can print out a nicely formatted result for you. You get the most accurate results if you make sure that supplementary memory is already allocated when you start counting. I therefore construct the object, call the methods for finding the memory, and then set the handle to null again. The garbage collector is then called many times, which should free up all unused memory. The memory is then counted, the object created, garbage collected, and the memory counted again. The difference is the amount of memory used by your object, voila!

```java
public class MemoryTestBench {
  public long calculateMemoryUsage(ObjectFactory factory) {
    Object handle = factory.makeObject();
    long mem0 = Runtime.getRuntime().totalMemory() -
      Runtime.getRuntime().freeMemory();
    long mem1 = Runtime.getRuntime().totalMemory() -
      Runtime.getRuntime().freeMemory();
    handle = null;
    System.gc(); System.gc(); System.gc(); System.gc();
    System.gc(); System.gc(); System.gc(); System.gc();
    System.gc(); System.gc(); System.gc(); System.gc();
    System.gc(); System.gc(); System.gc(); System.gc();
    mem0 = Runtime.getRuntime().totalMemory() -
      Runtime.getRuntime().freeMemory();
    handle = factory.makeObject();
    System.gc(); System.gc(); System.gc(); System.gc();
    System.gc(); System.gc(); System.gc(); System.gc();
    System.gc(); System.gc(); System.gc(); System.gc();
    System.gc(); System.gc(); System.gc(); System.gc();
    mem1 = Runtime.getRuntime().totalMemory() -
      Runtime.getRuntime().freeMemory();
    return mem1 - mem0;
  }
  public void showMemoryUsage(ObjectFactory factory) {
    long mem = calculateMemoryUsage(factory);
    System.out.println(
      factory.getClass().getName() + " produced " +
      factory.makeObject().getClass().getName() +
      " which took " + mem + " bytes");
  }
}
```

The ObjectFactory interface looks like this:

```java
public interface ObjectFactory {
  public Object makeObject();
}
```

## Basic Objects

Let's start with the easiest case, a BasicObjectFactory that simply returns a new instance of Object.

```java
public class BasicObjectFactory implements ObjectFactory {
  public Object makeObject() {
    return new Object();
  }
```

```
}
```

When we run this, we get the following output:

```
BasicObjectFactory produced java.lang.Object which took 8 bytes
```

### Bytes

I suggested earlier that bytes are *not* packed in Java and that memory usage is increased in 8 byte blocks. I have written the ByteFactory and the ThreeByteFactory to demonstrate this:

```java
public class ByteFactory implements ObjectFactory {
  public Object makeObject() {
    return new Byte((byte)33);
  }
}

public class ThreeByteFactory implements ObjectFactory {
  private static class ThreeBytes {
    byte b0, b1, b2;
  }
  public Object makeObject() {
    return new ThreeBytes();
  }
}
```

When we run these, we get the following output:

```
ByteFactory produced java.lang.Byte which took 16 bytes
ThreeByteFactory produced ThreeByteFactory$ThreeBytes which took 24 bytes
```

This is great (not). When I first started using Java I used to spend hours deciding whether a variable should be an int or short or a byte in order to minimize the memory footprint. **I was wasting my time.** As I said earlier, I don't know if this is only a problem under NT or if it's the same on all platforms. Knowing Java's dream of being equally inefficient on all platforms, I suspect that it would be the same.

### Booleans

Let's carry on and look at a smaller unit of information, the boolean. Now a boolean is simply a bit, true or false, yes or no, zero or one. If I have a class that contains 64 booleans, guess how much memory it will take? 8 for the class, and 4 for each of the boolean data members, i.e. 264 bytes!!! Since a boolean is essentially the same as a bit, we could have stored the same information in one long. If you don't believe me, have a look at the following class:

```java
public class SixtyFourBooleanFactory implements ObjectFactory {
  private static class SixtyFourBooleans {
    boolean a0, a1, a2, a3, a4, a5, a6, a7;
    boolean b0, b1, b2, b3, b4, b5, b6, b7;
    boolean c0, c1, c2, c3, c4, c5, c6, c7;
    boolean d0, d1, d2, d3, d4, d5, d6, d7;
    boolean e0, e1, e2, e3, e4, e5, e6, e7;
    boolean f0, f1, f2, f3, f4, f5, f6, f7;
    boolean g0, g1, g2, g3, g4, g5, g6, g7;
    boolean h0, h1, h2, h3, h4, h5, h6, h7;
  }
  public Object makeObject() {
    return new SixtyFourBooleans();
  }
}
```

When we run this, we get the following output:

```
SixtyFourBooleanFactory produced SixtyFourBooleanFactory$SixtyFourBooleans
  which took 264 bytes
```

Admittedly, the example was a little bit contrived, as you would seldom have that many booleans in one class, but I hope you get the idea.

Sun must have realised this problem so they made constants in java.lang.Boolean for TRUE and FALSE that both contain instances of java.lang.Boolean. I think that the constructor for Boolean should have been private to stop people from creating 16 byte objects that are completely unnecessary.

## Arrays of Boolean Objects

A Boolean Array takes up 16 bytes plus 4 bytes per position with a minimum of 8 bytes at a time. In addition to that, we obviously have to count the actualy space taken by Boolean objects.

```java
public class BooleanArrayFactory implements ObjectFactory {
  public Object makeObject() {
    Boolean[] objs = new Boolean[1000];
    for (int i=0; i<objs.length; i++)
      objs[i] = new Boolean(true);
    return objs;
  }
}
```

Try guess how many bytes would be taken up by a Boolean array of size 1000 with Boolean objects stuck in there. Ok, I'll help you: 16 + 4*1000 (for the pointers) + 16*1000 (for the actual Boolean objects) = 20016. Run the code and see if I'm right ;-) If we, instead of making a new Boolean object each time, use the Flyweights provided in Boolean, we'll get to 16 + 4*1000 = 4016 bytes used.

Primitives get packed in arrays, so if you have an array of bytes they will each take up one byte (wow!). The memory usage of course still goes up in 8 byte blocks.

```java
public class PrimitiveByteArrayFactory implements ObjectFactory {
  public Object makeObject() {
    return new byte[1000];
  }
}
```

When we run this, we get the following output:

```
PrimitiveByteArrayFactory produced [B which took 1016 bytes
```

## java.lang.String

Strings actually fare quite well since they can be "internalised" meaning that only one instance of the same String is kept. If you, however, construct your String dynamically, it will not be interned and will take up a bit of memory. Inside String we find:

```java
// ...
private char value[];
private int offset;
private int count;
private int hash = 0;
// ...
```

Say we want to find out how much "Hello World!" would take. We start adding up 8 (for the String class) + 16 (for the char[]) + 12 * 2 (for the characters) + 4 (value) + 4 (offset) + 4 (count) + 4 (hash) = 64 bytes. It's quite difficult to measure this, as we have to make sure the String is not internalized by the JVM. I used the StringBuffer to get this right:

```java
public class StringFactory implements ObjectFactory {
  public Object makeObject() {
    StringBuffer buf = new StringBuffer(12);
    buf.append("Hello ");
    buf.append("World!");
    return buf.toString();
  }
}
```

When we run this, we get, as expected, the following output:

```
StringFactory produced java.lang.String which took 64 bytes
```

## java.util.Vector

Now we get to the real challenge: How much does a java.util.Vector use in memory? It's easy to say, now that we have a MemoryTestBench, but it's not so easy to explain. We start by looking inside the java.util.Vector class. Inside we find the following:

```java
// ...
protected Object elementData[];
protected int elementCount;
// ...
```

Using the knowledge we already have, we decide that the amount of memory used will be 8 (for the class) + 4 (for the pointer to elementData) + 4 (for elementCount). The elementData array will take 16 (for the elementData class and the length) plus 4 * elementData.length. We then follow the hierarchy up and discover the variable $int$ $modCount$ in the superclass java.util.AbstractList, which will take up the minimum 8 bytes. For a Vector of size 10, we will therefore take up: 8 + 4 + 4 + 16 + 4*10 + 8 = 80 bytes, or simply 40 + 4*10 = 80 bytes, which agrees with our experiment:

```java
public class VectorFactory implements ObjectFactory {
  public Object makeObject() {
    return new java.util.Vector(10);
  }
}
```

When we run this, we get the following output:

```
VectorFactory produced java.util.Vector which took 80 bytes
```

So, what happens when we create a JTable with a DefaultTableModel with 100x100 cells? The DefaultTableModel keeps a Vector of Vectors so this will take 40 + 4*100 + (40 + 4*100) * 100 = 440 + 44000 = 44440 bytes just for the empty table. If we put an Integer in each cell, we will end up with another 100*100*16 = 160'000 bytes used up.

## java.util.LinkedList

What's better, a java.util.LinkedList or a java.util.ArrayList? Experienced followers of these newsletters will of course say: "Neither, the CircularArrayList is better" ;-). Let's see what happens when we put 10000 objects into an ArrayList (which uses the same amount of memory as the Vector) vs. a LinkedList. Remember that each Object takes up 8 bytes, so we will subtract 80000 bytes from each answer to get comparable values:

```java
import java.util.*;
public class FullArrayListFactory implements ObjectFactory {
  public Object makeObject() {
    ArrayList result = new ArrayList(10000);
    for (int i=0; i<10000; i++) {
      result.add(new Object());
    }
    return result;
  }
}

import java.util.*;
public class FullLinkedListFactory implements ObjectFactory {
  public Object makeObject() {
    LinkedList result = new LinkedList();
    for (int i=0; i<10000; i++) {
      result.add(new Object());
    }
    return result;
  }
}
```

```
        }
```

When we run this, we get the following output:

```
    FullArrayListFactory produced java.util.ArrayList which took 120040 bytes
    FullLinkedListFactory produced java.util.LinkedList which took 320048 bytes
```

When we subtract 80000 bytes from each, we find that the ArrayList takes up 40040 bytes (as expected) and the LinkedList uses 240048 bytes. How many of us consider issues like this when we code?

We have come to the end of yet another newsletter. I am trying to put newsletters together that will be worthwhile to send out, so as a result they will not always appear every week, unless I feel particularly inspired.

Until the next issue...

Heinz

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

### Dynamic Proxies - Short Tutorial

## Author: Dr. Christoph G. Jung

## Copyright 2001 Maxkab Solutions CC - South Africa
## Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to our fifth "The Java(tm) Specialists' Newsletter", published by Maximum Solutions, the company who may not be good at Java(tm), but Java is all we do, so we call ourselves "Java Specialists". Thank you for not unsubscribing in hordes (hmmm, I have not told you HOW to unsubscribe, so maybe that's why ;-)

Isn't it funny how employment agencies ask you ALL the things you know, like "which word processor have you used?" and "what version of Turbo Pascal do you know?" What relevance is that to today's newsletter? None.

Anyway, we've had an interesting week sorting out small Java problems and fixing bugs in my software. Many thanks to Dr. Christoph for writing this week's newsletter, which I really enjoyed reading. For this I owe you an Esplendido! Christoph started writing Java code at the end of 1999 for a very innovative ERP company in Germany called infor Business Solutions AG, and has quickly become a real fundi at writing application server code. Prior to that, he did a doctorate in Computer Science with emphasis on Robotics and Artificial Intelligence. I don't know where Java comes in, but a PhD is general education that is useful for many spheres of life, and it does not really matter what your topic was. Believe me, my PhD topic was even less useful ;-)

Anyway, enough from me, here is a "Short Tutorial to the Java2 platform Dynamic Proxy Facilities" ....... I wish we had known about this before we started pasting access control onto our 570'000 line application ......

# Short Tutorial to the Java2(TM) platform Dynamic Proxy Facilities

# C.G.Jung (Christoph.Jung@infor.de)

## A Bad Example

One the most useful design patterns when it comes to building extensible frameworks turns out to be the PROXY pattern. For example, imagine your notoriously paniqued project manager insisting on your introducing new security features into your companies existing business model:

```java
/**
 * A Class representing a single salary
 */
class Salary {
  Employee employee;
  long amount;

  public long getAmount() {
    return amount;
  }
}
```

Your straightforward approach to shield the objects from exporting sensible information, such as exorbitant management salaries, to ordinary users, such as you humble worm, would be to introduce a dedicated SecurityChecker (this is not a java.lang.Security tutorial, so please excuse the following very naive code):

```java
/**
 * Realises The Systems Security Policies
 */
class SecurityChecker {
  java.security.Permission manyBucksPermission;

  long check(long amount) throws SecurityException {
    if(amount>1000000)
      SecurityManager.checkPermission(manyBucksPermission);
    return amount;
  }
}
/**
 * A Modified Salary that is "Too-Much-Bucks"-Aware
 */

class Salary {
  Employee employee;
  long amount;
  // ugly reference
  static SecurityChecker sc;

  public long getAmount() {
    // isnt this the wrong place for doing that?
    return sc.check(amount);
  }
}
```

Now your project manager is happy because of being able to ask for his next salary increase. But, you (and your colleague that is the application but not the framework specialist) will quite likely curse him three times when he returns with next terms project plan that includes

1. heavily extending the business logic (amount will no more be stored immediately, but computed by a very complex, profile-oriented function, an additional int getAmount() is needed in Salary, an Order should be implemented, etc.)

2. expanding the successfully introduced security policies (maybe a check(Customer customer) should be added that restricts access to data associated with particular "high-sensitive" customers).

The reason for your deadly wishes is of course that in the above solution, we have heavily mixed up application data and methods (such as amount and long getAmount() ) with generic access control (such as check(long amount)) Extending and updating the 200 classes with 1000 methods of application logic involves understanding of when and how to correctly apply the available security checks. Because only you as the framework pro are able to do that, you will end up with a quite suboptimal division of responsibility and code.

## The Proxy Solution

In order to free your colleague from the heavy burden of the security framework and enable him to concentrate on his business, the better choice is hence to leave his Salary class alone, but rather provide him with an inherited PROXY class that intercepts any getAmount() calls to interface the security framework:

```
/**
 * proxy that behaves as a salary
 * and that performs a security check
 * before delegation
 */

class SalarySecurityProxy extends Salary {
  // this is not application, hence the reference is ok
  static SecurityChecker sc;
  // the object to which we delegate the calls to
  Salary realObject;

  public long getAmount() {
    // framework call after delegation
    return sc.check(realObject.getAmount());
  }
}
```

Now, your collegue is free to change and extend the application logic to realise project goal a (e.g., manipulating amount and getAmount()), while you can add additional independently further proxies and intercepting/ delegation calls to pursue project goal b. Strike.

If your initial application design has been smart by using mucho, mucho interfaces (you know that you should do that anyway, don't you?), PROXY can be even made compatible with complex inheritances:

```
/** use interfaces everywhere in your app-logic */
interface Salary {
...
}
/** and provide proper implementations */
class SalaryImpl implements Salary{
...
}
/** as well as framework proxies */
class SalarySecurityProxy implements Salary{
...
}
/** when you extend your model ... */
interface ManagementSalary extends Salary {
```

```
    ...
    }
    /** ... afterwards ... */
    class ManagementSalaryImpl extends SalaryImpl {
    ...
    }
    /** ... it's very easy */
    class ManagementSalarySecurityProxy extends SalarySecurityProxy {
    ...
    }
```

Double Strike.

The possibilities of the proxy are nearly endless and range from transparent SECURITY over LAZY RETRIEVAL, POOLING, CACHING, and TRANSACTIONAL MARKUP up to DISTRIBUTION (remember RMI stubs?). It is highly useful to combine such differently focused proxies in a CHAIN OF RESPONSIBILITY to a complex, at the same time extensible framework, e.g., SalarySecurityProxy (performs security checks) --> SalaryPersistenceProxy (retrieves the "real" salary from a database) --> Salary

Often, PROXY is used in combination with the FACTORY pattern in order to allow the transparent insertion of proxies and chains into the application logic (your colleague would'nt be bothered by their construction at all ... no more endless discussions ... no more fruitless explanations ... must be like heaven).

## Why Dynamic Proxies are needed

However, the main drawback of PROXY is that for each application class, each application method, and each additional framework functionality, you have to implement and maintain (sic!) a dedicated interceptor class/method. Depending on the output of your colleague, this can be quite a pain in the ass ... (BTW: I'm sure that some of you already collided with the winding RMIC procedure which actually is an automated proxy-generator and -compiler not to speak of its ubiquitious stub-deployment problem)

If reminding the mostly general nature of framework code, you ask yourself whether there is not a better solution to this issue. Actually, there is. It is called DYNAMIC PROXIES and one of the wonderful invents that came to your harddisk with your JRE/JDK1.3 installation.

DYNAMIC PROXIES heavily rely on the already matured reflection capabilities of the Java2 platform. They separate the "interception logic", that is the part of simulating an arbitrary interface, from the "invocation logic", that is the part of calling the framework before or after delegating any method call to the proper application target.

The "invocation logic", such as to connect to a method-based security checker, is realised on purpose by implementing the freshly introduced java.lang.reflect.InvocationHandler interface (more specifically, its reflection-based invoke-Method):

```
    /**
     * a generic, method-based security handler
     * that is not only useful for Salaries
     */

    class SecurityHandler extends InvocationHandler {
      static SecurityChecker sc;
      final Object realObject;

      /** contructor accepts the real subject */
      public SecurityHandler(Object real) {
        realObject=real;
      }
```

```java
    /** a generic, reflection-based secure invocation */
    public Object invoke(Object target,
        java.lang.reflect.Method method, Object[] arguments)
        throws Throwable {
      try{
        // call framework and then reflect the app-logic
        return sc.check(method).invoke(realObject,arguments);
      } catch(java.lang.reflect.InvocationTargetException e) {
        // reconvert nested application exceptions
        throw e.getTargetException();
      }
    }
  }
```

The "interception logic" is already hardwired into the 1.3 vm and the new java.lang.reflect.Proxy class. All it requires to combine it with the above SecurityHandler code is to call the static:

```java
  ManagementSalary salary;

  ManagementSalary smellsLikeSalary=(ManagementSalary)
    java.lang.reflect.Proxy.newProxyInstance(
      Salary.class.getClassLoader(),
      new Class[] {ManagementSalary.class},
      new SecurityHandler(salary));
```

The thus constructed smellsLikeSalary object now behaves like an implementation of ManagementSalary. In fact, it *IS* an implementation of ManagementSalary whose class has been dynamically constructed at runtime, hence dynamically. You can inspect it (smellsLikeSalary instanceof ManagementSalary), cast it ((Salary) smellsLikeSalary) and call methods on it (smellsLikeSalary.getAmount()). Method calls will in turn be dispatched to the invoke method of the given invocation handler (where in our example, target points to smellsLikeSalary, method points to the internal Salary.getAmount() representation, and the arguments array is empty). Return values and exceptions will be automatically converted/casted into the compatible types as given in the interface signature. That's it.

(No, actually that's not: java.lang.reflect.Proxy has a lot more advanced features and there are some tricky considerations when building proxies for multiple/conflicting interfaces - but that's too much for now, please refer to the API documentation for these issues.)

If PROXY has broad applicability, this holds for DYNAMIC PROXIES even more. For example, you could imagine a PersistenceHandler that hides JDBC-access to a row in a relational database table without requiring any additional object instances. For example, you could think of a JRMPRemoteHandler that implements stub-less (HHHOOOORRRAAAYYYY ... no more RMIC) RMI-access to a remote object via the TCP/IP-based Java Remote Method Protocol.

The latter usage is especially supported by DYNAMIC PROXIES, such as smellsLikeSalary actually being serializable, if the tied handler classes are. Under http://www.dreambean.com/download/rickard/SmartWorld-1.2.zip you will find executable code and documentation that extends SUN-JRMP1.2 with suitable handlers that can be distributed through naming services, files, etc. and that can be straightforwardly enriched by load-balancing, failover, clustering, and connection-pooling features.

The first (Open Source!) EJB product that uses these handlers instead of remote stubs is jBoss whose complementary server container design is indeed realised as flexible CHAIN OF RESPONSIBILITY ... Long live the (DYNAMIC) PROXIES!

---

Thanks for that, Christoph...

Next week will be a bit shorter and I will show you how to put method bodies in Java interfaces, which you should never need to do, IF the framework you are working within is perfect. In a less-than-perfect world such a technique can be useful, as fellow fish-chaser Niko Brummer told me. So, hold your breath until

next week.

Regards

Heinz and Christoph

---

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

- ☐ IT Professional
- ☐ General Computer
- ☐ Design and Graphics
- ☐ Linux
- ☐ Freeware/Shareware
- ☐ Games
- Intranet
- Computer Hardware
- Web Design
- ☐ Computer Software
- ☐ UNIX
- ☐ Web Software
- ☐ Windows Software
- ☐ Personal Finance
- ☐ Programming Software
- ☐ Servers
- ☐ C/C++
- ☐ Powerbuilder
- ☐ Perl
- ☐ XML
- ☐ Java
- ☐ SQL
- ☐ Design/Graphics
- ☐ Utilities
- ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📕  Final Methods

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 25th issue of "The Java(tm) Specialists' Newsletter". I hope that for at least *some* of you, your heart sank when you saw the title of this newsletter. No, your mailbox is not getting lighter, I just thought I'd write a bit about how I use the "final" keyword. Incidentally, on Monday we broke through the 1000th reader barrier (on an upward trend) so thanks to all of you who promoted this newsletter and sent it to friends and colleagues. Please remember to forward this newsletter to as many Java enthusiasts as you know who might be interested in receiving such a newsletter.

The last few newsletters where quite heavy, so this week I would like to look at style, specifically on uses of the "final" keyword. It is not a newsletter on how to abuse the "final" keyword, which might surprise some of the more loyal readers of this newsletter.

I would like to thank Carl Smoritcz from Germany for hosting an archive of my newsletters at http://www.smotricz.com/kabutz. Please let me know if you would like to include an archive on your website. I am currently in discussion with a designer to put together a website for my company, which will include the newsletters and information about the type of work my company does.

## Final

The keyword "final" in Java is used in different ways depending on the context. We can have final methods, final classes, final data members, final local variables and final parameters. A final class implicitly has all the methods as final, but not necessarily the data members. A final class may not be extended, neither may a final method be overridden.

Final primitive data members cannot be changed once they are assigned, neither may final object handle data members (Vector, String, JFrame, etc.) be reassigned to new instances, but if they are mutable (meaning they've got methods that allow us to change their state), their contents may

be changed. Since String is immutable, once a handle to it is final, we *could* consider it as a constant, if we ignore the effects of newsletter 14. So how do we use this construct in the real world?

## Final Methods

I personally try to avoid making a method final, unless there is a very good reason for it to be final. There are typically two reasons to make a method final, performance and design. Let's look at performance first:

When a method is final, it may be inlined. Before HotSpot compiling (JDK 1.1.x), these methods were usually inlined at compile time, whereas with HotSpot they are inlined at runtime, unless the compiler can guarantee that the inlined method will always be compiled together with the code that uses it.

```java
// somebody else's class
public class A {
  public static final void f() {
    System.out.println("A's f()");
  }
}


// our class
public class B {
  public void g() {
    A.f();
  }
}
```

In the past, at compile time our class would be turned into:

```java
// compiled class
public class B {
  public void g() {
    System.out.println("A's f()");
  }
}
```

The effect of this was that we had to make one less method call, and since method calls produce extra overhead, we saved some clock cycles. The disadvantage of this, made clear to me by an old (ok, experienced) COBOL programmer during one of my courses, was that whenever somebody else's class changed we would have to remember to recompile our class!

In JDK 1.[234].x with HotSpot(tm), Sun changed this so that the methods were no longer inlined at compile time, but rather by the HotSpot compiler at run time, IFF the performance measurements suggested that it would improve the overall performance of our code.

There are quite a few factors which will affect whether a method will be inlined or not, and we cannot assume that just because we make something final that it will definitely be inlined. As we saw in newsletter 21, it is a good idea anyway to always recompile all your code when you get a new version of someone else's library, so this is not necessarily a reason to NOT use final methods.

When you make a method final, no-one else will be able to override it again. You thus limit extensibility of your code by choosing to make the method or, even worse, your class final. I have been utterly frustrated in the past when I wanted to extend code where the developer had tried to add optimizations in the form of final. I thus never make a method or class final unless I specifically want to stop do others from overriding them.

So, when do I use final methods? If I have performance values that prove that final makes a difference then I would consider using it for performance reasons, otherwise I would only ever use

it for design reasons.

This is all old hat for you I'm sure, so let's look at final data members:

# Final data members

One of the difficulties in programming is coming up with good names for "things". (there, that just proves my point, doesn't it?) I remember an experienced (ok, old) C programmer who was programming in Java and decided to use very long names for everything, for example:

```java
public class SessionConnectorWithRetryAtLeastThreeTimes {
    private String connectionNameReceivedFromInternet;
    private int numberOfTimesThatWeShouldRetryAtLeast;
}
```

Alright, I'm exaggerating a little bit, but I hope you get the idea. The beauty of good names is that comments become very easy to write, sometimes even partly redundant. In Java, we can then write a constructor that takes the state for the object and assigns the correct data members. For example:

```java
public class SessionConnectorWithRetryAtLeastThreeTimes {
    private String connectionNameReceivedFromInternet;
    private int numberOfTimesThatWeShouldRetryAtLeast;
    public SessionConnectorWithRetryAtLeastThreeTimes(
        String c, int n) {
      connectionNameReceivedFromInternet = c;
      numberOfTimesThatWeShouldRetryAtLeast = n;
    }
}
```

The problem with our constructor is that we have to explain in our documentation what c and n represent. It would be much better to use the same names in the parameters of the constructor as we use for the data members, as it reduces the confusion. The standard way in Java of solving this problem is to use the same names for the parameters as we do for the data members and then to explicitely specify what we are referring to, using the "this" keyword.

```java
public class SessionConnectorWithRetryAtLeastThreeTimes {
    private String connectionNameReceivedFromInternet;
    private int numberOfTimesThatWeShouldRetryAtLeast;
    public SessionConnectorWithRetryAtLeastThreeTimes(
        String connectionNameReoeivedFromInternet,
        int numberOfTimesThatWeShouldRetryAtLeast) {
      this.connectionNameReceivedFromInternet =
        connectionNameReceivedFromInternet;
      this.numberOfTimesThatWeShouldRetryAtLeast =
        numberOfTimesThatWeShouldRetryAtLeast;
    }
}
```

The above code will compile and run, but not correctly. Take a few minutes to figure out what could possible be wrong with it...

.

.

.

.

I hope you didn't find the mistake. The first parameter of the constructor is spelt differently to the

data member, thanks to a simple spelling mistake. When we thus say

```
this.connectionNameReceivedFromInternet =
    connectionNameReceivedFromInternet;
```

both the names refer to the data member, so the data member will always null!

This is the reason why some companies try to pursuade their staff to augment their data members with strange characters (m_ or _) to differentiate them from parameters, rather than use the "this" trick. I know of at least two companies where such coding standards are used. The effect is that either the data members look ugly, or the parameters look ugly.

A simple way of preventing such mistakes, besides learning to touch type 100% correctly, is to make the data members final, where possible. When we do that, the code below will no longer compile, and we can find our mistakes much easier.

```
public class SessionConnectorWithRetryAtLeastThreeTimes {
    private final String connectionNameReceivedFromInternet;
    private final int numberOfTimesThatWeShouldRetryAtLeast;
    public SessionConnectorWithRetryAtLeastThreeTimes(
            String connectionNameReoeivedFromInternet,
            int numberOfTimesThatWeShouldRetryAtLeast) {
        this.connectionNameReceivedFromInternet =
            connectionNameReceivedFromInternet;
        this.numberOfTimesThatWeShouldRetryAtLeast =
            numberOfTimesThatWeShouldRetryAtLeast;
    }
}
```

As a matter or habit, I make all data members final wherever that is possible. Mistakes that would have taken me days to find now pop out at the next compile.

## Final local variables

There are two reasons I know for making a local variable or a parameter final. The first reason is that you don't want your code changing the local variable or parameter. It is considered by many to be bad style to change a parameter inside a method as it makes the code unclear. As a habit, some programmers make all their parameters "final" to prevent themselves from changing them. I don't do that, since I find it makes my method signature a bit ugly.

The second reason comes in when we want to access a local variable or parameter from within an inner class. This is the actual reason, as far as I know, that final local variables and parameters were introduced into the Java language in JDK 1.1.

```
public class Access1 {
    public void f() {
        final int i = 3;
        Runnable runnable = new Runnable() {
            public void run() {
                System.out.println(i);
            }
        };
    }
}
```

Inside the run() method we can only access i if we make it final in the outer class. To understand the reasoning, we have to look at what the compiler does. It produces two files, Access1.class and Access1$1.class. When we decompile them with JAD, we get:

```
public class Access1 {
    public Access1() {}
```

```
    public void f() {
      Access1$1 access1$1 = new Access1$1(this);
    }
  }
```

and

```
  class Access1$1 implements Runnable {
    Access1$1(Access1 access1) {
      this$0 = access1;
    }
    public void run() {
      System.out.println(3);
    }
    private final Access1 this$0;
  }
```

Since the value of i is final, the compiler can "inline" it into the inner class. It perturbed me that the local variables had to be final to be accessed by the inner class until I saw the above.

When the value of the local variable can change for different instances of the inner class, the compiler adds it as a data member of the inner class and lets it be initialised in the constructor. The underlying reason behind this is that Java does not have pointers, the way that C has.

Consider the following class:

```
  public class Access2 {
    public void f() {
      for (int i=0; i<10; i++) {
        final int value = i;
        Runnable runnable = new Runnable() {
          public void run() {
            System.out.println(value);
          }
        };
      }
    }
  }
```

The problem here is that we have to make a new local data member each time we go through the for loop, so a thought I had today while coding, was to change the above code to the following:

```
  public class Access3 {
    public void f() {
      Runnable[] runners = new Runnable[10];
      for (final int[] i={0}; i[0]<runners.length; i[0]++) {
        runners[i[0]] = new Runnable() {
          private int counter = i[0];
          public void run() {
            System.out.println(counter);
          }
        };
      }
      for (int i=0; i<runners.length; i++)
        runners[i].run();
    }
    public static void main(String[] args) {
      new Access3().f();
```

```
        }
    }
```

We now don't have to declare an additional final local variable. In fact, is it not perhaps true that int[] i is like a common C pointer to an int? It took me 4 years to see this, but I'd like to hear from you if you have heard this idea somewhere else.

I always appreciate any feedback, both positive and negative, so please keep sending your ideas and suggestions. Please also remember to take the time to send this newsletter to others who are interested in Java.

Heinz

---

**Errata**

In newsletter 23, I relied on hearsay for some information without checking it out myself. Yes, the amount of memory used by each Thread for its stack was incorrect. According to new measurements, it seems that each stack takes up approximately 20KB, not the 2MB stated in the original newsletter, so for 10000 threads we will need about 200MB. Most operating systems cannot handle that many threads very well anyway, so we want to avoid creating that many. Thanks to Josh Rehman for pointing out this mistake.

It seems that with JDK 1.1.8 each thread takes up 145KB, I'm not sure whether the stacks grow dynamically. Under the JDK 1.2.2 that comes with JBuilder 3, the stacks grow, so I managed to have 100 threads take up roughly 100 MB of memory, or 1MB each. After 1 MB is used up, the VM mysteriously returns without any message so I had to experiment a bit to get this information. Under JDK 1.3.0 I got DrWatsons when I tried to make the stack of each thread grow to any real size. It is unrealistic to expect our program to have stack depths of 10000 method calls, so we could probably quite safely use 50KB as a realistic stack size for each thread.

---

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📖 Finding Lost Frames

**Author: Dr. Heinz M. Kabutz**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 19th issue of "The Java(tm) Specialists' Newsletter". This newsletter is my contribution to advancing Java skills, and is meant as an informative, advanced newsletter that hopefully will tell you things you won't find in your average newsletter. Please let me know when my content becomes too ordinary or my style too stolid.

Thanks to all those of you who responded to last week's question, there were some interesting opinions flying around, seems like the real reasons why people go/come/stay in places is because of relationships and friendships, with money, technology, etc. only being secondary. With apologies to my readers from the erstwhile Soviet Union, it seems that was quite a well-known fact to the Politburo, but perhaps I've read too many spy stories.

## Finding Lost Frames

Something that I have encountered in some almost-complete projects, were dialogs that were not bound to frames. When you construct a dialog, you should specify who the owner is, which is either a frame or another dialog. It is, however, also possible to specify "null" as the owner, which causes very irritating problems. In MS Windows, if a modal dialog does not have an owner, and you have moved away from it, for example if you quickly switched to Outlook while waiting for it to start up, the only way to get back to the dialog is by using ALT+Tab. If you click on the frame icon on the toolbar, you will get just the frame, not the dialog. Clicking on the frame will cause a beep and that's all.

A few months ago, I asked myself the question: Is there a way in which we can find a frame that has already been created?

One of my first newsletters showed a GlobalHotkeyManager that allowed you to install your own event queue into the AWT, letting you catch any events that occur, before ANY other components get hold of them. Why don't we use the concept to catch Window events and then use those events to remember which frames are available?

The problem with the GlobalHotkeyManager, was that it only allowed exactly one event catcher to be present at a time. It is therefore only safe to use if none of the libraries you use link in a similar event queue. An alternative, suggested by F.S., was to register ourselves as a listener to the main AWT event system by calling Toolkit.getDefaultToolkit().addAWTEventListener().

Each time a window is activated, we grap the handle and add it to our list of frames that we know about. We can then write a method called "lookupFrame(String title)" which goes through the list and returns the first frame that we find with the specified title.

```java
//: FrameLookup.java
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.util.*;

public class FrameLookup {
  private final Collection frames = new LinkedList();
  // Singleton Pattern
  private static final FrameLookup instance = new FrameLookup();
  public static FrameLookup getInstance() {
    return instance;
  }
  private FrameLookup() {
    Toolkit.getDefaultToolkit().addAWTEventListener(
      new AWTEventListener() {
        public void eventDispatched(AWTEvent event) {
          System.out.println("Event Dispatched : " + event);
          if (event.getID() == WindowEvent.WINDOW_ACTIVATED) {
            if (event.getSource() instanceof Frame) {
              synchronized(frames) {
                frames.add(event.getSource());
              }
            }
          }
        }
      }, AWTEvent.WINDOW_EVENT_MASK);
  }
  public Frame lookupFrame(String title) {
    synchronized(frames) {
      Iterator it = frames.iterator();
      while(it.hasNext()) {
        Frame frame = (Frame)it.next();
        if (frame.getTitle().equals(title)) return frame;
      }
    }
    return null;
  }
}
```

I can now write a test program that creates a frame, forgets the handle, and then creates a dialog and uses the FrameLookup class to find the correct frame to use as owner.

```java
//: FrameLookupTest.java
import javax.swing.*;
import java.awt.*;
```

```java
public class FrameLookupTest {
  private static final String SWING_FRAME_TITLE = "Swing Frame";
  private static final String AWT_FRAME_TITLE = "AWT Frame";
  private static final String SWING_DIALOG_TITLE = "Swing Dialog";

  public static void main(String[] args) {
    FrameLookup framer = FrameLookup.getInstance();
    makeFrame();

    // we can find visible frame by using our FrameLookup utility
    System.out.println("Frame is " +
      FrameLookup.getInstance().lookupFrame(SWING_FRAME_TITLE));

    makeDialog();
  }

  private static void makeFrame() {
    JFrame frame = new JFrame(SWING_FRAME_TITLE);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.setSize(300, 300);
    frame.show();
  }

  private static void makeDialog() {
    // we can bind our JDialog to the Frame
    JDialog dialog = new JDialog(
      FrameLookup.getInstance().lookupFrame(SWING_FRAME_TITLE),
      SWING_DIALOG_TITLE, true);
    dialog.setSize(400, 100);
    dialog.show();
  }
}
```

This is great; we can discover the correct frame as owner, just by knowing its title! There are some minor problems, such as the fact that there is a racing condition involved. It can happen that we request to find the frame before it has been activated or shown, thereby our lookupFrame() method would return "null". Big Oooops. Our situation is now worse than before, as such intermittent failings are MUCH harder to find. Let's also not forget that the title of a frame does not have to be unique.

When I was preparing this newsletter, I got distracted in that I tried to reproduce the Swing dealock problem (which I didn't manage :(. While I was looking through the source code of JFrame/Frame/Window/Container/Component, I happened upon the method Frame[] java.awt.Frame.getFrames(), which Sun added in the JDK 1.2, according to the @version tag. Is it just me, or have I been reading the wrong publications, that don't tell me about these features? Please, if you have heard of Frame.getFrames(), send me an email.

Anyway, with my newly acquired knowledge, I ran off and wrote a second version of the FrameLookup class that looks like this, and sorts out the racing condition problem:

```java
//: FrameLookup.java take 2
import java.awt.*;
public class FrameLookup {
  // Singleton Pattern
  private static final FrameLookup instance = new FrameLookup();
  public static FrameLookup getInstance() {
```

```java
        return instance;
    }
    private FrameLookup() { }
    public Frame lookupFrame(String title) {
        Frame[] frames = Frame.getFrames();
        for (int i=0; i<frames.length; i++) {
            if (frames[i].getTitle().equals(title))
                return frames[i];
        }
        return null;
    }
}
```

We still have the problem that titles alone could be ambigious, but the getFrames() method is definitely the correct way to find a Frame if you need to.

Until next week, and please remember that the more people read this newsletter each week, the more corporate time I can waste collectively, currently I'm wasting about 2 man-months of your development time each week. i.e. please keep on forwarding these newsletters to friends and colleagues who use Java ;-)

Heinz

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📑 Hooking into the shutdown call

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

I am sitting outside in a mild South African summer evening, looking up at the Southern Cross hoping the milky way will inspire me to bring you a newsletter that is worth not pressing delete on. I am always interested in hearing about "gotchas" that you found while using Java, so please keep on sending me your ideas.

Since the GUI is 63% of the JDK, I was tempted to send another interesting bit about setting the focus to a component besides the first one on a modal dialog. This seemingly simple problem had me quite stumped, but luckily the Cape Town Java User Group gave me one solution and I found another solution myself. Next week I will write about that, so stay tuned. I know that "real" programmers don't write GUIs, but if you are a "real" Java programmer, other less real programmers might ask you how to do these fancy tricks, and it's good to then have an answer ;-) There is another newsletter in the pipeline that tells you how to serialize GUI components across the network.

## Hooking into the shutdown call

Last year, while I was trying out the jBoss EJB application server, I happened to notice that you could shutdown the server by pressing CTRL+C. Somehow, the server "knew" you had killed the program, and shutdown all the necessary services before exiting the Java VM.

My Java-Guru-In-The-Making colleague told me that it was possible to add a "shutdown hook" which would be called when the application was being shutdown. This shutdown event would happen when you call System.exit(), which in turn gets called when you press CTRL+C. As was hinted in last week's newsletter, once you use any GUI components, the VM starts up GUI threads that you cannot shutdown cleanly. This means that the only option you have of exiting your program is to call System.exit()!!!

However, System.exit() can be called from anywhere in your program and if it gets called, we want to have a central place where we can add methods that will be called when we want to exit. This central place was added, without much fanfare (actually no fanfare at all!), in JDK 1.3, together with many other cool features like the java.awt.Robot.

Here is an example of how you could add a shutdown hook into the current VM runtime. You do this by constructing

a new Thread and passing that to the runtime to start when it wants to shutdown.

```java
public class ShoutdownTrick {
  public static void main(String args[]) {
    Runtime.getRuntime().addShutdownHook(new Thread() {
      public void run() {
        System.out.println("You wanna quit, hey?");
        System.out.println("... fry eggs on your CPU.");
        while(true);
      }
    };
    System.out.println("Let's take a break...");
    try {
      Thread.sleep(5000);
    }
    catch(InterruptedException ex) {}
    System.out.println("That's it, I'm outta here");
    System.exit(0);
    System.out.println("This line will not show!");
  }
}
```

Try run this on your machine and you will notice that the only way to stop the program is via the task manager. This of course highlights one of the problems with writing code to call during shutdown. What happens when the shutdown code does not exit? I am told that in Delphi you can solve the Halting Problem, unfortunately Java is not that powerful, so the VM cannot know whether the shutdown code will stop or not. Ideally, you should in the shutdown code only do what is absolutely necessary to put the system in a state that is usable next time you want to run the program, i.e. close DB connections cleanly, close log files, etc.

A feature I did not know about until tonight is that you can bypass the shutdown hooks and actually "halt" the VM without further ado. So, the obvious next step in our Shutdown Hook example is to add another thread that will halt the VM after some timeout:

```java
public class ShutdownTrickWithHaltTimeout {
  public static void main(String args[]) {
    Runtime.getRuntime().addShutdownHook(new Thread() {
      public void run() {
        System.out.println("You wanna quit, hey?");
        System.out.println("... fry eggs on your CPU.");
        while(true);
      }
    });
    Runtime.getRuntime().addShutdownHook(new Thread() {
      public void run() {
        try {
          Thread.sleep(10000);
        } catch(InterruptedException ex) {}
        // halt will bail out without calling further shutdown hooks or
        // finalizers
        Runtime.getRuntime().halt(1);
      }
    });
    System.out.println("Let's take a break...");
    try {
      Thread.sleep(5000);
    }
```

```
        catch(InterruptedException ex) {}
        System.out.println("That's it, I'm outta here");
        System.exit(0);
        System.out.println("This line will not show!");
      }
    }
```

The problem with this code is that even though you know that after 10 seconds you will definitely halt the VM, you will have to wait for 10 seconds each time you try to exit the VM, because there is no way of knowing if the other threads are finished yet. This brings out another interesting feature, which is that all the shutdown hooks are threads and it seems that they are all started at the same time. The obvious solution is to make the "Halting" shutdown thread a Daemon thread, but, alas, at that point in the VM processing, it doesn't matter whether a thread is a Daemon thread or not, it will run until it is finished.

It seems that, even though our friends at Sun have noticed that such a shutdown hook feature is necessary, they have not fully thought through the possibilities of what could go wrong and how to solve it. Perhaps a System.exit(int status, long timeout) would be appropriate?

There are some really cool extras in JDK 1.3 which are not very widely publicised, or perhaps I'm reading the wrong magazines?

Shall we take a bet how long it takes before all the System.exit(0) code in the world is replaced with Runtime.getRuntime().halt(0)?

Until next week

Heinz

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📧 Implementing a SoftReference based HashMap

# Author: Dr. Heinz M. Kabutz

## Copyright 2001 Maxkab Solutions CC - South Africa
## Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 15th issue of "The Java(tm) Specialists' Newsletter", this week we are looking at an extremely useful addition to the Java language, namely soft and weak references. In a nutshell, the main difference between the two seems to be that the SoftReference has some notion of remembering when last it was accessed, which may be used by the GC (if it feels like it) to release little-used SoftReferences first.

Next week I will be doing some training in Germany, so I will not be able to send out a newsletter, as I have not decided on a list server yet. Hopefully the week after that you'll have your normal fix again. Please continue forwarding this newsletter to your local JUG, friends and family who are interested in Java. If you do send the newsletter to a closed user list, kindly tell me, I like having an indication of how many souls are reading my newsletter. Similarly, if you quote from my newsletter, be so kind as to attribute the source.

Many thanks to Sydney Redelinghuys for his input and corrections in this newsletter. (Sydney wants to spend a bit of time working in London, so if you are in the London area and would like the services of one of the most brilliant Java programmers I know, please send Sydney an email.

Once again I have seen the value in actually testing my code, as there was a serious error in my code which I only found through my "unit" test.

# Implementing a SoftReference based HashMap

Java is slow. Java is a memory hog.

But, if I have to write a network application, I would not hesitate for a second to use Java. Why? Because Java shines in threaded, network-based applications and because over the years, I have recognised the weaknesses of Java and have learnt (the hard way) what I should do, or not do, to avoid running into too

serious problems. Recognising the problems takes a while to learn, which is why most Java jobs require you to have at least 2 years of Java programming experience.

One of the most common places of memory wastage is in hash maps, so SUN have provided a WeakHashMap to minimize memory usage in caches implemented using maps. A WeakHashMap stores the keys using WeakReference objects, which means in layman's terms that as soon as the key is not referenced from somewhere else in your program, the entry may be removed and is available for garbage collection. (Have a look at the JavaDocs for the java.util.WeakHashMap and java.lang.ref.WeakReference for more information) Say you write a middle tier that provides an OO view of your database via an application server. What you really want is for the values to be automatically released, rather than the keys. If you put all the objects into a normal HashMap, you can easily run out of memory when you access many different objects from different clients. But if you store the objects in a WeakHashMap they are cleared as soon as your clients is not referencing them anymore. What you really want, however, is to only have the objects released when the VM is running low on memory, since that is when you get problems.

Enter SoftReferences. As far as I understand, a SoftReference will only be garbage collected when the VM is running low on memory and the object it is pointing to is not accessed from a normal (hard) reference. This is probably a better option to use for the HashMap values than a WeakReference, but the default JDK collections don't include a GC-friendly HashMap based on values and neither does it provide a SoftReference based HashMap.

Before I show you a SoftHashMap implementation, based on ideas by Sydney (what's up doc?) Redelinghuys, I need to explain some ideas which will make our SoftHashMap more optimal.

1. Each time we change the map (put, remove, clear) or ask for its size, we first want to go through the map and throw away entries for which the values have been garbage collected. It is quite easy to find out which soft references have been cleared. You can give the SoftReference a ReferenceQueue to which it is added when it is garbage collected.

2. We don't want our hash map to be bullied by the garbage collector, so we provide the option of the map itself keeping a hard link to the last couple of objects (typically 100).

3. The SoftHashMap will use a variant of the Decorator pattern to add this functionality to an internally kept java.util.HashMap. I'm busy working on a Design Patterns course based on the GOF book, let me know if you want further information.

Without further ado, here comes the SoftHashMap:

```java
//: SoftHashMap.java
import java.util.*;
import java.lang.ref.*;

public class SoftHashMap extends AbstractMap {
    /** The internal HashMap that will hold the SoftReference. */
    private final Map hash = new HashMap();
    /** The number of "hard" references to hold internally. */
    private final int HARD_SIZE;
    /** The FIFO list of hard references, order of last access. */
    private final LinkedList hardCache = new LinkedList();
    /** Reference queue for cleared SoftReference objects. */
    private final ReferenceQueue queue = new ReferenceQueue();

    public SoftHashMap() { this(100); }
    public SoftHashMap(int hardSize) { HARD_SIZE = hardSize; }

    public Object get(Object key) {
        Object result = null;
        // We get the SoftReference represented by that key
        SoftReference soft_ref = (SoftReference)hash.get(key);
```

```java
    if (soft_ref != null) {
      // From the SoftReference we get the value, which can be
      // null if it was not in the map, or it was removed in
      // the processQueue() method defined below
      result = soft_ref.get();
      if (result == null) {
        // If the value has been garbage collected, remove the
        // entry from the HashMap.
        hash.remove(key);
      } else {
        // We now add this object to the beginning of the hard
        // reference queue.  One reference can occur more than
        // once, because lookups of the FIFO queue are slow, so
        // we don't want to search through it each time to remove
        // duplicates.
        hardCache.addFirst(result);
        if (hardCache.size() > HARD_SIZE) {
          // Remove the last entry if list longer than HARD_SIZE
          hardCache.removeLast();
        }
      }
    }
    return result;
  }

  /** We define our own subclass of SoftReference which contains
   not only the value but also the key to make it easier to find
   the entry in the HashMap after it's been garbage collected. */
  private static class SoftValue extends SoftReference {
    private final Object key; // always make data member final
     /** Did you know that an outer class can access private data
      members and methods of an inner class?  I didn't know that!
      I thought it was only the inner class who could access the
      outer class's private information.  An outer class can also
      access private members of an inner class inside its inner
      class. */
    private SoftValue(Object k, Object key, ReferenceQueue q) {
      super(k, q);
      this.key = key;
    }
  }

  /** Here we go through the ReferenceQueue and remove garbage
   collected SoftValue objects from the HashMap by looking them
   up using the SoftValue.key data member. */
  private void processQueue() {
    SoftValue sv;
    while ((sv = (SoftValue)queue.poll()) != null) {
      hash.remove(sv.key); // we can access private data!
    }
  }
  /** Here we put the key, value pair into the HashMap using
```

```java
    a SoftValue object. */
  public Object put(Object key, Object value) {
    processQueue(); // throw out garbage collected values first
    return hash.put(key, new SoftValue(value, key, queue));
  }
  public Object remove(Object key) {
    processQueue(); // throw out garbage collected values first
    return hash.remove(key);
  }
  public void clear() {
    hardCache.clear();
    processQueue(); // throw out garbage collected values
    hash.clear();
  }
  public int size() {
    processQueue(); // throw out garbage collected values first
    return hash.size();
  }
  public Set entrySet() {
    // no, no, you may NOT do that!!! GRRR
    throw new UnsupportedOperationException();
  }
}
```

And here comes some test code that demonstrates to a certain degree that I'm not talking complete bullsh*t. Soft and weak references are quite difficult to experiment with as there is a lot of freedom left to the writers of the JVM of how they must implement them. I wish the implementation would hold back longer before removing these references, that the JVM would wait until it is really running low on memory before clearing them, but unfortunately I am not the one who wrote the JVM. I have tried to question the authors of the java.lang.ref package to find out what the strategy is for references in future versions, but have not had any response yet.

```java
//: SoftHashMapTest.java
import java.lang.ref.*;
import java.util.*;

public class SoftHashMapTest {
  private static void print(Map map) {
    System.out.println("One=" + map.get("One"));
    System.out.println("Two=" + map.get("Two"));
    System.out.println("Three=" + map.get("Three"));
    System.out.println("Four=" + map.get("Four"));
    System.out.println("Five=" + map.get("Five"));
  }
  private static void testMap(Map map) {
    System.out.println("Testing " + map.getClass());
    map.put("One", new Integer(1));
    map.put("Two", new Integer(2));
    map.put("Three", new Integer(3));
    map.put("Four", new Integer(4));
    map.put("Five", new Integer(5));
    print(map);
    byte[] block = new byte[10*1024*1024]; // 10 MB
    print(map);
```

```
      }
    public static void main(String[] args) {
        testMap(new HashMap());
        testMap(new SoftHashMap(2));
      }
  }
```

Until next week, and please remember to forward this newsletter and send me your comments.

Heinz

---

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

☐ IT Professional  ☐ General Computer  ☐ Design and Graphics  ☐ Linux  ☐ Freeware/Shareware

☐ Games  Intranet  Computer Hardware  Web Design  ☐ Computer Software

☐ UNIX  ☐ Web Software  ☐ Windows Software  ☐ Personal Finance  ☐ Programming Software

☐ Servers  ☐ C/C++  ☐ Powerbuilder  ☐ Perl  ☐ XML

☐ Java  ☐ SQL  ☐ Design/Graphics  ☐ Utilities  ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Implementation code inside interfaces

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 6th issue of "The Java(tm) Specialists' Newsletter", a weekly newsletter appearing each Thursday, which explores some of the interesting parts of the Java(tm) language. My company, Maximum Solutions, has been very active with Java programming, consulting, training and mentoring. I particularly want to thank my client DataVoice (Pty) Ltd in Stellenbosch, South Africa, for giving me the opportunity to spend hours programming in Java on one of the first real Java projects in South Africa, perhaps even the biggest, with more lines of pure Java code than the JDK 1.3, and actually paying me to have fun.

Last night I worked on one of my most successful programs. It was a small 56 line program that I wrote in QuickBasic in April 1988, which my dad, who has a drinking straw manufacturing company, has used, with only minor modifications, to print thousands and thousands of labels for the boxes in which he packs his drinking straws. Perhaps one day you will see the logo "Maximum Solutions, the QuickBasic specialists!"

It's a terrible program, impossible to decipher, even contains a GOTO statement (!) but it does the job extremely reliably. I had to change his address last night (yes, it's hard coded!) and was thinking about what effort would be involved in writing the same program in Java. I would probably store all the customer details in an Access database, have autocompleting combo boxes, etc., but it would cost a fortune in time and energy to produce, far more than I could save my making it "nicer".

It is very important that as Java enthusiasts we keep an objective view of what applications lend themselves to Java and which do not. Otherwise we might find that we end up with something that is too expensive or too slow for its intended purpose. Overselling Java might benefit us in the short term, as companies scurry to hire us, but will damage us if Java gets a bad name. A Java Architect can command a salary of US$ 170'000 in USA at the moment, let's live up to the expectation and be responsible in our claims.

Now to the trick of the week, which is something you should never need to do. Please avoid doing this under all circumstances, because there are much better places to put method code than in interfaces, but I want to show you what is possible with inner classes. The call-back mechanism shown below is actually quite useful at times, especially if you want to make asynchronous database updates, but that is a topic for another newsletter.

I want to thank Niko Brummer from DataVoice for this idea, although he vehemently denies having ever really resorted to writing implementation code in an interface. Niko is a very deep person who likes to think of alternative ways of doing things, so thanks to Niko for this crazy idea :)

```
/**
  This interface contains an interfaces representing a Method and an
```

```java
    interface showing the Result of the method call using the callback
    pattern.  It also contains a data member (automatically public static
    final) which is an anonymous inner class implementing our Method
    interface.
*/
public interface CodeInsideInterface {
  public interface Method {
    public void run(Result callback);
  }
  public interface Result {
    public void result(Object answer);
    public void exception(Exception problem);
  }
  Method calculateIQ = new Method() {
    // I always write my data members as final if possible, this catches a
    // lot of bugs at compile time
    private final java.io.BufferedReader stdin = new java.io.BufferedReader(
      new java.io.InputStreamReader(System.in));
    public void run(Result callback) {
      int iq = 100;
      try {
        System.out.print("Do you know Java (y/n)? ");
        if ("y".equals(stdin.readLine())) iq += 20;
        System.out.print("Do you know QuickBasic (y/n)? ");
        if ("y".equals(stdin.readLine())) iq += 20;
        System.out.print("Do you use the Basic GOTO statement (y/n)? ");
        if ("y".equals(stdin.readLine())) iq -= 30;
        System.out.print("Do you frequently use Java reflection (y/n)? ");
        if ("y".equals(stdin.readLine())) iq -= 50;
        callback.result(new Integer(iq));
      } catch(java.io.IOException ex) {
        callback.exception(ex);
      }
    }
  };
}

/**
  This test class demonstrates how to call the method on the interface.
 */
public class CodeInsideInterfaceTest implements CodeInsideInterface {
  public static void main(String[] args) {
    CodeInsideInterfaceTest test = new CodeInsideInterfaceTest();
    test.calculateIQ.run(new CodeInsideInterface.Result() {
      public void result(Object answer) {
        System.out.println("Your IQ is " + answer);
      }
      public void exception(Exception ex) {
        ex.printStackTrace();
      }
    });
  }
}
```

Well, there you go. Try and understand what is happening in the code, because it will teach you something about inner classes and how the callback mechanism works.

In the first newsletter I mentioned something about having multiple event queues in AWT / Swing and how I didn't know what the purpose is. Last Sunday I was pondering how I could possibly catch ALL events that occur in AWT / Swing and after playing around for a few hours managed to figure out how the event queues work. Next week I will show you the answer to my question in the first newsletter, i.e. why does AWT allow more than one event queue, how does it work, and what practical application is there... Send me email if you want to get back copies of newsletters. We are working on a web archive.

Regards

Heinz

---

## java.awt.EventQueue

**Author: Dr. Heinz M. Kabutz**

**Copyright 2001 Maxkab Solutions CC - South Africa
Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 8th issue of "The Java(tm) Specialists' Newsletter"! Ok, it is only the 7th issue, just testing if you're awake. This week I want to tell you the answer to my question in the first newsletter, namely, "under what circumstances is it possible to have more than one AWT event queue?" I didn't particularly go looking for an answer, but found one "by accident" while looking for a solution to another problem.

But, before I tell you how to use multiple event queues, let me bore you with a tale of why I went looking for this. A bit over a year ago, SUN released JDK 1.3 beta, and one of my colleagues, Java GUI fundi Herman Lintvelt, told me that it contained a new class called java.awt.Robot. This "Robot" could be instructed to issue native mouse events, keyboard events or take screen shots. The purpose of this Robot was to make it possible to write tests that emulated real users by issuing native actions using a platform-independent Java interface. It is difficult to find testers willing to do repetitive, boring testing, such as clicking ALL the buttons on an application each time a build is made. The Robot could be instructed to jump to a specific place on the screen and press mouse buttons and take screen shots if necessary.

Maximum Solutions quickly got stuck in and developed a testing framework around this "Robot". It is driven by scripts in which you can specify the text on the component that should be "clicked". The framework found the exact location of the component on the screen and issued a native windows click using the java.awt.Robot. Components could thus be located precisely inspite of layout managers. The code contained some magic tricks, as you might imagine, which I will not reveal (for now - maybe later, once I've sold enough copies of the framework). Luckily I had fellow Java Contractor Guild member Sydney Redelinghuys on my payroll and together we eeked information from the VM using the standard interfaces which a casual observer would not see.

The framework could be instructed to take screen shots at certain check- points of the script, or to take screen shots if it detected an error. It could find components located on different tab sheets, components not visible on the screen due to scrolling, items in combo boxes, etc.

The main "acceptance" problem we have experienced with the UIRobot is that it is quite difficult to write a comprehensive script, especially if you are not a programmer by profession. Most testers I have met struggle to write such scripts, and would prefer an automatic procedure of recording the script.

## Multiple event queues

In order to make my UIRobot more acceptable to the marketplace, I needed a way in which I could unobtrusively hook

myself into the AWT event system. I wanted a hotkey that users of the UIRobot could employ from any window to open the UIRobot dialog. The user could then "record" a script, "play" a script or "edit" a script, much like the macro recorder in MS Office (TM), using text on the components to locate them again. The only thing I expect the client code to do is to call

```
Class.forName("com.maxoft.ui.robot.UIRobot");
```

in order to give the UIRobot class a chance to register itself.

Once that happens, I want the hotkey to be available from any frame, dialog, component, focused or not, etc. It should be a global hotkey that you can press to activate the UIRobot dialog. I had a look at the EventQueue for my first newsletter and noticed that it followed, roughly, the Chain of Responsibility design pattern. You can register a new event queue and make it responsible for handling any new events that arrive.

I saw this pattern quite quickly, but it took me 3 hours (!) to finally get it working. Had I been more careful, it should have taken not more than 5 minutes, but sometimes I am a bit slow on the uptake. One of the disadvantages of having written too much generic Java code is that I don't recognise "private" methods as an obstacle, because I can just invoke them anyway. Designers of frameworks are not necessarily very skilled at guessing how I want to extend their framework so they often make a method private instead of protected. I even went so far as to decompile SUN's implementation of java.awt.Toolkit, namely sun.awt.SunToolkit, to try and find some way of hooking into the event queue. In the end the correct and most simple way of doing this was to write a subclass of EventQueue, called MyEventQueue, and to register it as the now reigning king of the Democratic Republic of Events with the command:

```
Toolkit.getDefaultToolkit().getSystemEventQueue().push(
    new MyEventQueue());
```

The reason it took me 3 hours to figure these couple of lines out was because I overrode the postEvent() method, instead of the dispatchEvent() method, duh! An example of MyEventQueue could look like this:

```java
//: MyEventQueue.java
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class MyEventQueue extends EventQueue {
    protected void dispatchEvent(AWTEvent event) {
        // the only functionality I add is that I print out all the events
        System.out.println(event);
        super.dispatchEvent(event);
    }
}
```

So, what type of functionality can you achieve with this code? You can write a global hotkey manager, you can write a recorder that generates scripts for the UIRobot or you can disable all user actions while the GUI is busy with something else. Those of you who've tried to disable GUI input have probably used the GlassPane of the JFrame which can catch all mouse events, but not keyboard shortcuts. Thanks to David Geary for that idea in his classic book on Swing!

I mentioned to one of our system engineers the possibility of using this event queue mechanism as a global hotkey manager. He got very excited and called Herman away from the company month-end barbacue to come talk to me. We had been struggling to get application-wide global hotkeys working for 3.5 years in our application and one mayor customer site was holding back on a purchase because of that problem. Herman and I sat down and we came up with the GlobalHotkeyManager, where you can register any input / action combination and it matches the two for you on a global scale. Note that you need JDK 1.3 to use ActionMap and InputMap. The other parts work in JDK 1.2.

```java
//: GlobalHotkeyManager.java
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class GlobalHotkeyManager extends EventQueue {
    private static final boolean DEBUG = true; // BUG? what's that? ;-))
    private static final GlobalHotkeyManager instance =
        new GlobalHotkeyManager();
    private final InputMap keyStrokes = new InputMap();
    private final ActionMap actions = new ActionMap();
```

```
    static {
      // here we register ourselves as a new link in the chain of
      // responsibility
      Toolkit.getDefaultToolkit().getSystemEventQueue().push(instance);
    }
    private GlobalHotkeyManager() {} // One is enough - singleton
    public static GlobalHotkeyManager getInstance() {
      return instance;
    }
    public InputMap getInputMap() {
      return keyStrokes;
    }
    public ActionMap getActionMap() {
      return actions;
    }
    protected void dispatchEvent(AWTEvent event) {
      if (event instanceof KeyEvent) {
        // KeyStroke.getKeyStrokeForEvent converts an ordinary KeyEvent
        // to a keystroke, as stored in the InputMap.  Keep in mind that
        // Numpad keystrokes are different to ordinary keys, i.e. if you
        // are listening to
        KeyStroke ks = KeyStroke.getKeyStrokeForEvent((KeyEvent)event);
        if (DEBUG) System.out.println("KeyStroke=" + ks);
        String actionKey = (String)keyStrokes.get(ks);
        if (actionKey != null) {
          if (DEBUG) System.out.println("ActionKey=" + actionKey);
          Action action = actions.get(actionKey);
          if (action != null && action.isEnabled()) {
            // I'm not sure about the parameters
            action.actionPerformed(
              new ActionEvent(event.getSource(), event.getID(),
                actionKey, ((KeyEvent)event).getModifiers()));
            return; // consume event
          }
        }
      }
      super.dispatchEvent(event); // let the next in chain handle event
    }
  }
```

Together with the GlobalHotkeyManager.java we have a test program:

```
//: GlobalHotkeyManagerTest.java
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class GlobalHotkeyManagerTest extends JFrame {
  private final static String UIROBOT_KEY = "UIRobot";
  private final KeyStroke uirobotHotkey = KeyStroke.getKeyStroke(
    KeyEvent.VK_R, KeyEvent.CTRL_MASK + KeyEvent.ALT_MASK, false);
  private final Action uirobot = new AbstractAction() {
    public void actionPerformed(ActionEvent e) {
      setEnabled(false); // stop any other events from interfering
      JOptionPane.showMessageDialog(GlobalHotkeyManagerTest.this,
```

```java
                          "UIRobot Hotkey was pressed");
                setEnabled(true);
            }
        };
    public GlobalHotkeyManagerTest() {
        super("Global Hotkey Manager Test");
        setSize(500,400);
        getContentPane().setLayout(new FlowLayout());
        getContentPane().add(new JButton("Button 1"));
        getContentPane().add(new JTextField(20));
        getContentPane().add(new JButton("Button 2"));
        GlobalHotkeyManager hotkeyManager = GlobalHotkeyManager.getInstance();
        hotkeyManager.getInputMap().put(uirobotHotkey, UIROBOT_KEY);
        hotkeyManager.getActionMap().put(UIROBOT_KEY, uirobot);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); // JDK 1.3
        setVisible(true);
    }
    public static void main(String[] args) {
        new GlobalHotkeyManagerTest();
    }
}
```

We basically match a KeyStroke (CTRL+ALT+R) with an Action. Since the action can be invoked from anywhere, we must remember to switch it off while we are handling it, otherwise it could be invoked again by mistake. Try out what happens when you don't disable the action and press the hotkey twice.

This is one of the most interesting things I've discovered in Swing and is so extremely useful that I do not understand why it is not more widely publicised. Please let me know if you've done something similar in your coding. It seems that SUN are quite good at adding new features or improving code without bothering to tell anyone, or at least not me! ;-) At more than 500'000 lines of code in the JDK 1.3, it becomes tiresome to read through it all each time a new release comes out.

By now, you have hopefully seen the value of understanding OO Design Patterns if you want to become good at Java. I have found that Java lends itself to good OO design, certainly more than C++.

I want to thank all of you who took the time to read and respond to my newsletters, your feedback is what really makes this worthwhile.

Regards

Heinz

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📖 Logging   Part I

**Author: Dr. Heinz M. Kabutz**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the third "The Java(tm) Specialists' Newsletter", a low-volume newsletter that is aimed at Java programmers who are facing the real world every day. I hope you are enjoying reading this information and that it will help you to be more productive in your work. I want to thank Java-Guru-In-The-Making Dr. Christoph Jung for volunteering to write a piece on dynamic proxies in an upcoming newsletter. Please let me know if there are topics that you would like discussed or if you would like to share some of your own experiences through this newsletter. This newsletter is distributed free of charge to anyone interested, so please forward it to friends and colleagues who might find the information in this newsletter useful.

I drew some fire from respectable quarters that some of the code I presented in the last newsletter was a little bit on the obscure side. The Thread example was meant to illustrate the concept of making an anonymous inner class and immediately putting it to use, as opposed to just using it as a parameter. The most probable place you would find code as shown in the last newsletter would be in the unit tests of my code. If you have a method that requires as a parameter a Vector, you can use the example from the previous newsletter to succinctly pass it test data. Even though the examples might seem a bit obscure at first glance, I believe that once you've seen the "Anonymous Inner Class Init Block" example, you will begin to recognise it easily enough so it will lose most of its obscurity.

If you are not writing unit test for your code, why not? There are two good books that contain information on unit tests, Refactoring by Martin Fowler and eXtreme Programming by Kent Beck. Unit tests have saved my butt a number of times, because I am not a very "good" programmer, in that I make mistakes all the time and end up messing up old working code in the process - does that sound familiar? (So, if you let me work on some of your code and you don't have unit tests, BEWARE!) When people ask me what my job is I tell them I am a professional bug writer, because that's what I'm best at. Unit tests only work if the whole development team is using the concept, much like CASE tools such as Rational Rose. I fought a losing battle 2 years ago when most of the

team, on a project I was on, stopped using Rational Rose and I spent most of my time updating other developers' diagrams.

Talking of unit tests, it is possible to use the Robot (java.awt.Robot) to write unit tests for GUI's. Click on TellMeAboutTheUIRobot for more information.

# Logging Part 1 of ... (haven't decided yet ;-)

While we are on the topic of things going wrong, we should always expect things to go wrong in the field and it is very important to be able to remotely figure out what went wrong and to prevent it from going wrong in future. An example of a remote fix was the Mars bug that had to be debugged and repaired remotely (not by me ;-) - the Risks digest had some interesting articles on that. With the internet being so widely accessible, we have a great challenge because our programs might fail somewhere in a remote part of this planet that might not be very pleasant for a support team to visit, for example amongst a man-eating pygmy tribe in Java (non-tm). Preferably we would ask them to email us a log of what went wrong so that we can look at the log and determine what the problem was.

When you start with a new project it is very easy to give logging too little thought, with unpleasant consequences. It is also possible to give it too much thought and design such a complex logging system that the rest of the programming team cannot wait and starts writing their own logging functions (System.out.println springs to mind). In that case you are back to the first problem of too little thought, because each developer develops his method as he goes along. This newsletter is not going to tell you which way is best, you have to decide for yourself, but it will show you some tricks and gotcha's that I've seen several times.

In designing a logging system, there are several questions we have to face:

1. How do we show when the event happened?
2. What if the log size gets too big?
3. How do you switch off logging?
4. Are we willing to make the client code messy to improve performance?

The combination of these factors can sway your logging system in various ways, which is why I will not say which is best. I will now present the first of a series of case studies of logging systems that one might want to write:

## Case Study 1:

Frank the Unix hacker has been writing a 3-tier application server that uses several third-party libraries which output logging information such as "Connected to ISAP" or "Disconnected from AISN" or "State changed to TLA". Since Frank is a hacker, he knows about JAD.EXE and is forever grateful to Pavel for making all CLASS files open-source, but he is a bit worried about rewriting the third-party code to output to his logging system because of something called "copyright law", whatever that is. The problem is that he needs to know exactly in which order things happen in respect to his own application server, so perferably the log should show all output to the console in the same order in which it appeared. Since Frank is a UNIX hacker, he remembers a little utility called "tee" that would represent a t-piece in a pipe of output so that you could send piping to two destinations.

For example:

```
echo hello world | tee output > output2
```

would send the string "hello world" to two output files, "output" and "output2".

Frank decides to write an IO stream decorator that behaves as a TeeOutputStream. (Disclaimer: I wrote my first TeePrintStream in the middle of 1997 - I have since seen the same thing published in a magazine.) If you don't understand decorators, have a look at the IO chapter of the excellent book "Thinking in Java" by Bruce Eckel or visit a good Java course.

The TeeOutputStream would look something like this:

```java
// TeeOutputStream
import java.io.*;
public class TeeOutputStream extends FilterOutputStream {
  private final OutputStream out2;
  public TeeOutputStream(OutputStream out1, OutputStream out2) {
    super(out1);
    this.out2 = out2;
  }
  public void write(int b) throws IOException {
    super.write(b);
    out2.write(b);
  }
  public void flush() throws IOException {
    super.flush();
    out2.flush();
  }
  public void close() throws IOException {
    super.close();
    out2.close();
  }
}
```

The TeeOutputStream would now be able to take two output streams and send the same information to both of them as shown in the main method.

```java
// ... the main class that starts up the application server
  public static void main(String[] args) {
    PrintStream out1 = System.out;
    OutputStream out2 = new BufferedOutputStream(
        new FileOutputStream("LOG"));
    TeeOutputStream newOut = new TeeOutputStream(out1, out2);
    System.setOut(new PrintStream(newOut, true));
    out1 = System.err;
    newOut = new TeeOutputStream(out1, out2);
    System.setErr(new PrintStream(newOut, true));
    // ... some more code...
  }
```

My personal preferance would be to re-write those lines as follows, but you should do what is most comfortable to you:

```java
// ... the main class that starts up the application server
  public static void main(String[] args) {
    OutputStream log = new BufferedOutputStream(
        new FileOutputStream("LOG"));
    System.setOut(
      new PrintStream(new TeeOutputStream(System.out,log),true));
    System.setErr(
      new PrintStream(new TeeOutputStream(System.err,log),true));
    // ... some more code...
  }
```

The astute among you should now be jumping up and down pointing at the glaringly obvious mistake in the above code. There is no provision for the log file growing and filling up the hard disk. Frank (correctly) assumes that his program is never going to run for a long enough period that the size of the log file would be bigger than 10 KB, but once all his bugs are removed, something should be done to prevent the big log file. The biggest log file that an advanced support man sent me was 1.1 GB in size when I used Frank's TeeOutputStream system.

Anyway, the beauty with Frank's approach is that whenever he wants to print to the log he just does a "System.out.println()" or exception.printStackTrace(). Anything that goes to the console also goes to his log in the same order in which it appeared to the user. He can also switch off logging to a file by not setting the System.out PrintStream to print to the tee, or he can set his program to only print to the log.

If we look back at the questions we face with the logging system, we see that we don't have a

default way of showing WHEN the event occured, although that could be added by changing the write() method of the FileOutputStream to print the date/time after each newline. The size of the log file can be monitored by implementing a different FileOutputStream. The logging to file can be switched off easily. The client code is very simple - System.out.println("Changed to state TLA");

That's the end of Frank's logging system. As you have seen, there is a lot of scope for changing it, depending on your requirements, but there is no right answer that fits all scenarios. Next week I will present another scenario with different advantages...

Please remember to forward this free newsletter to whoever you think may be interested.

With regards

Heinz

---

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📋 Logging   Part II

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the fourth issue of "The Java™ Specialists' Newsletter" and the first issue this year, or as some ignoramuses believe, this millenium. Next week Dr. Christoph Jung will tell us how the new Java dynamic proxies work, don't miss it. It's an excellent piece of writing, so if you are not subscribed to this newsletter yet, make sure you don't miss it and subscribe. Please forward this newsletter to friends, family and fauna that you think might find it useful and interesting.

[If anything, read the second-last paragraph of this newsletter, you might end up winning a free lunch :]

## Logging Part 2 of ... (still cannot decide ;-)

Last year I started talking about things that go wrong in our code. I probably sent the newsletter out too close to Christmas (merry to you, btw), because I had hardly any response. Alternatively you've added me to your junk-mail-auto-delete-mail-pest filter (sniff sniff) in which case you won't read this anyway. I hope your inbox is looking emptier and that you will find time to read this newsletter. Or maybe your code always works so there's nothing to Log ;-) [naah]

Over the past years, two very knowledgeable Java programmers asked me independently how one can find out within a method where the method was invoked from. This is particularly useful when you want to print out logging information, such as where in the program the log was printed from. My answer to them was both times a typical "consulting" answer in the form of "what an interesting problem, how would you solve it?" and then I went on to ramble about using JNI, parsing the stack trace, using the Visitor pattern or writing the whole thing with Jini. i.e. I didn't have a clue!

Graciously, one of these two men, Gary Plante, sent me some code the other day which he discovered could do just that. It works by parsing a stack trace, and I took the liberty of fiddling here and there with his code, so it probably looks nothing like what you sent me, Gary :) According to my tests, this whole idea is probably a bit on the slow side. Also, to make my code easier to understand,

and to avoid relying on side-effects of the PrintWriter class, I removed most of Gary's performance optimisations with the result that my code may be a little bit more robust but it is 2.5 times slower than Gary's. I have included Gary's code "as is" to show you how it could also be done. We start counting lines at different indexes, beware.

```java
/**
  The stack trace starts with the following lines:
  java.lang.Throwable
      at StackTrace2.getStackTraceAtLine(StackTrace2.java:19)

  It does not make sense to get these first 2 lines so we will
  ignore them.  The number we pass into the method is the depth
  of method calls, where 0 is the current line of code, 1 is the
  line of code that called this code, 2 is the line of code that
  called the line of code that called this code, etc.

  0:    at StackTrace2Test.g(StackTrace2Test.java:10)
  1:    at StackTrace2Test.f(StackTrace2Test.java:3)
  2:    at StackTrace2Test.main(StackTrace2Test.java:17)
 */
import java.io.*;
public class StackTrace2 {
  private static final Throwable tracer = new Throwable();
  private static final StringWriter sw =
    new StringWriter(1024);
  private static final PrintWriter out =
    new PrintWriter(sw, false);

  private StackTrace2() {} // Avoid direct creation

  public static String getCallStack(int depth) {
    synchronized (tracer) {
      if (depth < 0) throw new IllegalArgumentException();
      // skip the first 2 lines
      int lineOfInterest = depth + 3;
      // set the buffer back to zero
      sw.getBuffer().setLength(0);
      tracer.fillInStackTrace();
      tracer.printStackTrace(out);
      out.flush();
      LineNumberReader in = new LineNumberReader(
        new StringReader(sw.toString()));
      try {
        String result;
        while((result = in.readLine()) != null) {
          if (in.getLineNumber() == lineOfInterest)
            return beautify(result);
        }
      }
      catch(IOException ex) {} // we'll just return null
      return null;
    }
  }
```

```java
    private static String beautify(String raw) {
      raw = raw.trim(); // we don't want any whitespace
      if (raw.startsWith("at ")) // we also cut off the "at "
        return raw.substring(3);
      return raw;
    }
}
```

You could use this information in a VERY simple Log class as follows:

```java
public class Log {
    public static void it(String msg) {
        //caller of it()
        String source = StackTrace2.getCallStack(1);
        System.out.println(source + " : " + msg);
    }
}
```

An example of how this could be used is in

```java
public class StackTrace2Test {
    public void f() {
        g();
    }
    public void g() {
        Log.it("where am I now?");
        System.out.println(StackTrace2.getCallStack(0));
        System.out.println(StackTrace2.getCallStack(1));
        System.out.println(StackTrace2.getCallStack(2));
        System.out.println(StackTrace2.getCallStack(3));
        System.out.println(StackTrace2.getCallStack(-1));
    }
    public static void main(String[] args) {
        new StackTrace2Test().f();
    }
}
```

The output would look as follows:

```
StackTrace2Test.g(StackTrace2Test.java:6) : where am I now?
StackTrace2Test.g(StackTrace2Test.java:7)
StackTrace2Test.f(StackTrace2Test.java:3)
StackTrace2Test.main(StackTrace2Test.java:15)
null
Exception in thread "main" java.lang.IllegalArgumentException
        at StackTrace2.getCallStack(StackTrace2.java:22)
        at StackTrace2Test.g(StackTrace2Test.java:11)
        at StackTrace2Test.f(StackTrace2Test.java:3)
        at StackTrace2Test.main(StackTrace2Test.java:15)
```

Attached you can find the other StackTrace.java and StackTraceTest.java files which are quite a bit more difficult to understand, but at least they do not create a new LineNumberReader each time you ask it for your line number.

Attached you can find the other StackTrace.java and StackTraceTest.java files which are quite a bit more difficult to understand, but as mentioned above, they are 2.5 times faster. If performance of your logging system is a problem you can easily have a boolean in your Log class that turns this level of detailed processing off.

The best solution is probably to do this type of low-level access in JNI. I will invite the first person who sends me a correct solution that uses JNI and is really fast, to join me for a steak over lunch at the Cattle Baron (non-CapeTown residents will have to fly to Cape Town at their own expense!)

Right, I hope you found this information interesting, I always appreciate your feedback, both positive and negative.

Heinz

---

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

| | | | | |
|---|---|---|---|---|
| ☐ IT Professional | ☐ General Computer | ☐ Design and Graphics | ☐ Linux | ☐ Freeware/Shareware |
| ☐ Games | Intranet | Computer Hardware | Web Design | ☐ Computer Software |
| ☐ UNIX | ☐ Web Software | ☐ Windows Software | ☐ Personal Finance | ☐ Programming Software |
| ☐ Servers | ☐ C/C++ | ☐ Powerbuilder | ☐ Perl | ☐ XML |
| ☐ Java | ☐ SQL | ☐ Design/Graphics | ☐ Utilities | ☐ Handhelds |

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📰  Multicasting in Java

**Author: Paul van Spronsen (Blue Label Software, email: pvspr@yahoo.com)**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

This week we again have a guest author, Paul van Spronsen who owns Blue Label Software in South Africa, a small software development company. I have written a **lot** of code with Paul and he is the best Java programmer that I've had the pleasure of working with. So, if your project needs some extra horse power (ok, brain power ;-), send him an email and I'm sure his company will be able to help you.

A special word of thanks also to Robin Beetge for hacking together a dirty Perl script to generate XML tags for the syntax highlighting. Your little script is saving me a lot of time :)

---

# Multicasting in Java

## 1. Introduction

This article deals primarily with the subject of multicast communication in Java. I have, however, included some background information to refresh the memory of those who have forgotten how much they know about data communications. If the concepts "datagram", "IP fragment", "reliable protocol" or "multicast" are not clear to you, try referring to the appendices. If the appendices appear shrouded in mystery, go back to your data comms lecturer and demand a refund.

## 2. Sending multicast datagrams

In order to send any kind of datagram in Java, be it unicast, broadcast or multicast, one needs a java.net.DatagramSocket:

```
DatagramSocket socket = new DatagramSocket();
```

One can optionally supply a local port to the DatagramSocket constructor to which the socket must bind.

This is only necessary if one needs other parties to be able to reach us at a specific port. A third constructor takes the local port AND the local IP address to which to bind. This is used (rarely) with multi-homed hosts where it is important on which network adapter the traffic is received. Neither of these is necessary for this example.

This sample code creates the socket and a datagram to send and then simply sends the same datagram every second:

```java
DatagramSocket socket = new DatagramSocket();

byte[] b = new byte[DGRAM_LENGTH];
DatagramPacket dgram;

dgram = new DatagramPacket(b, b.length,
    InetAddress.getByName(MCAST_ADDR), DEST_PORT);

System.err.println("Sending " + b.length + " bytes to " +
    dgram.getAddress() + ':' + dgram.getPort());
while(true) {
    System.err.print(".");
    socket.send(dgram);
    Thread.sleep(1000);
}
```

Valid values for the constants are:

- DGRAM_LENGTH: anything from 0 to 65507 (see section 5), eg 32
- MCAST_ADDR: any class D address (see appendix D), eg 235.1.1.1
- DEST_PORT: an unsigned 16-bit integer, eg. 7777

It is important to note the following points:

1. DatagramPacket does not make a copy of the byte-array given to it, so any change to the byte-array before the socket.send() will reflect in the data actually sent;
2. One can send the same DatagramPacket to several different destinations by changing the address and or port using the setAddress() and setPort() methods;
3. One can send different data to the same destination by changing the byte array referred to using setData() and setLength() or by changing the contents of the byte array the DatagramPacket is referring to;
4. One can send a subset of the data in the byte array by manipulating offset and length through the setOffset() and setLength() methods.

## 3. Receiving multicast datagrams

One can use a normal DatagramSocket to send and receive unicast and broadcast datagrams and to send multicast datagrams as seen in the section 2. In order to receive multicast datagrams, however, one needs a MulticastSocket. The reason for this is simple, additional work needs to be done to control and receive multicast traffic by all the protocol layers below UDP.

The example given below, opens a multicast socket, binds it to a specific port and joins a specific multicast group:

```java
byte[] b = new byte[BUFFER_LENGTH];
DatagramPacket dgram = new DatagramPacket(b, b.length);
MulticastSocket socket =
    new MulticastSocket(DEST_PORT); // must bind receive side
socket.joinGroup(InetAddress.getByName(MCAST_ADDR));
```

```
  while(true) {
    socket.receive(dgram); // blocks until a datagram is received
    System.err.println("Received " + dgram.getLength() +
      " bytes from " + dgram.getAddress());
    dgram.setLength(b.length); // must reset length field!
  }
```

Values for DEST_PORT and MCAST_ADDR must match those in the sending code for the listener to receive the datagrams sent there. BUFFER_LENGTH should be at least as long as the data we intend to receive. If BUFFER_LENGTH is shorter, the data will be **truncated silently** and dgram.getLength() will return b.length.

The MulticastSocket.joinGroup() method causes the lower protocol layers to be informed that we are interested in multicast traffic to a particular group address. One may execute joinGroup() many times to subscribe to different groups. If multiple MulticastSockets bind to the same port and join the same multicast group, they will all receive copies of multicast traffic sent to that group/port.

As with the sending side, one can re-use ones DatagramPacket and byte-array instances. The receive() method sets length to the amount of data received, so remember to reset the length field in the DatagramPacket before subsequent receives, otherwise you will be silently truncating all your incoming data to the length of the shortest datagram previously received.

One can set a timeout on the receive() operation using socket.setSoTimeout(timeoutInMilliseconds). If the timeout is reached before a datagram is received, the receive() throws a java.io.InterruptedIOException. The socket is still valid and usable for sending and receiving if this happens.

## 4. Multicasting and serialization

We have seen in the previous sections that we can multicast anything we can fit into a byte array. Conveniently for us, one of those things is a serialized object.

Object serialization is based on the assumption of a stream (ObjectOutput**Stream**, ObjectInput**Stream**), so we have to do a little massaging to squeeze this into our datagram paradigm. ObjectOutputStream writes a stream header (containing a magic number and version number) to the stream on construction and ObjectInputStream reads and checks this on construction (ever wondered why ObjectInputStream's constructor blocks until the ObjectOutputStream has been constructed on the sending side?). This is the reason one always attaches the ObjectOutputStream to the outgoing side of a socket before attaching the ObjectInputStream to the incoming side.

In order to multicast objects, we need to arrange that the stream header information is in each datagram. The simplest way to ensure this is to create a new ObjectOutputStream for each datagram we send and a new ObjectInputStream for each one we receive. We could probably avoid these instantiations by extending the two classes in question, but I'm not going into that here.

On the sending side, we can do something like this:

```
  ByteArrayOutputStream b_out = new ByteArrayOutputStream();
  ObjectOutputStream o_out = new ObjectOutputStream(b_out);

  o_out.writeObject(new Message());

  byte[] b = b_out.toByteArray();

  DatagramPacket dgram = new DatagramPacket(b, b.length,
    InetAddress.getByName(MCAST_ADDR), DEST_PORT); // multicast
  socket.send(dgram);
```

In addition, on the receiving side we can do something like this:

```
  byte[] b = new byte[65535];
  ByteArrayInputStream b_in = new ByteArrayInputStream(b);
  DatagramPacket dgram = new DatagramPacket(b, b.length);
```

```
socket.receive(dgram); // blocks
ObjectInputStream o_in = new ObjectInputStream(b_in);
Object o = o_in.readObject();
dgram.setLength(b.length); // must reset length field!
b_in.reset(); // reset so next read is from start of byte[] again
```

Note that one can re-use the ByteArray*Streams, byte arrays and DatagramPackets on both sides. Only the Object*Streams need be recreated.

## 5. Datagram sizes

The IP spec allows for datagrams up to 65535 bytes in length, *including* the IP header. If the underlying protocol layers cannot support this size (Ethernet's MTU is 1500 bytes), IP fragments the datagrams into several smaller datagrams. On the receive side, IP reassembles the datagram before delivering it to higher layer protocols, like UDP. If any of the fragments do not arrive at the destination, the entire datagram is discarded, i.e. there is no partial delivery of IP and therefore UDP datagrams.

Since the normal IP header is 20 bytes long and the UDP header is always 8 bytes long, one would expect the maximum UDP data length to be 65535-8-20 = 65507. Somehow, however, the combination of Win2k and JDK1.3.1 manages to successfully send as much as 65527 bytes per datagram. I would be interested to hear whether users of a real operating system experienced the same.

It is *very important* to note that although the IP spec allows for datagrams up to 65535 bytes, it only requires implementations to support up to 576 byte IP datagrams including IP and higher protocol headers. Since the maximum IP header length is 64 and the UDP header length is 8, it is safe to send up to 504 byte UDP datagrams and **expect** the receiving side to handle it (yes, even your Palm Pilot if it has a TCP/IP stack). I have not come across a full size (i.e. non-embedded) system that cannot handle the full 64k-1, though.

## 6. Effect of fault conditions

UDP does not gaurantee delivery or notification of non-delivery. If you send a unicast packet to a host that does not exist, is down or is not listening on that port, you will not know about it. If you send a broadcast or multicast packet and nobody receives it or is even listening, you will not know about it.

On Win2k the network adapter settings are reset if it is detected that the link is not available. With Ethernet, for example, if you unplug the LAN cable so that there is no link available, Win2K detects this and effectively shuts down the adapter at the IP level. It clears its IP address and will not attempt to use it. The effect of this is that sockets cannot bind to a port, so all new *Socket calls fail. Sockets that are already created function correctly if you unplug and replug the cable.

On my notebook, local communication (sender and listener on the same machine) began to fail when I unplugged the LAN cable. It gets nastier than this:- a listener started before I unplugged the cable could not hear traffic from a sender started after I had plugged the cable back in. But wait, there's more! I started another listener after the cable was back in and it *and the listeners started before I unplugged the cable*, all receive the multicasts again.

On WinNT4, my experience has been that the adapter is not "shutdown" when the cable is unplugged and one does not have these weird effects.

## 7. Multiple listeners and unicast packets

Since one can send unicast packets using the same MulticastSocket instance as for ones multicasts, it makes sense to mention how unicasts are handled when there is more than one listener, which can only be when they are all on the same machine.

Unicast traffic sent to the port will be received by only one of the listeners with a socket bound to the port. With my test setup, the last socket to bind to the port receives the unicast traffic. On WinNT4, the first one to bind receives it. I don't know of any rules covering how unicast traffic should be handled in the case of multiple listeners, so don't rely on it being handled in any particular way.

## 8. Further reading

See the RFCs for IP(791), UDP(768) and IP multicasting(1112). Compared to some of the ISO and IEEE stuff I've seen, they're recreational reading material.

# APPENDICES

## A. Protocol "reliability"

You may have heard TCP described as a "reliable" protocol and UDP as an "unreliable" protocol. It is easy, but dangerous, to jump to conclusions about what this means. Being "reliable" does not mean that TCP will deliver your data under all circumstances (try unplugging the LAN cable for a day and see). Being "unreliable", does not mean UDP will arbitrarily throw away your data. "Unreliable" is a loaded term and I prefer to use "non-reliable" which indicates more that it lacks the gaurantees of a "reliable" protocol, rather than labelling it as some sort of untrustworthy servant.

Enough about what reliability, or lack of it, does not mean. A "reliable" protocol like TCP guarantees that it will deliver your data correctly and in order of transmission *or inform you that it could not*.

A "non-reliable" protocol, like UDP, does what is called "best-effort delivery". Essentially, given enough available resources (buffers, bandwidth etc) UDP will deliver your data correctly. It will not deliver incorrect data, but it could deliver data in a different order to which it was sent or not at all.

The NFS (Network File System) protocol uses UDP to communicate between the server and the client. IMHO, this is a testament to the "reliability" of UDP as a transport. Of course, NFS implements its own reliability mechanisms (timeouts and retransmissions) on top of UDP to be sure.

## B. Stream vs Datagrams

The differences between TCP and UDP don't end with reliability. They are fundamentally different in their data model. TCP is stream based and UDP is datagram based. This means that with UDP, if data is lost or delivered out of order, it happens with datagram granularity.

Since TCP is stream based, it does not honour your message boundaries. If you implement your own message passing system using TCP, you will find that doing a send() call of n bytes on one side of the connection does not necessarily result in n bytes being returned by the "corresponding" read() call on the other side. TCP rides on top of IP, which is datagram based, so there is packetizing happening when TCP data is sent, but TCP is at liberty to split your send() up into several actual packets or to coalesce several send() operations into one packet.

## C. nCasting

In the case of TCP, the number of intended recipients of transmitted data is always exactly one (like a telephone call). In general, this is not the case. Everybody is aware of broadcast communication (like radio or television) where there is one sender and any number of recipients. As most people know the same exists in data communications.

Broadcast communication is frowned upon by network admins because they spend a huge portion of their budget trying to provide bandwidth using network switches, only to have this all defeated by broadcast traffic being delivered to every segment of their LANs. Broadcast communication also causes an interrupt and the associated processing on every node on the connected LAN, always. Ones Ethernet hardware, for example, cannot determine whether the host is interested in any particular broadcast packet and must therefore deliver the packet to the upper protocol layers to make the decision. This is the reason Doom 1.1 network games were banned on many LANs. The number of broadcasts used caused such high interrupt processing loads on all the hosts on networks where it was played. Thankfully, Doom 1.2 came along to avert boredom during my time at university.

Where broadcasting is a mechanism intended to deliver data to all hosts on a network or subnetwork, multicasting is a mechanism to deliver data to a group of interested hosts on a network. Many network adapters provide some sort of rudimentary multicast filtering. In many cases, a host not interested in a particular multicast group will not even be interrupted by its network hardware.

In the TCP/IP protocol family, UDP is used for broadcast and multicast (and some unicast) traffic. As a result, broadcast and multicast traffic is datagram based and non-reliable.

Reliability, datagram vs stream based and unicast vs multicast/broadcast traffic are all orthogonal concepts. It is not inconceivable to have a reliable, stream based multicast protocol, or any other combination of those features.

## D. IP Multicast addresses

All class D IP addresses are multicast addresses. Class D IP addresses are those that begin with 1110, that is, all addresses from 224.0.0.0 to 235.255.255.255. Some are pre-assigned for specific applications, but most are available for forming ad hoc multicast groups. There is a mapping between IP multicast addresses and Ethernet addresses, described in **RFC1112:** *"An IP host group address is mapped to an Ethernet multicast address by placing the low-order 23-bits of the IP address into the low-order 23 bits of the Ethernet multicast address 01-00-5E-00-00-00 (hex). Because there are 28 significant bits in an IP host group address, more than one host group address may map to the same Ethernet multicast address."*

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📰 Non-virtual Methods in Java

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 21st issue of "The Java(tm) Specialists' Newsletter". In South Africa the age of adulthood is 21, so I hereby declare this newsletter to be "grown up". No more childish jokes, running in the streets or teasing Sun for writing such a stupid language ;-)

This newsletter serves to illustrate why you should always recompile ALL your code when you get a new build of someone's library. It also, as a side issue, demonstrates how you can write non-virtual methods in Java.

Before I get started, a word of thanks to Dr. Jung for giving me this idea on Monday. I persuaded him to promise me another of his excellent newsletters, which should have been due at least this week, but he managed to deflect my attention with an idea that led to this newsletter.

Please take a few minutes to think of who you know that would be interested in receiving this newsletter and forward it to them.

## Non-virtual Methods in Java

In C++ you can mark a method to be virtual", which tells the compiler that you will want to use the most derived method in the object hierarchy. Virtual therefore means that if you have a class A with method f() and a subclass B with the method f(), and you call the method f() on a handle of A pointing to a B, then B's f() gets called. If you left out the "virtual" keyword, it would cause A's f() to get called, i.e. it is bound at compile time, rather than runtime.

In Java, on the other hand, ALL methods are virtual, i.e. the most derived method is always called, unless of course (read on). During many of the Java courses I presented, I was faced with the question from hardened criminals (oh no, I meant C++ programmers) of why Java does not support non-virtual methods (these questions usually get asked by the same guys who ask why Java doesn't support multiple implementation inheritance and operator overloading *groan*).

Let's look at some code:

```java
//: A.java
public class A {
  public void f() { System.out.println("A's f()"); }
}

//: B.java
public class B extends A {
  public void f() { System.out.println("B's f()"); }
}

//: C.java
public class C {
  public static void main(String[] args) {
    A a = new A();
    a.f();
    ((A)new B()).f();
    B b = new B();
    b.f();
  }
}
```

When we run this, we get the result of:

```
A's f()
B's f()
B's f()
```

A typical question for the Sun Certified Java Programmer exam, and quite obvious to most of us.

The question is, if I have an object of instance B, is it possible to call its parent's f()? Consider class D:

```java
//: D.java
import java.lang.reflect.*;
public class D {
  public static void main(String[] args) throws Exception {
    Method f = A.class.getDeclaredMethod("f", new Class[0]);
    f.invoke(new A(), new Object[0]);
    f.invoke((A)new B(), new Object[0]);
    f.invoke(new B(), new Object[0]);
  }
}
```

We are calling f() of class A, but still, the output remains:

```
A's f()
B's f()
B's f()
```

A few months ago, I was asked about this and after battling for a while, gave up and declared that it is not possible to run class C and get the output of:

```
A's f()
A's f()
B's f()
```

Now for a little trick that changes all that, we simply make f() private in A and recompile ONLY A:

```
//: A.java
public class A {
   private void f() { System.out.println("A's f()"); }
}
```

We have not recompiled B, C or D, so when we run C, we expect to get some warning or error, but alas, our output for C becomes:

```
A's f()
A's f()
B's f()
```

I tried this out using the JDK 1.3.0_01 and JBuilder 3.0 JDK 1.2 and it worked without problems. Please send me a note if you find a version of Java that somehow gives back a VerifyError or an AccessException.

Our class D does not work, but gives us a runtime error, because A.f() is now private. This is the first time where I've seen reflection resulting in safer code!

When should you use this idea? Please don't, but please at the same time be aware that if you get a new library, even if it's the same version number, you have to recompile every line of your Java code, just to be sure.

A better way of achieving the same goal of non-virtual methods is to use static methods which we then pass a handle to an instance of the class. This much clearer approach was suggested by a founding member of The Contractor's Guild.

```
//: A.java
public class A {
   public static void f(A a) { System.out.println("A's f()"); }
}

//: B.java
public class B extends A {
   public static void f(B b) { System.out.println("B's f()"); }
}

//: C.java
public class C {
   public static void main(String[] args) {
      A a = new A();
      a.f(a);
      a = new B();
      a.f(a);
      B b = new B();
      b.f(b);
   }
}
```

In summary, always remember to recompile all your classes, since you don't know which ones will be affected by someone else's change!

---
*Warning Advanced:*
*In JDK 1.1.x, final methods were inlined at compile-time, further necessitating a complete compile if a library was changed.*
---

Thanks for taking the time to read this newsletter, please also take the time to give me feedback, I always appreciate it.

Heinz

---

## Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

| ☐ IT Professional | ☐ General Computer | ☐ Design and Graphics | ☐ Linux | ☐ Freeware/Shareware |
|---|---|---|---|---|
| ☐ Games | Intranet | Computer Hardware | Web Design | ☐ Computer Software |
| ☐ UNIX | ☐ Web Software | ☐ Windows Software | ☐ Personal Finance | ☐ Programming Software |
| ☐ Servers | ☐ C/C++ | ☐ Powerbuilder | ☐ Perl | ☐ XML |
| ☐ Java | ☐ SQL | ☐ Design/Graphics | ☐ Utilities | ☐ Handhelds |

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📖 Package Versioning

**Author: Herman Lintvelt (Polymorph Systems)**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 26th issue of "The Java(tm) Specialists' Newsletter". My silence these past two weeks was due to having my hands full moving my outlook contacts list to the new mailserver. Yes, it has finally happened! In addition, I've done some cosmetic changes to the archive of newsletters and as you can see, the newsletter is now sent out in HTML format with syntax highlighted code. I want to thank Peter Carruthers for letting me use some ideas from his newsletter for this new format.

I am very grateful to fellow Java guru Herman Lintvelt for authoring this week's newsletter. He is the only person I know who understands the javax.swing.JTree, perhaps I should call him Dr. Swing. In case you need some serious Java GUI development done, please send him an email, or if you have GUI code written by amateurs that needs to be cleaned up.

Herman did his Bachelor of Science Degree in Computer Science in 1998 at "University" of Stellenbosch, and since then has worked almost exclusively in Java, especially using Swing and JDBC, with bits of JAI and JMF. After only 2 years as sucker^H^H^H^H^H^Hemployee, he decided it's not for him and recently started Polymorph Systems to enter the "freelance contractor" market.

Please remember to forward this newsletter to as many Java enthusiasts as you know who might be interested in receiving such a newsletter. Special thanks to the person who sent the last newsletter to their local JUG, it caused about 30 subscriptions :-).

And now for Herman's first newsletter...

---

# Package Versioning

While having a (rare) bit of idle time in this coldest, wettest winter of probably the last decade down here at the Southern tip of Africa, I thought it might be good to write about package versioning. Talking of wettest, I live on a wine-farm near Stellenbosch, and were it not for my trusty old '76 Jeep, I would have been cut off from civilization last week, as the dirt road was turned into a muddy river, but this is not a 4x4 newsletter, so back to Java(tm)...

Versioning of packages is a very good discipline to follow. Yes, if you have source-control package like SourceSafe, CVS, Clearcase, etc, you can (probably) be rest assured that you can keep track of your sourcecode versions. But what happens if your jar-files are finally (after months of intensive testing by expert testers ;-) distributed to the client and the *totally unexpected* happens: a bug shows up! You need to be able to determine whether the client had an old version of the package, as the newest version definitely fixed all the remaining bugs.

I like simple things (that's why this newsletter is not complex), and with package versioning it must be the same: it must be easy and fast to keep the package version information up to date (else I will be too tempted to ignore the versioning process). In a project I worked on previously, we added a class to each java package which implemented the same interface and returned the major, minor, patch and build numbers as hard-coded strings. Added to this was a "PackageVersionInterrogator" class that would then get these strings from all the packages when the debug logger required them. This had a few disadvantages:

1. the debug logger had to know all the packages for which it had to get the package versions (thus any packages from where a class might \*possibly\* be loaded, would have to be included in the list of known packages for the debug logger)

2. if the Java app could not even start up, then the client could not browse the jar-file for versioning information, as the version info was contained in class files.

But Alas!!, once again our friends at Sun had a lot of foresight. A while back I "discovered" the existence of the java.lang.Package class, and what's more: this class contains all kinds of nice accessor methods for versioning information! How do you use it? Easy, for every jar file in your project (where every jar file can contain one or more java packages), you need to create your own manifest.mf file (or whatever name you give to it), and invoke the option of the jar-tool to use this manifest file.

Let's look at how this manifest file will look. Suppose we have some classes in a package called com.worldsgreatestApps.java.media and some in a package called com.worldsgreatestApps.java.media.ui. When we run the jar-tool to create a jar-file for these classes, we use the following manifest.mf file (which will be put in the \META-INF subdir inside the jar file):

```
Manifest-Version: 1.0
Main-Class: jmftest.PackageTest
Specification-Title: Java Media Framework
Specification-Version: 2.0
Specification-Vendor: Sun

Name: com/worldsgreatestApps/java/media/
Implementation-Title: WorldsGreatestApps Super Media Framework
Implementation-Version: 1.0.2
Implementation-Vendor: Polymorph Systems

Name: com/worldsgreatestApps/java/media/ui
Specification-Title: Java Media Framework
Specification-Version: 2.0
Specification-Vendor: Sun
Implementation-Title: WorldsGreatestApps Super Media Framework UI classes
Implementation-Version: 1.1.1
Implementation-Vendor: Polymorph Systems
```

Some observations:

1. Main-Class is not part of this versioning, it indicates the main class to run in the jar file, and thus enables you to "run" this jar file directly by using java -jar jarfilename.jar.

2. Specification-[Title/Version/Vendor] specifies information about the specification implemented inside the package(s)

3. Implementation-[Title/Version/Vendor] specifies information about the specific implementation contained in the package.

4. Name (and the empty line in front of it) specifies the name of a package, and must preceed the section for every package in the manifest file. Thus all packages in the jar-file for which we need versioning info (should be all the packages) will have a section in the manifest file.

5. Both the Specification and Implementation entries can be specified in the "general" section of the manifest file

(like the first Specification entries in the example), and these entries will then be used to describe packages that do not have specific entries in the manifest file.

6. The *X-Version* entry values are -by convention- in the form of *major.minor.micro(/patch)*, but it can basically be any descriptive string.

7. The last line of the manifest file should be ended with a newline, else the last line will not be read in probably by the classloader.

How do I use this information in my java code? Well, if the class loader finds package information like this for the package that a class belongs to, it will create a java.lang.Package object for that class, which you can get a handle to by calling on your classinstance.getClass().getPackage(). This object has some methods to get all the info specified in the manifest file.

If no information is specified for a specific package, the classloader first looks to see if there is a "general section" for the version information (i.e. Specification and Implementation entries at the top of the file, before the first empty line). If it is there, it will use this info to populate the Package object; else it will create a Package object that will return null on the version-info method calls.

What's the advantages of using this method to do your package versioning? Well, it's simple and easy to use. Your debug logger can use the standard Java(tm) API to get package version information, and can get a handle on all the loaded packages by calling Package.getPackages(), for example:

```java
/** simple class to list all loaded packages with */
public class PackageLister {
  //...
  public static void listPackages() {
    Package[] allPackages = Package.getPackages();
    System.out.println("All loaded packages:");
    for(int i=0; i < allPackages.length; i++) {
        System.out.println("" + (i+1) +
          ": " + allPackages[i].getName() +
          ": " + allPackages[i].getImplementationTitle() +
          ", version: " + allPackages[i].getImplementationVersion());
    }
  }
  //...
}
```

Another big advantage: on the client site it's easy to open the jar file and view the manifest file manually if (by some operating system or network error of course) your application won't even start up enough to produce debug output.

I hope this inspired those *few* of you who are not using package versioning techniques to start doing it, and those who are using overly complex techniques or tools, to simplify. I'm very interested to know what versioning "systems" are being used out there, so please feel free to email me at herlintvelt@icqmail.com with your ideas and descriptions of the systems you are using for versioning your projects.

Vriendelike groete

Herman

# Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

☐ IT Professional  ☐ General Computer  ☐ Design and Graphics  ☐ Linux  ☐ Freeware/Shareware

☐ Games  Intranet  Computer Hardware  Web Design  ☐ Computer Software

☐ UNIX  ☐ Web Software  ☐ Windows Software  ☐ Personal Finance  ☐ Programming Software

☐ Servers  ☐ C/C++  ☐ Powerbuilder  ☐ Perl  ☐ XML

☐ Java  ☐ SQL  ☐ Design/Graphics  ☐ Utilities  ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📖 Insane Strings

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 14th issue of "The Java(tm) Specialists' Newsletter", where we look at things that other newsletters would not dare to mention. Please please use the ideas presented in this newsletter with caution, they can really mess up a project if used (incorrectly). A lot of them are educational rather than practical so that we can understand Java better. This is true especially for this newsletter, in which we mutate Strings.

Thanks if you forwarded last week's newsletter to some of your colleagues and friends, I got a few subscriptions as a result of referrals. The membership has been growing steadily since I started this newsletter in December 2000. It has now broken through the 400 barrier and I'm thinking of moving it to a proper list server before reaching 500. My main concern is that if it is a free list server it might start sending you spam, which I really would not like. Any suggestions of free list servers that do not generate spam would be most welcome. My current preference is for Topica - which I will choose unless I hear from you.

## Playing with your sanity - Strings

Have a look at the following code:

```java
public class MindWarp {
  public static void main(String[] args) {
    System.out.println(
      "Romeo, Romeo, wherefore art thou oh Romero?");
  }
  private static final String OH_ROMEO =
    "Romeo, Romeo, wherefore art thou oh Romero?";
  private static final Warper warper = new Warper();
```

```
        }
```

If we are told that the class Warper does not produce any visible output when you construct it, what is the output of this program? The most correct answer is, "you don't know, depends on what Warper does". Now THERE's a nice question for the Sun Certified Java Programmer Examination.

In my case, running "java MindWarp" produces the following output

```
C:> java MindWarp <ENTER>
Stop this romance nonsense, or I'll be sick
```

And here is the code for Warper:

```java
import java.lang.reflect.*;

public class Warper {
  private static Field stringValue;
  static {
    // String has a private char [] called "value"
    // if it does not, find the char [] and assign it to value
    try {
      stringValue = String.class.getDeclaredField("value");
    } catch(NoSuchFieldException ex) {
      // safety net in case we are running on a VM with a
      // different name for the char array.
      Field[] all = String.class.getDeclaredFields();
      for (int i=0; stringValue == null && i<all.length; i++) {
        if (all[i].getType().equals(char[].class)) {
          stringValue = all[i];
        }
      }
    }
    if (stringValue != null) {
      stringValue.setAccessible(true); // make field public
    }
  }
  public Warper() {
    try {
      stringValue.set(
        "Romeo, Romeo, wherefore art thou oh Romero?",
        "Stop this romance nonsense, or I'll be sick".
          toCharArray());
      stringValue.set("hi there", "cheers !".toCharArray());
    } catch(IllegalAccessException ex) {} // shhh
  }
}
```

How is this possible? How can String manipulation in a completely different part of the program affect our class MindWarp?

To understand that, we have to look under the hood of Java. In the language specification it says in §3.10.5:

"Each string literal is a reference (§4.3) to an instance (§4.3.1, §12.5) of class String (§4.3.3). String objects have a constant value. String literals-or, more generally, strings that are the values of constant expressions (§15.28)-are "interned" so as to share unique instances, using the method String.intern."

The usefulness of this is quite obvious, we will use less memory if we have two Strings which are

equivalent pointing at the same object. We can also manually intern Strings by calling the intern() method.

The language spec goes a bit further:

1. Literal strings within the same class (§8) in the same package (§7) represent references to the same String object (§4.3.1).

2. Literal strings within different classes in the same package represent references to the same String object.

3. Literal strings within different classes in different packages likewise represent references to the same String object.

4. Strings computed by constant expressions (§15.28) are computed at compile time and then treated as if they were literals.

5. Strings computed at run time are newly created and therefore distinct.

6. The result of explicitly interning a computed string is the same string as any pre-existing literal string with the same contents.

This means that if a class in another package "fiddles" with an interned String, it can cause havoc in your program. Is this a good thing? (You don't need to answer ;-)

Consider this example

```
public class StringEquals {
  public static void main(String[] args) {
    System.out.println("hi there".equals("cheers !"));
  }
  private static final String greeting = "hi there";
  private static final Warper warper = new Warper();
}
```

Running this against the Warper produces a result of true, which is really weird, and in my opinion, quite mind-bending. Hey, you can SEE the values there right in front of you and they are clearly NOT equal!

BTW, for simplicity, the Strings in my examples are exactly the same length, but you can change the length quite easily as well.

Last example concerns the HashCode of String, which is now cached for performance reasons mentioned in "Java Idiom and Performance Guide", ISBN 0130142603. (Just for the record, I was never and am still not convinced that caching the String hash code in a wrapper object is a good idea, but caching it in String itself is almost acceptable, considering String literals.)

```
public class CachingHashcode {
  public static void main(String[] args) {
    java.util.Map map = new java.util.HashMap();
    map.put("hi there", "You found the value");
    new Warper();
    System.out.println(map.get("hi there"));
    System.out.println(map);
  }
  private static final String greeting = "hi there";
}
```

The output under JDK 1.3 is:

```
You found the value
{cheers !=You found the value}
```

Under JDK 1.2 it is

```
null
{cheers !=You found the value}
```

This is because in the JDK 1.3 SUN is caching the hash code so if it once calculated, it doesn't get recalculated, so if the value field changes, the hashcode stays the same.

Imagine trying to debug this program where SOMEWHERE, one of your hackers has done a "workaround" by modifying a String literal. The thought scares me.

There is of course a small keyword that would have stopped this problem, namely "final". I got into the habit a few months ago to make all my data members final where possible, and it has paid off more than once. Surprisingly, the char array in String is not final.

Consider the following example code:

```java
public class Bla {
  private char[] c1 = "hello".toCharArray();
  private final char[] c2 = "bye".toCharArray();
  public String toString() {
    return c1 + ", " + c2;
  }
}

import java.lang.reflect.*;

public class BlaTest {
  private static Field c1;
  private static Field c2;
  static {
    try {
      c1 = Bla.class.getDeclaredField("c1");
      c1.setAccessible(true);
      c2 = Bla.class.getDeclaredField("c2");
      c2.setAccessible(true);
    } catch(NoSuchFieldException ex) { }
  }
  public static void main(String[] args) {
    Bla bla = new Bla();
    try {
      c1.set(bla, "mutatedc1".toCharArray());
      c2.set(bla, "mutatedc2".toCharArray());
    } catch(IllegalAccessException ex) {
      ex.printStackTrace();
    }
    System.out.println(bla);
  }
}
```

When I run my program, I can quite happily change c1, but when I try to change c2 I get an exception. String has no reason for value to be non-final, so it should be final. If you have contacts at SUN, please forward them this newsletter and ask them to make value final. It might stop some nasty Java viruses from completely messing up the JVM.

Until next week, and please remember to forward this newsletter and send me your comments.

Heinz

# Pick your topics, and we'll send you great deals, free information, and special offers by email from Focalex.

☐ IT Professional  ☐ General Computer  ☐ Design and Graphics  ☐ Linux  ☐ Freeware/Shareware

☐ Games  Intranet  Computer Hardware  Web Design  ☐ Computer Software

☐ UNIX  ☐ Web Software  ☐ Windows Software  ☐ Personal Finance  ☐ Programming Software

☐ Servers  ☐ C/C++  ☐ Powerbuilder  ☐ Perl  ☐ XML

☐ Java  ☐ SQL  ☐ Design/Graphics  ☐ Utilities  ☐ Handhelds

Get FREE STUFF, special offers and information on the Java, C/C++,
SQL, XML and another languages programming topics YOU want to hear
about delivered to your inbox!
Unix, Windows Software/HardWare, Handhelds - just pick YOUR
interests and start getting the stuff you want today. The best
free info and deals on computers, software and much more by email.
**SUBSCRIBE for FREE HERE**

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📰 Self-tuning FIFO Queues

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 24th issue of "The Java(tm) Specialists' Newsletter". Winter has arrived in Cape Town with great rain and cold temperatures, much to the chagrin of the author. The only way to combat such is to sit next to the heater with a glass of South African 1997 Zonnebloem Cabernet Sauvignon and write Java newsletters. This newsletter has taken the most time to date - performance tests tend to have that effect on me :(((

Before I go on today's voyage, some feedback regarding the Socket Wheel idea:

I'm not quite sure who gave the best contributions regarding possible improvements for last newsletter's SocketWheel. Neil Bartlett (Algorithmics Canada) and John Vlissides (IBM USA) pointed out that the design was very close to the Reactor pattern by Schmidt. This is a great plus mark for patterns; they truly are things that OO developers around the world would come up with independently. Josh Rehman pointed out that it is not too clever to try minimze memory as that is very cheap nowadays, which I agree with. He also pointed out that JDK 1.4 has non-blocking IO making it possibly a lot easier to implement such a SocketWheel. Ecce Jezuch (Poland) suggested using the available() method to find out how many bytes would be available without blocking, but unfortunately under Windows the JDK always returned 0.

James Pereira provided some excellent information regarding sockets. It's quite technical, so I'm including it verbatim:

"Registered Ports, ports between 1024 and 49151, are listed by the IANA and on most systems can be used by applications or programs executed by users. Table C.2 specifies the port used by the server process as its contact port. The IANA registers uses of these ports as a convenience to the Internet community. To the extent possible, these same port assignments are used with UDP. The Registered Ports are in the numerical range of 1024-49151. The Registered Ports between 1024 and 5000 are also referred to as the Ephemeral Ports. At least on Windows , The TCP/Stack (OS) re-uses these ports internally on every socket connection cycling from 1024...5000 wrapping around to 1024 again. This could lead to some interesting problems if sockets are opened and close very quickly as there is usually a time delay before that port is made available again...

"Second, the number of user-accessible ephemeral ports that can be used to source outbound connections is configurable with the MaxUserPort registry entry (HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters). By default, when an application requests any socket from the system to use for an outbound call, a port between the values of 1024 and 5000 is supplied. You can use the MaxUserPort registry entry to set the value of the highest port number to be used for outbound connections. For example, setting this value to 10000 would make approximately 9000 user ports available for outbound connections. For more details, see RFC 793. See also the MaxFreeTcbs and MaxHashTableSize registry settings (HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters).

"BTW: The stack and memory size issue is only an NT issue. On NT the stack size is configurabale via linker and EditBin.exe unfortunately we don't want to mess with the VM."

This information should solve the problem that I encountered with too many sockets causing exceptions.

Ideally we should be able to specify stack size per thread as some threads will need a lot and others only a little, so I still think that we have a problem with many threads and their stack size.

Another excellent contribution was made by Craig Larman, author of "Java 2 Performance and Idiom Guide", who suggested using an approach of multiplexing sockets described in his book to minimize the number of sockets needed per client connection. Next week I will try and write about this idea and give you a solution that hopefully will work acceptably. I always recommend Craig's book as an excellent book for the Java fundi who wants to have his mind activated regarding performance ideas. It is very difficult to write anything about performance as it is so dependent on your implementation and hardware. This leads me to this week's newsletters on self-tuning FIFO queues....

## Self-tuning FIFO Queues

HotSpot(tm) has caused a lot of trouble for the Java Specialist in the field, especially those of us with a few years experience. All of a sudden, all the hard-earned performance knowledge was wiped out in one foul swoop. The only thing left for us to do was to write the simplest code that could possibly work and let the HotStop compiler sort out performance for us. I liken it to the great financial crisis of Japan in the 90's, where no one knew whether he was coming or going and all the old certainties went out of the window. Luckily, unlike in Japan, we are not taking this too seriously, so our windows are still closed. Fortunately everyone knows that Java is slow ;-)

A lot of the performance tricks we used in old code actually make our code slower under HotSpot. Since we don't know what the performance of our code will be for a specific platform, would it be completely hairbrained to write self-tuning code? Write 3 algorithms, let the program measure on the target platform which performs best, and choose that algorithm for the duration of the VM.

To illustrate this idea, I want to write a FIFO queue that is based on a java.util.List implementation. A while ago I discovered that java.util.ArrayList is sometimes faster than java.util.LinkedList for FIFO queue implementations. The switch over occurs at a specific point in time, which we can measure beforehand. The switch-over point is dependent on the VM, whether we are using HotSpot(tm), etc. For example, on my little notebook with 256MB RAM and Pentium III 700, the cross-over point is 50 elements in the list when I use HotSpot, but 500 elements when I switch off hotspot compiling (sometimes!).

The interface that the FIFO queues will implement is very simple:

```java
public interface FIFO {
  /** Add an object to the end of the FIFO queue */
  boolean add(Object o);
  /** Remove an object from the front of the FIFO queue */
  Object remove();
  /** Return the number of elements in the FIFO queue */
  int size();
}
```

We implement this interface and extend ArrayList and LinkedList:

```java
// FIFOArrayList.java
import java.util.ArrayList;
```

```java
public class FIFOArrayList extends ArrayList implements FIFO {
  public Object remove() {
    return remove(0);
  }
}
// FIFOLinkedList.java
import java.util.LinkedList;
public class FIFOLinkedList extends LinkedList implements FIFO {
  public Object remove() {
    return remove(0);
  }
}
```

We also write a SwappingOverFIFOQueue which has values for HIGH and LOW water marks. When we reach a HIGH water mark and we are busy using an ArrayList, we start using a LinkedList. On the contrary, if we reach a LOW water mark and we are busy using a LinkedList we start using an ArrayList.

In foresight to my next example, I have made it possible to set the watermarks, which also checks the optimal list types for all the lists currently in the system. We have to be careful to not get a memory leak by keeping handles to the instances of the SwappingOverFIFOQueue so we use WeakReferences to hold the references. Have a look at newsletter 15 for a discussion on Weak / Soft References.

```java
// SwappingOverFIFOQueue.java
import java.util.*;
import java.lang.ref.WeakReference;
public final class SwappingOverFIFOQueue implements FIFO {
  /** The low value after which we switch over to ArrayList */
  private static int LOW = 30;
  /** The high value after which we switch down to LinkedList */
  private static int HIGH = 70;
  /** This list contains weak references of instances of this
      class */
  private static List instances = new LinkedList();
  /** We add the weak references in an initializer block */
  {
    instances.add(new WeakReference(this));
  }
  /** When we set the low and high water marks we go through all
      the existing instances and check for the optimal list type.
      If the references is null we remove the WeakReference from
      our instance list. */
  public static void setWaterMarks(int low, int high) {
    LOW = low;
    HIGH = high;
    Iterator it = instances.iterator();
    while(it.hasNext()) {
      WeakReference ref = (WeakReference)it.next();
      SwappingOverFIFOQueue q = (SwappingOverFIFOQueue)ref.get();
      if (q == null) {
        it.remove();
      } else {
        q.checkOptimalListType();
      }
    }
  }
```

```java
    private List list = new ArrayList();
    public Class getListType() { return list.getClass(); }
    private int size = 0;
    public boolean add(Object o) {
      try {
        list.add(o);
        return true;
      } finally {
        if (++size == HIGH) checkOptimalListType();
      }
    }
    public Object remove() {
      try {
        return list.remove(0);
      } finally {
        if (--size == LOW) checkOptimalListType();
      }
    }
    public int size() {
      return size;
    }
    private void checkOptimalListType() {
      if (size >= HIGH && (!(list instanceof LinkedList))) {
        list = new LinkedList(list);
      } else if (size <= LOW && (!(list instanceof ArrayList))) {
        list = new ArrayList(list);
      }
    }
  }
```

My test program takes the number of entries in the queue and then illustrates how often we can add/remove in 2 seconds for each of the queues. I found that you get the best performance results when you run your tests for about 2 seconds each, so I count iterations rather than milliseconds.

```java
  // SwappingOverFIFOQueueTest.java
  import java.util.*;
  public class SwappingOverFIFOQueueTest {
    private static int ENTRIES;
    public static void test(FIFO queue) {
      for (int i=0; i<ENTRIES; i++) {
        queue.add(new Object());
      }
      long up_to = System.currentTimeMillis() + 2000;
      int iterations = 0;
      while(System.currentTimeMillis() <= up_to) {
        queue.add(new Object());
        queue.remove();
        iterations++;
      }
      System.out.println(queue.getClass());
      System.out.println("\t" + iterations + " iterations");
    }
    public static void main(String[] args) {
```

```
        if (args.length != 1) {
          System.out.println(
            "Usage: java SwappingOverFIFOQueueTest entries");
          System.exit(1);
        }
        ENTRIES = Integer.parseInt(args[0]);
        System.out.println("Entries = " + ENTRIES);
        test(new FIFOArrayList());
        test(new FIFOLinkedList());
        SwappingOverFIFOQueue q = new SwappingOverFIFOQueue();
        test(q);
        System.out.println("Current queue implementation " +
          q.getListType().getName());
      }
    }
```

On my notebook, when I run this program with 0 entries, I get the following output:

```
  Entries = 0
  class FIFOArrayList
          4552883 iterations
  class FIFOLinkedList
          2551017 iterations
  class SwappingOverFIFOQueue
          3594810 iterations
```

With 100 entries I get:

```
  Entries = 100
  class FIFOArrayList
          1800877 iterations
  class FIFOLinkedList
          2509328 iterations
  class SwappingOverFIFOQueue
          2158451 iterations
```

And with 10000 entries I get:

```
  Entries = 10000
  class FIFOArrayList
          49500 iterations
  class FIFOLinkedList
          812933 iterations
  class SwappingOverFIFOQueue
          758657 iterations
```

We can thus see that the SwappingFIFOQueue is always faster than the worst case and slower than the best case, as one would logically expect. However, I chose the HIGH and LOW values from some tests that I made on my notebook, for that specific JVM. If I take the JDK 1.2.2 that comes with JBuilder, for 100 entries I get:

```
  Entries = 100
  class FIFOArrayList
          1434122 iterations
  class FIFOLinkedList
          1307108 iterations
  class SwappingOverFIFOQueue
          1178115 iterations
```

Or if I use the -Xint mode for JDK 1.3 under Windows to switch off the hotspot compiler, for 100 entries I get

```
Entries = 100
class FIFOArrayList
        497550 iterations
class FIFOLinkedList
        480599 iterations
class SwappingOverFIFOQueue
        392314 iterations
```

In both cases, the values of the SwappingOverFIFOQueue were worse than for both the ArrayList and the LinkedList.

We therefore write a Profiler that finds ideal HIGH/LOW water marks for the JVM that is running on your system and sets up the SwappingOver water marks.

```java
// SwappingOverFIFOQueueProfiler.java
import java.util.*;
/*
  For the sake of brevity we only consider two implementations
  of java.util.List, namely java.util.ArrayList and
  java.util.LinkedList. */
public class SwappingOverFIFOQueueProfiler {
  private static boolean isArrayListFaster(int entries) {
    System.out.println("isArrayListFaster(" + entries + ")");
    return countIterations(new ArrayList(), entries) >
      countIterations(new LinkedList(), entries);
  }
  private static int countIterations(List list, int entries) {
    for (int i=0; i<entries; i++) {
      list.add(new Object());
    }
    long end = System.currentTimeMillis() + 1000;
    int iterations = 0;
    while(System.currentTimeMillis() <= end) {
      iterations++;
      list.add(new Object());
      list.remove(0);
    }
    return iterations;
  }

  static {
    int checks = 0;
    int watermark = 1;
    int bestWatermark = 0;
    for (int i=0; i<16; i++) {
      if (isArrayListFaster(watermark)) {
        bestWatermark = Math.max(watermark, bestWatermark);
        watermark *= 2.0;
      } else {
        watermark *= 0.75;
        if (watermark <= bestWatermark)
          watermark *= 1.25;
```

```
            }
        }
        System.out.println("Best watermark = " + bestWatermark);
        int low = (int)(bestWatermark * 0.75);
        int high = (int)(bestWatermark * 1.25);
        System.out.println("Setting LOW to " +  low +
            " and HIGH to " + high);
        SwappingOverFIFOQueue.setWaterMarks(low, high);
    }
    public static void main(String[] args) {
        SwappingOverFIFOQueueTest.main(new String[] { "0" });
        SwappingOverFIFOQueueTest.main(new String[] { "100" });
        SwappingOverFIFOQueueTest.main(new String[] { "10000" });
    }
}
```

If we load this class in our system then it will do measurements of where the swap-over between ArrayList and LinkedList performance occurs. On my computer, with JDK 1.3 and HotSpot, the swap-over was measured to happen at about 32 entries in the list. When I switch off the HotSpot, it occurs at about 121 entries, and under JDK 1.2.2 it happens at about 303.

After spending about 10 hours on this stupid newsletter, I have to conclude that it would be better to stick to a LinkedList for a FIFO queue as it is a better "average" performer. Perhaps the lesson I've learnt from this newsletter is that we must be careful of writing code which is too complicated as it tends to be more difficult to optimize. As performance guru Craig Larman pointed out though, we must be sure to not ignore performance altogether; our customers might just kill the project if the prototypes perform like dogs.

I always appreciate any feedback, both positive and negative, so please keep sending your ideas and suggestions. Please also remember to take the time to send this newsletter to others who are interested in Java.

Heinz

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Serializing GUI Components Across Network Part I

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 13th issue of "The Java(tm) Specialists' Newsletter". Let's hope my luck does not run out in your estimation and that you don't assign my to a Java loony bin, especially with the things I do in this newsletter.

Please forward this free advanced Java newsletter to as many people as you know who might be interested in Java.

## Serializing GUI Components Across Network

When Swing came out, I was puzzled by the following warning in the javadocs:

Warning: Serialized objects of this class will not be compatible with future Swing releases. The current serialization support is appropriate for short term storage or RMI between applications running the same version of Swing. A future release of Swing will provide support for long term persistence.

My thoughts were: No one in their right mind would want to serialize GUI components, so why are they serializable in the first place? Luckily for me, some of my friends are not in their right mind :) Many thanks to twin-dad-to-be Niko Brummer for this extremely interesting, and probably totally useless idea, though he assures me that he is using it in a real program. Some of the code in this newsletter is from him, other parts I have added. You can easily spot his code by the absense of bugs.

There are some very interesting gotchas that occur because of the dear Swing event dispatch thread, who, having failed at all attempts to find a mate, is still single, and will be for the forseeable future. Due to his hermit nature, we have to be very careful that we do not read from the GUI components or write to them from any thread except the GUI thread, or else face the wrath of a thousand deadlocks (see first newsletter in this series).

If you are serializing the component in response to pushing a GUI button, it will work, because then you are using the event dispatch thread. If you are doing it from any other thread, you may get problems (in

Niko's case the event dispatch thread crashed). The solution is to wrap the gui component in an object of which you control the serialization by specifying your own writeObject() method.

I have written a ComponentSerializer class which wraps the read write functionality and you can either write a Component to an OutputStream or read it from an InputStream without having to worry about what thread you are currently in.

```java
//: ComponentSerializer.java
import java.io.*;
import java.awt.*;
import javax.swing.*;
import java.lang.reflect.*; // wouldn't be right for me to send
                // you a newsletter that doesn't use reflection :)

public class ComponentSerializer {
  public void write(Component comp, OutputStream out)
      throws IOException {
    System.out.println("writing " + comp);
    ObjectOutputStream oout = new ObjectOutputStream(out);
    oout.writeObject(new ComponentEncapsulator(comp));
    oout.reset();
    oout.flush();
  }
  public Component read(InputStream in)
      throws IOException, ClassNotFoundException {
    System.out.println("reading component");
    ObjectInputStream oin = new ObjectInputStream(in);
    ComponentEncapsulator enc =
      (ComponentEncapsulator)oin.readObject();
    return enc.getComponent();
  }
  private class ComponentEncapsulator implements Serializable {
    private final Component comp;
    public ComponentEncapsulator(Component comp) {
      this.comp = comp;
    }
    public Component getComponent() {
      return comp;
    }
    private IOException defaultWriteException;
    private void writeObject(final ObjectOutputStream out)
        throws IOException {
      if (SwingUtilities.isEventDispatchThread()) {
        // This is all that is necessary if we are already in
        // the event dispatch thread, e.g. a user clicked a
        // button which caused the object to be serialized
        out.defaultWriteObject();
      } else {
        try {
          // we want to wait until the object has been written
          // before continuing.  If we called this from the
          // event dispatch thread we would get an exception
          SwingUtilities.invokeAndWait(new Runnable() {
            public void run() {
```

```java
                    try {
                        // easiest way to indicate to the enclosing class
                        // that an exception occurred is to have a member
                        // which keeps the IOException
                        defaultWriteException = null;
                        // we call the actual write object method
                        out.defaultWriteObject();
                    } catch(IOException ex) {
                        // oops, an exception occurred, remember the
                        // exception object
                        defaultWriteException = ex;
                    }
                }
            });
            if (defaultWriteException != null) {
                // an exception occurred in the code above, throw it!
                throw defaultWriteException;
            }
        } catch(InterruptedException ex) {
            // I'm not quite sure what do here, perhaps:
            Thread.currentThread().interrupt();
            return;
        } catch(InvocationTargetException ex) {
            // This can actually only be a RuntimeException or an
            // Error - in either case we want to rethrow them
            Throwable target = ex.getTargetException();
            if (target instanceof RuntimeException) {
                throw (RuntimeException)target;
            } else if (target instanceof Error) {
                throw (Error)target;
            }
            ex.printStackTrace(); // this should not happen!
            throw new RuntimeException(ex.toString());
        }
    }
  }
}
```

I apologize for all the comments in the ComponentSerializer, as we all know, too many comments are often indicative of poorly written code, but I cannot think of a simpler, yet correct, way of doing this. This ComponentSerializer class should handle just about any java.awt.Component derivative thrown at it from any thread, except for components which reference non-serializable components.

We can then write a GUIServer, which accepts a ComponentEncapsulator via TCP/IP and constructs a JFrame containing the component within the ComponentEncapsulator. We only cater for one component per socket, but that could easily be changed.

```java
//: GUIServer.java
import java.io.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.net.*;
```

```java
public class GUIServer {
  public static final int PORT = 4123;
  private static final ComponentSerializer compser =
    new ComponentSerializer();
  public GUIServer() throws IOException {
    System.out.println("Super-duper GUI SERVER started");
    ServerSocket ss = new ServerSocket(PORT);
    while(true) {
      Socket socket = ss.accept();
      try {
        JFrame frame = new JFrame(
          "Component received from " + socket);
        Component comp = compser.read(socket.getInputStream());
        frame.getContentPane().add(comp);
        frame.pack();
        frame.show();
      } catch(IOException ex) {
        ex.printStackTrace();
      } catch(ClassNotFoundException ex) {
        ex.printStackTrace();
      }
    }
  }
  public static void main(String[] args) throws IOException {
    new GUIServer();
  }
}
```

We can then take a Component, for example a JScrollPane containing a JTable, and send it to any OutputStream, e.g. to the network, to the file system for short-term storage, etc. The warning in the Swing source essentially tells us we should not write GUI components onto DAT tapes for long-term archiving.

```java
//: GUIExample.java
import java.io.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.net.*;

public class GUIExample extends JFrame {
  public static final int PORT = 4123;
  private static final ComponentSerializer compser =
    new ComponentSerializer();
  private JScrollPane scrollPane;
  public GUIExample() {
    super("GUIExample Frame");
    scrollPane = new JScrollPane(new JTable(3,4));
    getContentPane().add(scrollPane);
    getContentPane().add(new JButton(
      new AbstractAction("Serialize Table") {
        public void actionPerformed(ActionEvent e) {
          System.out.println("Now we serialize synchronously");
          try {
            Socket socket = new Socket("localhost", PORT);
```

```
            compser.write(scrollPane, socket.getOutputStream());
            socket.close();
          } catch(IOException ex) {
            ex.printStackTrace();
          }
        }
      }), BorderLayout.SOUTH);
    setSize(400, 200);
    show();
  }
  public static void main(String[] args) throws Exception {
    GUIExample ex = new GUIExample();
    ex.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
  }
}
```

Please try this out - I was quite amazed that it actually worked! Start the GUIServer class, then start the GUIExample and press the button. The GUIServer should now open up another JFrame containing the JTable. Now edit the original JTable (press enter after editing, otherwise you'll get an exception when you try to serialize the table), and click on the button again. The GUIServer should now open up yet another JFrame with the JTable containing the latest values. Though I cannot think of a good application for this at the moment, I think it's quite neat that you can do that.

I've also written a second example to show you how to do an asynchronous serialization of components via another thread.

```
//: GUIExample2
import java.io.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.net.*;

public class GUIExample2 extends JFrame {
  public static final int PORT = 4123;
  private static final ComponentSerializer compser =
    new ComponentSerializer();
  private JPanel personalData;
  public GUIExample2() {
    super("Asynchronous Sending GUIExample2 Frame");
    personalData = new JPanel(new GridLayout(0, 2, 5, 5));
    personalData.add(new JLabel("Name: "));
    personalData.add(new JTextField());
    personalData.add(new JLabel("Age: "));
    personalData.add(new JTextField());
    getContentPane().add(personalData, BorderLayout.NORTH);
    getContentPane().add(new JButton(
      new AbstractAction("Serialize Personal Data") {
        public void actionPerformed(ActionEvent e) {
          asyncSerialize(personalData);
        }
      }), BorderLayout.SOUTH);
    setSize(400, 200);
    show();
  }
```

```java
    private void asyncSerialize(final Component comp) {
      new Thread() { {start();} // start from initializer block
        public void run() {
          try {
            Socket socket = new Socket("localhost", PORT);
            compser.write(comp, socket.getOutputStream());
            socket.close();
          } catch(IOException ex) {
            ex.printStackTrace();
          }
        }
      };
    }
    public static void main(String[] args) throws Exception {
      GUIExample2 ex = new GUIExample2();
      ex.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
  }
```

When you try this out, you'll notice that the entire JPanel gets sent across the network to the server.

Were I to use this in a production environment framework, I would add the option of writing asynchronously to the ComponentSerializer and probably do the writing via a ThreadPool.

Next week I will demonstrate that the following can be **true**:

```java
    "hi there".equals("cheers !")
```

Seeing is believing, so until next week

Heinz

---

## 📑 Serializing GUI Components Across Network Part II

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Hi again,

Imagine my horror this morning when I tried to run the code from last nights newsletter and discovered that it generated an exception! On frantic searching I figured that the JBuilder 3.0 compiler produces a different result to the SUN JDK 1.3 compiler. I had only run the program from within JBuilder, so I got quite a surprise that it did not work in SUN. The reason I'm blaming the compiler is because when I ran the class files generated by JBuilder 3.0 with the JDK 1.3 VM it works perfectly.

The problem is that if you call out.defaultWriteObject(), the two compilers have different ideas of which object you are calling this from, because you are inside an inner class. The SUN compiler thinks you are calling it from ComponentSerializer and that is not serializable.

The solution is quite simple, just move the ComponentEncapsulator class out of the ComponentSerializer class and it should work with the JDK 1.3 compiler.

We thus end up with ComponentSerializer:

```java
//: ComponentSerializer.java
import java.io.*;
import java.awt.*;
public class ComponentSerializer {
  public void write(Component comp, OutputStream out)
      throws IOException {
    System.out.println("writing " + comp);
    ObjectOutputStream oout = new ObjectOutputStream(out);
    oout.writeObject(new ComponentEncapsulator(comp));
    oout.reset();
```

```java
          oout.flush();
       }
       public Component read(InputStream in)
           throws IOException, ClassNotFoundException {
         System.out.println("reading component");
         ObjectInputStream oin = new ObjectInputStream(in);
         ComponentEncapsulator enc =
           (ComponentEncapsulator)oin.readObject();
         return enc.getComponent();
       }
     }
```

and ComponentEncapsulator:

```java
   //: ComponentEncapsulator.java
   import java.io.*;
   import java.awt.*;
   import javax.swing.*;
   import java.lang.reflect.*; // wouldn't be right for me to send
               // you a newsletter that doesn't use reflection :)
   class ComponentEncapsulator implements Serializable {
     private final Component comp;
     private IOException defaultWriteException;
     public ComponentEncapsulator(Component comp) {
       this.comp = comp;
     }
     public Component getComponent() {
       return comp;
     }
     private void writeObject(final ObjectOutputStream out)
         throws IOException {
       if (SwingUtilities.isEventDispatchThread()) {
         // This is all that is necessary if we are already in
         // the event dispatch thread, e.g. a user clicked a
         // button which caused the object to be written
         out.defaultWriteObject();
       } else {
         try {
           // we want to wait until the object has been written
           // before continuing.  If we called this from the
           // event dispatch thread we would get an exception
           SwingUtilities.invokeAndWait(new Runnable() {
             public void run() {
               try {
                 // easiest way to indicate to the enclosing class
                 // that an exception occurred is to have a member
                 // which keeps the IOException
                 defaultWriteException = null;
                 // we call the actual write object method
                 out.defaultWriteObject();
               } catch(IOException ex) {
                 // oops, an exception occurred, remember the
                 // exception object
```

```
                defaultWriteException = ex;
            }
        }
    });
    if (defaultWriteException != null) {
        // an exception occurred in the code above, throw it!
        throw defaultWriteException;
    }
} catch(InterruptedException ex) {
    // I'm not quite sure what do here, perhaps:
    Thread.currentThread().interrupt();
    return;
} catch(InvocationTargetException ex) {
    // This can actually only be a RuntimeException or an
    // Error - in either case we want to rethrow them
    Throwable target = ex.getTargetException();
    if (target instanceof RuntimeException) {
        throw (RuntimeException)target;
    } else if (target instanceof Error) {
        throw (Error)target;
    }
    ex.printStackTrace(); // this should not happen!
    throw new RuntimeException(ex.toString());
}
            }
        }
    }
```

This highlights again that we have to be quite careful with Java. In the big project that we work on, we have standardized on the JDK 1.3 compiler and ALL our classes have to be compiled with that. This happened only after much nailbiting because of slight compiler differences of various IDEs.

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

### Serializing Objects Into Database

**Author: Dr. Heinz M. Kabutz**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 20th issue of "The Java(tm) Specialists' Newsletter", where I look at how we can serialize objects into a database using JDBC. Thank you for your continued support, it makes the hours of research and writing worthwhile :-)

This week's newsletter has been the most frustrating one to write; databases usually have that effect on me. Forgive me if this newsletter seems too elementary, I've actually in the past had to dig into the C code of the JDBC-ODBC bridge to figure out why it wasn't working, i.e. it's more complicated than it seems. Java truly is a "write-once-debug-everywhere" language, especially when it comes to JDBC drivers.

If you're crazy enough to use MSSQL as your database, which isn't THAT crazy considering the beautiful administrative tools available and the ease with which it can be set up, you might consider using the JDBC-ODBC bridge to access the database. It's a dog, full of bugs, but it's free (if you don't count the developer's time in writing workarounds as significant). Also, "better the devil you know", which according to "The Wordsworth Dictionary of Cliche" means "Trust the person or thing you are familiar with rather than risking the unknown." Other JDBC drivers are bound to also contain bugs, and at least we know the bugs in the bridge!

## Serializing Objects Into Database

A problem I faced a few months ago, before JDK 1.3 was released with an improved JDBC-ODBC bridge, was how to serialize objects into a relational database using JDBC. At the time, it was quite tricky to do because of bugs in the ODBC bridge. As soon as the object became bigger than 2000 bytes, the method shown below ceased to work. I don't need to mention that I tested it with less than 2000 bytes the first time I wrote the code?

Before we look at how we can write binary objects into a database table, what type of data type should we use? Should we use VARBINARY or IMAGE? The answer, I suppose, depends on how big the object is that you want to write to the database, and which database you're actually talking to (hence my comments regarding write-once-debug-everywhere).

We also have to consider the underlying database when we get a SQLException, so that we can determine what actually went wrong. If the connection goes down temporarily due to a flaky network, it is no good for our application server to crash permanently. In a future newsletter, I will demonstrate how you can find out what went wrong by looking at the SQLException.

Back to the problem of writing serialized Java objects into the database. The most intuitive way of doing it, which as I mentioned before only works in the ODBC bridge of JDK 1.3, is to take the Object, stream it to a ByteArrayOutputStream via an ObjectOutputStream, convert the ByteArrayOutputStream to a byte array and then call the setBytes method on the prepared statement. Note that you have to use PreparedStatement, rather than the normal Statement class, if you want to serialize objects into the database.

To convert it back to a Java object, we simply do the reverse process of reading the byte array, putting it into a ByteArrayInputStream and passing that to an ObjectInputStream. We then read the object, and voila, we have read the object back again.

```java
//: DatabaseTest.java
import java.sql.*;
import java.io.*;
import java.util.Vector;
public class DatabaseTest {
  public static void write(
      Object obj, PreparedStatement ps, int parameterIndex)
      throws SQLException, IOException {
    ByteArrayOutputStream baos = new ByteArrayOutputStream();
    ObjectOutputStream oout = new ObjectOutputStream(baos);
    oout.writeObject(obj);
    oout.close();
    // This will NOT work in JDBC-ODBC bridge under JDK 1.2.2
    // as soon as the size of the byte array is bigger than 2000
    ps.setBytes(parameterIndex, baos.toByteArray());
  }
  public static Object read(ResultSet rs, String column)
      throws SQLException, IOException, ClassNotFoundException {
    // This will NOT work in JDBC-ODBC bridge under JDK 1.2.2
    // as a SQL NULL data value is not handled correctly.
    byte[] buf = rs.getBytes(column);
    if (buf != null) {
      ObjectInputStream objectIn = new ObjectInputStream(
        new ByteArrayInputStream(buf));
      return objectIn.readObject();
    }
    return null;
  }
  public static void main(String[] args) throws Exception {
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    Connection con = DriverManager.getConnection(
      "jdbc:odbc:MailingList", "sa", "");

    Statement st = con.createStatement();
    st.executeUpdate("INSERT BlobTable (Data) VALUES (NULL)");
    st.close();

    PreparedStatement ps = con.prepareStatement(
      "INSERT INTO BlobTable (Data) VALUES (?)");
    write(new Vector(2) {{ add("Hello"); add("World");}}, ps, 1);
```

```java
      ps.execute();
      Vector veryBig = new Vector(10);
      for (int i=0; i<10; i++) veryBig.add(new byte[10000]);
      write(veryBig, ps, 1);
      ps.execute();
      write("What Gives?", ps, 1);
      ps.execute();
      write(null, ps, 1);
      ps.execute();
      ps.close();

      st = con.createStatement();
      ResultSet rs = st.executeQuery("SELECT Data FROM BlobTable");
      while(rs.next()) {
        System.out.println(read(rs, "Data"));
      }
      rs.close();
      st.close();
    }
  }
```

To test this code, you'll have to set up a DSN to point to your MSSQL database (or a real database), and set up a table in your database called BlobTable with an IMAGE type for the column called "Data". In my example, I am writing a SQL NULL value into the table, then a small Vector (using dynamic aggregate initialization described in Newsletter 002) and after that a very big Vector, followed by a String and a Java null object. Lastly, I close the PreparedStatement, and read all the objects back and print them to the screen.

The output on my system is:

```
null
[Hello, World]
[[B@2f0db, [B@12d342, [B@6b97fd, [B@478e57, [B@5224ee, [B@76a746,
[B@5ff48b, [B@2ffc70, [B@663e3d, [B@4901] What Gives?
null
```

All very simple, except that in a real environment we don't really want to use IMAGE types in our database, unless absolutely necessary. We would prefer using VARBINARY as they take less space and form part of the block of data in the row. How do we know the size of a Java object in memory? I've worked out some formulae for "guessing" the size of a Java object by looking at its data members, which is a topic for another newsletter. In the meantime, the easiest is to take the biggest set of data that must be supported by your object and simply try out how much space it will take. There is no "sizeof()" method in Java to do that for you, unfortunately.

Some other issues related to the above problem is that a PreparedStatement is bound to a Connection, which means that we cannot rebind it to another Connection. The result is that we cannot use Connection pooling very well for this, as we then have to construct a new PreparedStatement each time we want to insert a Java object into the database, which defeats the reason for having a PreparedStatement. In the JDBC 3.0 specification, there is support for rebinding PreparedStatements, but we'll have to wait a bit before we'll have JDBC 3.0 compliant drivers.

How do you do this with JDK 1.2.x? You have to use the PreparedStatement.setBinaryStream(...) method to write the binary stream directly into the database. The problem is that when you read the binary stream again, and the database contained a SQL NULL, you get some nasty exceptions that are hard to figure out. Obviously, the JDK 1.3.x ODBC bridge throws different exceptions, so I don't advocate using the JDK 1.2.x bridge for writing objects into databases.

---
*Warning Advanced:*

*A memory leak, discovered by the best and most ruthless Java bug finder I know, occurs when you call the ResultSet.getTimestamp() method. Rather call ResultSet.getString() and convert the resultant String to a Date object. He's recently been granted permanent residence in our beautiful country of South Africa, which is quite a feat, considering the tough immigration laws. Congratulations!*
*---*

Please send me your comments / experiences on this. I always appreciate any feedback, both positive and negative, and please remember to send this newsletter to others who might be interested in Java.

Heinz

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Setting focus to second component of modal dialog

**Author: Dr. Heinz M. Kabutz**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 12th issue of "The Java(tm) Specialists' Newsletter". Please forward this free newsletter to as many people as you know who might be interested in "advanced" Java topics. You are welcome to send me questions on topics in my newsletters, I will do my best to answer them.

The code in these newsletters has been tested using JDK 1.3.

# Setting focus to second component of modal dialog

A few weeks ago I got stumped by a seemingly simple problem. I was trying to write a login dialog that would remember the last username entered and put the focus on the password field if an old username was found. I battled against the tide of Swing, even posted a question to the local Java User Group mailing list, but eventually I performed some obscure tricks to conquer this basic beginner's problem.

---
*Warning Advanced:*
*A problem with dialogs is that they are very often not bound to a parent frame, especially modal dialogs. This is not very good, because if you move to another application and move back to your Java application via the task bar in Windows, you cannot see the dialog. This single "bug" has caused a lot of confusion for users who think their Java application has hung up, but if they ALT+TAB to the application they can see the dialog again. A good solution is to create a frame at position -1000, -1000 and use that as the owner if the dialog does not have an owner. It is also possible to write a class which works out when a new window is shown and maps the title to the frame. This way you can find existing frames given a title. No, I won't tell you in this newsletter how to do that, no space.*
---

My LoginDialog looked something like this:

```java
//: LoginDialog.java
import javax.swing.*;
import java.awt.*;
public class LoginDialog extends JDialog {
  private final JTextField userName = new JTextField(8);
```

```
      private final JPasswordField password = new JPasswordField(8);
      public LoginDialog(Frame owner) {
        super(owner, "Login Dialog", true);
        getContentPane().setLayout(new GridLayout(0,2,5,5));
        getContentPane().add(new JLabel("Username:"));
        getContentPane().add(userName);
        getContentPane().add(new JLabel("Password:"));
        getContentPane().add(password);
        pack();
        Windows.centerOnScreen(this);
        show();
      }
      public String getUserName() { return userName.getText(); }
      public String getPassword() { return password.getText(); }
      public static void main(String[] args) {
        JFrame owner = new JFrame("Login Dialog");
        owner.setLocation(-1000, -1000);
        owner.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        owner.show();
        new LoginDialog(owner);
      }
    }
    //: Windows.java
    import java.awt.*;
    public class Windows {
      public static void centerOnScreen(Window window) {
        Dimension d = Toolkit.getDefaultToolkit().getScreenSize();
        window.setLocation(
          (d.width - window.getSize().width) / 2,
          (d.height - window.getSize().height) / 2);
      }
    }
```

As mentioned before, I wanted my focus to start on the password field, rather than the user name field. So, the obvious place to set the focus is after the call to "centerOnScreen", i.e. change the code to

```
// ...
  pack();
  centerOnScreen(this);
  password.requestFocus();
  show();
}
// ...
```

Unfortunately, you can only change the focus to components which are visible on the screen, and since the dialog has not been shown yet, trying to set the focus has no effect.

The obvious solution to this problem is to request the focus after the show() has been called. But, since this is a modal dialog, show will only return once the dialog has been closed, so even though the component is now visible, we will only request focus once we have closed the dialog, which does not help us awefully much.

Again, the seemingly obvious solution to this problem is to call the requestFocus method using SwingUtilities.invokeLater(), but you are not guaranteed that the dialog will then be visible, and if it is not, you again have no effect. You could of course wait for 10 seconds and then request focus, but that would result in a rather awkward user interface.

I posted this problem to a local Java user group and got one response to how this could be solved. But first I will show you my solution, which is terribly obscure, but I could not come up with anything better. Please send me your solutions if they differ from these:

## Solutions 1

We want to pass the focus on as soon as we get the focus in the username field. We thus add a focus listener to the

userName field, which transfers the focus to the next component when the focusGained method is called. We only want to do that when the dialog is constructed, so when the focusLost method is called we remove the listener again. LoginDialog would now look like this:

```java
//: LoginDialog2.java
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class LoginDialog2 extends JDialog {
  private final JTextField userName = new JTextField(8);
  private final JPasswordField password = new JPasswordField(8);
  public LoginDialog2(Frame owner) {
    super(owner, "Login Dialog", true);
    getContentPane().setLayout(new GridLayout(0,2,5,5));
    getContentPane().add(new JLabel("Username:"));
    getContentPane().add(userName);
    getContentPane().add(new JLabel("Password:"));
    getContentPane().add(password);
    pack();
    Windows.centerOnScreen(this);
    userName.addFocusListener(new FocusListener() {
      public void focusGained(FocusEvent e) {
        userName.transferFocus();
      }
      public void focusLost(FocusEvent e) {
        userName.removeFocusListener(this); // refers to listener
      }
    });
    show();
  }
  public String getUserName() { return userName.getText(); }
  public String getPassword() { return password.getText(); }
  public static void main(String[] args) {
    JFrame owner = new JFrame("Login Dialog");
    owner.setLocation(-1000, -1000);
    owner.show();
    owner.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    new LoginDialog2(owner);
  }
}
```

Yes, it is fairly obscure, but so is solution # 2, given to me by my "Bruce Eckel Handson" student, Charl Smit from CCH in South Africa. Thanks Charl.

## Solution 2

What we can also do is issue a focus gained event for the password field which will be actualised once the event queue gets a chance.

```java
//: LoginDialog3.java
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
public class LoginDialog3 extends JDialog {
  private final JTextField userName = new JTextField(8);
  private final JPasswordField password = new JPasswordField(8);
  public LoginDialog3(Frame owner) {
    super(owner, "Login Dialog", true);
    getContentPane().setLayout(new GridLayout(0,2,5,5));
    getContentPane().add(new JLabel("Username:"));
```

```java
      getContentPane().add(userName);
      getContentPane().add(new JLabel("Password:"));
      getContentPane().add(password);
      pack();
      Windows.centerOnScreen(this);
      changeFocus(userName, password);
      show();
    }
    private void changeFocus(final Component source,
        final Component target) {
      SwingUtilities.invokeLater(new Runnable() {
        public void run() {
          target.dispatchEvent(
            new FocusEvent(source, FocusEvent.FOCUS_GAINED));
        }
      });
    }
    public String getUserName() { return userName.getText(); }
    public String getPassword() { return password.getText(); }
    public static void main(String[] args) {
      JFrame owner = new JFrame("Login Dialog");
      owner.setLocation(-1000, -1000);
      owner.show();
      owner.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      new LoginDialog3(owner);
    }
  }
```

This also works perfectly, but I cannot decide which is more obscure. I suppose the 2nd solution is "better" because we can move the focus changing code out of the class into a general GUI utilities class and do this type of focus changing in a consistent way throughout the project. Also, it is probably easier with the 2nd solution to hop to any component on the screen, rather than just transfer the focus to the next component.

You be the judge. Please let me know if you have a better solution to this problem, by sending email to h.kabutz@computer.org.

Until next week, when I will look at what happens when you send GUI components over the network, ideas sponsored by Niko Brummer.

Heinz

---

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📋 Switching on Object Handles Part I

### Author: Dr. Heinz M. Kabutz

### Copyright 2001 Maxkab Solutions CC - South Africa
### Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 17th issue of "The Java(tm) Specialists' Newsletter", after a few very busy days in the Mauritian Paradise. Mauritius is a wonderful place to go to, with extremely friendly people all around, treating you like kings even in business. It's definitely worth a vacation, I wish I had gotten one while I was there ;-)

Please remember to forward this newsletter to anyone who might be interested, friends and foe.

## Simulating Switch Statements on Handles with try-catch

This week I will talk about a really completely useless idea, how to use switch/case statements on type-safe constants in Java. This idea occurred to me while I was talking to a bunch of programmers about TCP/IP programming, I don't think my topic of conversation had anything to do with the hare-brained idea presented in this newsletter. My listeners saw my eyes take on a distant gaze and I muttered "hmmm, I wonder if..." so here you go.

First I need to bore you with a monologe of why switch statements are bad and why you should never use them. Switch statements herald from a time before we used Object Orientation, Encapsulation and Polymorphism, and were mostly used to write methods which would do different things depending on the type of record we had passed the method. For example, say we had a CAD system, with a triangle, rectangle and circle, we could say:

```java
public interface Constants {
    int TRIANGLE_SHAPE = 0;
    int RECTANGLE_SHAPE = 1;
    int CIRCLE_SHAPE = 2;
}
```

Without encapsulation, we would then have a struct or class without methods, looking like this:

```java
public class Shape {
  public int type;
  public java.awt.Color color;
}
```

We would then have a CAD system for drawing these shapes, such as

```java
public class CADSystem implements Constants {
  public void draw(Shape shape) {
    switch(shape.type) {
      case TRIANGLE_SHAPE: // some code which draws a triangle
        System.out.println("Triangle with color " + shape.color);
        break;
      case RECTANGLE_SHAPE: // some code which draws a rectangle
        System.out.println("Rectangle with color " +shape.color);
        break;
      case CIRCLE_SHAPE: // some code which draws a circle
        System.out.println("Circle with color " + shape.color);
        break;
      default: // error only found at runtime
        throw new IllegalArgumentException(
          "Shape has illegal type " + shape.type);
    }
  }
}
```

This was the old procedural way of writing such code. The result was code where it was extremely challenging to add new types. In addition, in Java such code is very dangerous because we don't have enumerated types and you cannot switch on object references (well, have a look further down on how you actually "can"). You therefore could not be sure at compile time if you had defined the method for all the various types in your CADSystem.

The answer in OO is to use inheritance, polymorphism and encapsulation, the above example would thus be written as:

```java
public abstract class Shape {
  private final java.awt.Color color;
  protected Shape(java.awt.Color color) { this.color = color; }
  public java.awt.Color getColor() { return color; }
  public abstract void draw();
}

public class Triangle extends Shape {
  public Triangle(java.awt.Color color) { super(color); }
  public void draw() {
    System.out.println("Triangle with color " + getColor());
  }
}

public class Rectangle extends Shape {
  public Rectangle(java.awt.Color color) { super(color); }
  public void draw() {
    System.out.println("Rectangle with color " + getColor());
  }
}
```

```java
public class Circle extends Shape {
  public Circle(java.awt.Color color) { super(color); }
  public void draw() {
    System.out.println("Circle with color " + getColor());
  }
}

public class CADSystem {
  public void draw(Shape shape) {
    shape.draw();
  }
}
```

Now if we forget to implement one of the draw methods, we'll immediately get a compile-time error. Of course, if we extend Rectangle and forget to implement the draw method we'll get the wrong shape, so a certain level of diligence in testing is still required.

It is possible to take a switch statements and transform it to polymorphism using various refactorings. In previous newsletters I mentioned the book "Refactoring" by Martin Fowler. In case you hadn't noticed, I am a fan (of the book, that is). In that book you can find refactorings to transform a switch/case statement to polymorphism or polymorphism back to a switch/case statement.

So, in the unfortunate case (haha, pun intended) that you want to use a switch-type of construct but you don't want to worry about the anonymity of using int's as type identifiers, how do you do it?

We demonstrate by using a TransactionType class which defines the transaction isolation levels you find in most enterprise systems. The isolation types are None, Read Uncommitted, Read Committed, Repeatable Read and Serializable. The point of this newsletter is not to describe transaction isolations, so I won't go into what they all mean. Rumour has it though, that if you use them without knowing what they mean, you will get a system which doesn't work, HA.

We define a TransactionType superclass with a private constructor, so that it is not possible to construct instances of these types or to subclass it, except from within the type. The constructor takes a name as a description, which can be returned via the toString() method. The reason why the type class has to be Throwable will become clear in the example.

We then make public static inner classes for each of the types, again with private constructors, and make public static final instances of these types in each of the inner classes. The reason why we need classes and instances will also become clearer in the example.

```java
//: TransactionType.java
public class TransactionType extends Throwable {
  private final String name;
  private TransactionType(String name) {
    this.name = name;
  }
  public String toString() { return name; }
  public static class None extends TransactionType {
    public static final TransactionType type = new None();
    private None() { super("None"); }
  }
  public static class ReadUncommitted extends TransactionType {
    public static final TransactionType type =
      new ReadUncommitted();
    private ReadUncommitted() { super("ReadUncommitted"); }
  }
  public static class ReadCommitted extends TransactionType {
    public static final TransactionType type =
      new ReadCommitted();
```

```java
      private ReadCommitted() { super("ReadCommitted"); }
  }
  public static class RepeatableRead extends TransactionType {
    public static final TransactionType type =
      new RepeatableRead();
    private RepeatableRead() { super("RepeatableRead"); }
  }
  public static class Serializable extends TransactionType {
    public static final TransactionType type =
      new Serializable();
    private Serializable() { super("Serializable"); }
  }
}
```

How does such a type help us to make safe types which we can switch on? The answer is that we use a construct which is not really meant to be used as a switch, but which acts as one nevertheless, namely the throw-catch construct. We simply throw the type, which can be a handle to a TransactionType object, and the exception handling mechanism sorts out which catch to call. Yes, I can hear you all groaning now with pearls of sweat caused by fear, but this really does work. For syntactic sugar, we can import the inner classes using "import TransactionType.*" after which we can refer to the inner class simply by their name "ReadCommitted". We can of course also use the full name such as Transaction.None instead of importing the inner classes.

```java
  //: SwitchingOnObjects.java
  import TransactionType.*;
  public class SwitchingOnObjects {
    public static void switchStatement(TransactionType transact) {
      try {
        throw transact;
      } catch(TransactionType.None type) {
        System.out.println("Case None received");
      } catch(ReadUncommitted type) {
        System.out.println("Case Read Uncommitted");
      } catch(ReadCommitted type) {
        System.out.println("Case Read Committed");
      } catch(RepeatableRead type) {
        System.out.println("Case Repeatable Read");
      } catch(TransactionType type) {
        System.out.println("Default");
      }
    }
    public static void main(String[] args) {
      switchStatement(TransactionType.None.type);
      switchStatement(ReadUncommitted.type);
      switchStatement(ReadCommitted.type);
      switchStatement(RepeatableRead.type);
      switchStatement(Serializable.type);
    }
  }
```

Try it out, the exception handling mechanism works quite well for this. There are a few pointers you have to follow if you want to use this:

1. Don't ever catch "Throwable" as the default case. You should rather catch the type base class, such as TransactionType. Otherwise you run the risk of catching RuntimeException and Error classes, such as OutOfMemoryError.

2. Make sure that ALL the types are public inner classes of the type base class.

3. Make sure that all the constructors are private.

4. Lastly, rather use polymorphism to achieve this effect. Switch/Case code is really messy to maintain and very error- prone.

I've already donned my asbethos suit for the criticisms from hard-nosed C/C++ programmers who think switch/case is great and from weeny Java purists who think that switch/case is completely unacceptable. Flame away...

Until next week, and please remember to forward this newsletter in its entirety to as many Java users as you know.

Heinz

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

   **Switching on Object Handles Part II**

## Author: Dr. Heinz M. Kabutz

## Copyright 2001 Maxkab Solutions CC - South Africa
## Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

As expected, I received interesting comments as a result of my last newsletter, and more is sure to follow after the long weekend.

---

Here some comments from someone at EnterpriseDeveloper, makers of the excellent JCertify product for practicing for the SUN Certified Java Programmer Examination:

OK abestos suit man,

I'm more concerned about the encouragement of using exceptions to modify/implement regular program flow.

I just picked up some code from another group where they were doing Exceptions to break out of Whiles, For's, thowing exceptions as a kind of break, ... Not only is it a performance hit, it makes for really "nasty" code.

Perhaps you can put a summary statement on your next issue reminding people that Exceptions are not meant to be used as Loop control / Loop Control flow enhancement. I could see someone putting your switch idea inside a BIG For loop - say a list of 1,000 items, and causing 1,000 exceptions.

Very interesting reading - I always like "pushing the envelope", I'm just somewhat concerned about the influence to those "more dangerous" with such an idea. There is an interpretation among those newer to Java that Exception Handling is good for general Program Flow enhancement ( Loop breaking, break statement replacement, throwing Exceptions within a repeating loop,

Exceptions are very much abused by those coming from the C world, or non OO people coming from C++

## My response:

I did some tests and found that the performance wasn't that bad, it was actually better than a list of if-instanceof-elses. Note that I am specifically not constructing a new exception, and the construction of exceptions is what takes the time, as far as I know.

Throwing exceptions to change program flow is VERY bad, I agree 100%. I am intending on sending a follow-up to yesterdays newsletter in which I warn again that you shouldn't do this. Perhaps one of the "dangerous" will say: but I've been doing that for a long time...

## His response:

There is some degradation to being in a Try / Catch block, but nothing compared to the creation of an Exception. What do you mean by "Not that Bad" - less than 5% degradation ( tight loop )?

*YOU* may not be creating exceptions when using this, and you are probably interpreting it as "I am only condoning the use of this particular set of lines of code". It may encourage *OTHERS* to continue, or even expand, bad Exception Handling practices. They won't know about the performance cost, and they won't understand this could get "nasty" really fast by extending this concept to other programming practices ( and adding REAL Exceptions ).

But I still like the "pushing the envelope" aspect of that code.

---

In addition, I've included code for a TransactionType class that has a type id int which can be used in a switch statement. This is not as safe, but "cleaner" to use instead of abusing try. I also included a test for multiple if-instanceof-else which turned out slower, and lastly an implementation of using the strategy pattern, which was the fastest of all.

```java
//: TransactionType.java
public class TransactionType extends Throwable {
  private static int counter = 0;
  private final int id = counter++;
  public final int getId() { return id; }
  private final String name;
  private TransactionType(String name) {
    this.name = name;
  }
  public String toString() { return name; }
  public static class None extends TransactionType {
    public static final TransactionType type = new None();
    private None() { super("None"); }
  }
  public static class ReadUncommitted extends TransactionType {
    public static final TransactionType type =
      new ReadUncommitted();
    private ReadUncommitted() { super("ReadUncommitted"); }
  }
  public static class ReadCommitted extends TransactionType {
    public static final TransactionType type =
      new ReadCommitted();
    private ReadCommitted() { super("ReadCommitted"); }
  }
  public static class RepeatableRead extends TransactionType {
    public static final TransactionType type =
      new RepeatableRead();
    private RepeatableRead() { super("RepeatableRead"); }
  }
  public static class Serializable extends TransactionType {
    public static final TransactionType type =
      new Serializable();
    private Serializable() { super("Serializable"); }
  }
}
```

Now we have a performance test which investigates the various options. Fortunately, the fastest was using Polymorphism and the Strategy design pattern, as I had hoped.

```java
//: Performance.java
import TransactionType.*;
```

```java
public class Performance {
  public static void switchOnObject(TransactionType transact) {
    try {
      throw transact;
    } catch(TransactionType.None type) {
    } catch(ReadUncommitted type) {
    } catch(ReadCommitted type) {
    } catch(RepeatableRead type) {
    } catch(TransactionType type) {
    }
  }
  public static void switchOnInt(TransactionType transact) {
    switch(transact.getId()) {
    case 0: break;
    case 1: break;
    case 2: break;
    case 3: break;
    case 4: break;
    default: break;
    }
  }
  public static void switchInstanceof(TransactionType transact) {
    if (transact instanceof None) {
    } else if (transact instanceof ReadUncommitted) {
    } else if (transact instanceof ReadCommitted) {
    } else if (transact instanceof RepeatableRead) {
    }
  }
  public static abstract class TransactionTypeStrategy {
    public abstract void doSomething();
  }
  public static class RepeatableReadStrategy extends
      TransactionTypeStrategy {
    public void doSomething() {}
  }
  public static void main(String[] args) {
    long time = -System.currentTimeMillis();
    for (int i=0; i<1000000; i++) {
      switchOnObject(RepeatableRead.type);
    }
    time += System.currentTimeMillis();
    System.out.println("Switching 1000000 times on objects " +
      time + "ms");

    time = -System.currentTimeMillis();
    for (int i=0; i<1000000; i++) {
      switchOnInt(RepeatableRead.type);
    }
    time += System.currentTimeMillis();
    System.out.println("Switching 1000000 times on ints " +
      time + "ms");
```

```
      time = -System.currentTimeMillis();
      for (int i=0; i<1000000; i++) {
        switchInstanceof(RepeatableRead.type);
      }
      time += System.currentTimeMillis();
      System.out.println("Switching 1000000 times using " +
        "instanceofs " + time + "ms");

      time = -System.currentTimeMillis();
      TransactionTypeStrategy strategy =
        new RepeatableReadStrategy();
      for (int i=0; i<1000000; i++) {
        strategy.doSomething();
      }
      time += System.currentTimeMillis();
      System.out.println("Switching 1000000 times using " +
        "Strategy Polymorphism pattern " + time + "ms");
    }
  }
```

Result on my notebook:

```
  Switching 1000000 times on objects 481ms
  Switching 1000000 times on ints 40ms
  Switching 1000000 times using instanceofs 1012ms
  Switching 1000000 times using Strategy Polymorphism pattern 30ms
```

So, for those where it's not clear yet, please don't use exceptions for ANYTHING except error handling. Don't use them to jump out of loops, do switch statements, or find out what methods you were called from.

Incidentally, in one of my first newsletters (004), I mentioned that I would like to be able to get a stack trace without having to construct an exception. So far, no-one has come up with a solution, but I've found a way which *might* work using JNI.

Regards, until next week...

Heinz

Receive our  newsletter with new tips!  More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## 📧 Socket Wheel to handle many clients

**Author: Dr. Heinz M. Kabutz**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 23rd issue of "The Java(tm) Specialists' Newsletter", where I try and get back to my roots of distributed performance evaluation. My PhD thesis was entitled "Analytical Performance Evaluation of Concurrent Communicating Systems using SDL and Stochastic Petri Nets", or something like that. The main idea was to automatically map protocols designed in the Specification and Description Language (SDL) to a modelling language called Stochastic Petri Nets for the simple reason that there are a lot of well known analytical techniques available for evaluating a Stochastic Petri Net's performance. But, all that was of another era when it did not matter that it would take a very long time to analyse a protocol of any real size. In fact, that was last millenium, i.e. long long ago, i.e. don't bother asking me any questions about it ;-)

When we measure performance we have to consider mainly two criteria: memory and cpu cycles. I was able to significantly reduce the amount of memory needed for the server by the idea presented here, but I did not manage to increase the speed at which clients are serviced, although it converged. If you think of anything that would improve the speed of the SocketWheel, please let me know, and you will earn instant fame in over 40 countries by being immortalized in my next newsletter.

Please forward this newsletter to as many people as you know who are interested in programming in Java at more-than-entry-level.

## Socket Wheel to handle many clients

The typical way of implementing a server that needs to "talk back" to the client is to construct a thread for each client that is connected, normally through a thread pool. For example, consider the Server.java file:

```java
// Server.java
import java.net.*;
import java.io.*;
public class Server {
  public static final int PORT = 4444;
```

```java
    public Server(int port) throws IOException {
      ServerSocket ss = new ServerSocket(port);
      while(true) {
        new ServerThread(ss.accept());
      }
    }
    private class ServerThread extends Thread {
      private final Socket socket;
      public ServerThread(Socket socket) {
        this.socket = socket;
        start();
      }
      public void run() {
        try {
          ObjectOutputStream out = new ObjectOutputStream(
            socket.getOutputStream());
          ObjectInputStream in = new ObjectInputStream(
            socket.getInputStream());
          while(true) {
            in.readObject();
            out.writeObject(new String("test"));
            out.flush();
            out.reset();
          }
        } catch(Throwable t) {
          System.out.println("Caught " + t + " - closing thread");
        }
      }
    }
    public static void main(String[] args) throws IOException {
      new Server(PORT);
    }
  }
```

What this does is simply read an object and write an object for the duration of the client being connected to the thread. When the client disconnects, the thread will stop. The code is not very "clean", I should handle closing of Sockets better than it is, but I don't want to cloud over the issue at stake here.

A client would typically look like this (send an object, read an object, wait some time, etc.):

```java
// Client.java
import java.net.*;
import java.io.*;
public class Client {
  public Client(int port) throws Exception {
    Socket socket = new Socket("localhost", port);
    ObjectOutputStream out = new ObjectOutputStream(
      socket.getOutputStream());
    ObjectInputStream in = new ObjectInputStream(
      socket.getInputStream());
    for (int i=0; i<10; i++) {
      out.writeObject(new Integer(i));
      out.flush();
      out.reset();
```

```
        System.out.println(in.readObject());
        Thread.sleep(1000);
      }
    }
    public static void main(String[] args) throws Exception {
      new Client(Server.PORT);
    }
  }
```

In order to test what happens when a lot of clients connect I wrote a MultiClient class, which constructs 3500 sockets and an equivalent number of object output and input streams to use for sending messages. It then cycles through the sockets and writes to them one object, then cycles through them again and reads one object. The reason I took 3500 sockets is that on my little notebook I could not open more than 3500 sockets, don't ask why, I don't know. That number is *probably* system dependent, so if you get an exception when trying to create a socket, try what happens when you have less sockets.

```
// MultiClient.java
public class MultiClient {
  public MultiClient(int port) throws Exception {
    long time = -System.currentTimeMillis();
    Socket[] sockets = new Socket[3500];
    ObjectOutputStream[] outs =
      new ObjectOutputStream[sockets.length];
    ObjectInputStream[] ins =
      new ObjectInputStream[sockets.length];
    for (int i=0; i<sockets.length; i++) {
      sockets[i] = new Socket("localhost", port);
      outs[i] = new ObjectOutputStream(
        sockets[i].getOutputStream());
      ins[i] = new ObjectInputStream(
        sockets[i].getInputStream());
    }
    System.out.println("Constructed all sockets");
    for (int j=0; j<32; j++) {
      long iterationTime = -System.currentTimeMillis();
      for (int i=0; i<sockets.length; i++) {
        outs[i].writeObject(new Integer(i));
        outs[i].flush();
        outs[i].reset();
      }
      System.out.println(j + ": Written to all sockets");
      for (int i=0; i<sockets.length; i++) {
        ins[i].readObject();
      }
      System.out.println(j + ": Read from all sockets");
      iterationTime += System.currentTimeMillis();
      System.out.println(j + ": Iteration took " +
        iterationTime + "ms");
    }
    time += System.currentTimeMillis();
    System.out.println("Writing to " + sockets.length +
      " sockets 32 times took " + time + "ms");
  }
  public static void main(String[] args) throws Exception {
```

```
        new MultiClient(Server.PORT);
    }
  }
```

This all works quite nicely, except that each thread in the JDK1.3 implementation of the VM takes up 20KB for its stack. When you add up all the other memory taken up for streams and sockets, it comes to 97MB used up on the server, just to handle a paltry 3500 clients! So, if we had 35000 clients connecting, with the presumption that our machine can handle that many sockets, we not only create 35000 threads (which each don't do that much, but it still takes a long time to construct them all, even if you use a thread pool) but we also gobble up almost 1GB of memory!!!

I was pondering this problem a few days ago and came up with an idea to use polling (yes, I know, polling sucks) to avoid making a thread for each client. Instead of having a server which uses a thread for each client that gets connected, we have a list of sockets that are connected to the server. Writing to a socket will block if the TCP buffer is full, which is about 64000 bytes, so we could have the problem of the server being hung up completely if the client decides to not service the stream. I will conveniently ignore that problem in this newsletter.

We want to have an ObjectOutputStream and an ObjectInputStream associated with each Socket, so we make an inner class to contain those values, which we call a SocketBucket. To make connecting fast, we keep two lists of SocketBuckets, one for the new sockets and one for the already connected sockets. We then run through all the sockets and try to read from each of them with a timeout of 1 millisecond. If there is nothing to read we get an InterruptedIOException and go to the next socket.

If we found at least one socket that had some data, we immediately go looking again, otherwise we go dream for a while and then go looking for more data. There are many different ways in which we could tune this approach, for example, you could keep a set of the last sockets which had data and push their priority up or down, depending on stochastic prediction techniques (not that I remember what that means - just sounded cool!). One of the disadvantages with this approach is also that the server has to wait for an entire millisecond before looking at the next socket. It would be much better to wait less, otherwise if you have 1000 sockets connected, it will take 1 second just to check if any of the sockets have data waiting. Unfortunately, 1 ms is the shortest that we can wait with Java sockets.

Here is the code for the SocketWheel:

```java
// SocketWheel.java
import java.net.*;
import java.io.*;
import java.util.*;
public class SocketWheel {
  // the list contains SocketBuckets
  private final List sockets = new LinkedList();
  // we don't want to block a new connection while we are busy
  // serving the existing ones
  private final List newSockets = new LinkedList();
  public SocketWheel() {
    new ServerThread();
  }
  public void addSocket(Socket socket) throws IOException {
    synchronized(newSockets) {
      newSockets.add(new SocketBucket(socket));
      newSockets.notify();
    }
  }
  private class SocketBucket {
    public final Socket socket;
    public final ObjectOutputStream out;
    public final ObjectInputStream in;
    public SocketBucket(Socket socket) throws IOException {
```

```java
        this.socket = socket;
        out = new ObjectOutputStream(socket.getOutputStream());
        in = new ObjectInputStream(socket.getInputStream());
        socket.setSoTimeout(1);  // VERY short timeout
      }
    }
    private class ServerThread extends Thread {
      public ServerThread() {
        super("ServerThread");
        start();
      }
      public void run() {
        long dreamTime = 10;
        boolean foundSomething;
        while(true) {
          try {
            synchronized(newSockets) {
              sockets.addAll(newSockets);
              newSockets.clear();
            }
            foundSomething = false;
            Iterator it = sockets.iterator();
            while(it.hasNext()) {
              SocketBucket bucket = (SocketBucket)it.next();
              try {
                bucket.in.readObject();
                foundSomething = true;
                bucket.out.writeObject(new String("test"));
                bucket.out.flush();
                bucket.out.reset();
              } catch(InterruptedIOException ex) {
                // just skip this socket
              } catch(IOException ex) {
                it.remove();
              }
            }
            if (foundSomething) {
              dreamTime = 6;
            } else {
              if (dreamTime < 1000)
                dreamTime *= 1.5;
              else dreamTime = 1000;
              synchronized(newSockets) {
                // only sleep if we didn't find anything
                newSockets.wait(dreamTime);
              }
            }
          } catch(Throwable t) {
            System.out.println("Caught " + t + " - remove socket");
          }
        }
      }
```

```
      }
   public static void main(String[] args) throws IOException {
      SocketWheel wheel = new SocketWheel();
      ServerSocket ss = new ServerSocket(Server.PORT);
      while(true) {
        Socket socket = ss.accept();
        wheel.addSocket(socket);
      }
   }
  }
```

When I connect to the SocketWheel server with the MultiClient, the server uses up only 32MB of RAM, basically one third of the other server, but it is a little bit slower to use this approach as opposed to threading, and it is a lot more complicated. In addition, the whole example sometimes gets stuck, I don't know why. If the MultiClient stops proceeding and the CPU goes to 0%, you'll have to restart the MultiClient. (If you spot the problem, please let me know, I suspect it's an underlying C implementation problem which is why I'm not pursuing it.)

With the SocketWheel, the test took 2:36 minutes to complete, the normal threaded Server took only 2:23 minutes to complete, CPU was at 100% both times, disk usage was 0%. The difference in speed is not that great, whereas the memory usage is only 34MB in the SocketWheel server, i.e. roughly 1/3 of the threaded server.

When we change the MultiClient to only use 350 sockets, the SocketWheel takes 14 seconds, the threaded server only 11 seconds, the SocketWheel uses 9.4MB, the threaded server 14MB, a smaller difference, probably because that includes the total memory used by java.exe measured with the task manager, i.e. it includes the memory used by the JVM.

It was quite fun writing this SocketWheel, and was actually a lot easier than I thought possible. Please tell me if you've done something similar or if you think of ways to improve the speed of the SocketWheel server. I tried having a couple of threads in the SocketWheel, but the whole system just got stuck more often and did not improve speed. Please don't use the SocketWheel "as is" unless you're willing to discover and fix the bug that makes it get stuck and cater for clients not reading their sockets.

I always appreciate any feedback, both positive and negative, so please keep sending your ideas and suggestions. Please also remember to take the time to send this newsletter to others who are interested in Java.

Heinz

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

### The minimalist application server Or Classloading revisited
**Author: Dr. Christoph G. Jung, Sponsored by infor Business Solutions AG (http://www.infor.de)**

**Copyright 2001 Maxkab Solutions CC - South Africa**
**Reproduced as part of the JavaTips with permission.**

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 22nd issue of "The Java(tm) Specialists' Newsletter", sponsored by infor business solutions AG, an ERP company based in Germany. Many thanks go to Christoph Jung for again making an excellent contribution to "The Java(tm) Specialists' Newsletter" by taking the time to write about some of the advanced things he's busy with. This is the longest newsletter so far, and also the most in-depth. Don't start expecting the same quality when I write ;-), in fact, I might skip next week's letter to give you some time to digest this one.

As always, I would like to ask you to forward this newsletter to anybody you know who might be interested in Java. Please also remember that if you are a beginner in Java, some of these news- letters will go over your head. Don't give up!

## The minimalist application server Or Classloading revisited

A few weeks ago, I was in the middle of an ICQ chat with our admired Dr. Kabutz when he suddenly uttered a definite "hi, hi" which took me by surprise as it was different to the ubiquitous "emoticons" or smileys that I am used to. Nevertheless, I was immediately able to catch the thousands of denotations that this very special textual laughter was transporting. In this case, the semantics can be very well subsumed by the German term "verschmitzt" which is difficult to translate to English but would probably be a combination of "roguish", "twisty", "experienced" and "savvy". Similarly, "ha, ha" (a very open reaction to a joke) and "ho, ho" (kind of patronizing Santa Claus stance) are able to express much more context than any smiley construction I know of, at a minimum overhead of bandwidth. Measurements using WinZip have shown verbose laughter statistically being only 5% more expensive than its certainly poorer smiley counterpart.

And now ... I hope you kept a copy of newsletter 18 where Heinz illuminated us with the insight that class identity in the virtual machine is quite independent of even fully-qualified class names. The magical constructs that he used for his investigations were the java.lang.ClassLoader and suitable derivatives, such as the java.net.URLCLassloader. These are, amongst other innovations of the Java(tm) 2 Runtime, indeed a fascinating subject for experimentation.

## Classes and ClassLoaders

Perhaps you found that having several classes called "A" in your java.exe was not very useful. On the contrary, it probably required quite an effort to keep the various sources and class files of that newsletter in separate directories. However, I would like to demonstrate how this rather theoretical possibility has significant practical value when it comes to building real programs that are able to simultaneously host multiple Java applications. The most well known examples of such programs are found in the J2EE(tm) application servers.

For this purpose, let us briefly revisit what the head of Maximum Solutions Ltd., by the way, one of the best Java consultancies I know of ;-) [hk: hihi], has demonstrated to our surprised eyes and what may be already apparent from the terminology: A java.lang.ClassLoader is an object that is used to locate external resources (such as *.class file placed in a directory or zipped into a jar-file) and to import the contained byte code for constructing java.lang.Class instances at runtime.

Each java.lang.Class in the JVM is hence associated with its so-called "defining classloader" that has imported it and that can be obtained using the Class.getClassLoader() method. For example, regard the following classes stock.Warehouse (interface to a mono-product warehouse exposing a reservation method) and sales.Order (remotely connects/links to a warehouse to satisfy the order in quantities of size "bunch").

```
// stock/Warehouse.java
package stock;
/** Sample remote interface that is linked by sales.Order. */
public interface Warehouse extends java.rmi.Remote {
  /** Reserve items in this warehouse */
  void reserve(int quantity) throws java.rmi.RemoteException;
}


// sales/Order.java
package sales;
import java.rmi.*;
import stock.Warehouse;
/** Class that links in another jar */
public class Order {
  /** the stock.Warehouse which hosts our items */
  private final Warehouse wHouse;
  /** how much do we need? */
  private final int amount;
  /** quantize reservations */
  private static int bunch=5;
  /** constructs a new order */
  public Order(String itemName, int quantity)
      throws java.net.MalformedURLException, NotBoundException,
        RemoteException {
    // look into the rmi registry to locate the warehouse
    wHouse=(Warehouse)Naming.lookup("//localhost/"+itemName);
    this.amount=quantity;
  }
  /** method that delegates reservation to warehouse */
  public void reserve() throws RemoteException {
    for(int count = 0; count < amount; count += bunch)
      wHouse.reserve(bunch);
  }
}
```

Although sophisticated names and nifty logic should suggest an increasing level of practicability compared to Heinz's previous excursions, please remember that this is a technical newsletter and not a business logic

course. Hence, instead of flaming about the obvious flaws and, at the same time, selling the attached files as ready-made ERP-competitor to our infor:COM, compile the sources into two separate directories "classes/stock" and "classes/sales":

```
javac -d classes\stock\ stock\*.java
javac -classpath classes\stock\ -d classes\sales\ sales\*.java
```

```java
// server/Loader.java
package server;
import java.net.*;
import java.io.*;
public class Loader {
  /** Demonstration of some classloading issues */
  public static void main(String[] args) throws Exception {
    // construct a tiny classloader hierarchy
    ClassLoader stockLoader = new URLClassLoader(
      new URL[] {getStockLocation()});
    ClassLoader salesLoader = new URLClassLoader(
      new URL[] {getSalesLocation()}, stockLoader);
    // load order class
    Class orderC = salesLoader.loadClass("sales.Order");
    System.out.println(orderC + " loaded from " + salesLoader +
      "; defined by " + orderC.getClassLoader());
    // load warehouse class
    Class wHouseC = salesLoader.loadClass("stock.Warehouse");
    System.out.println(wHouseC + " loaded from " + salesLoader +
      "; defined by " + wHouseC.getClassLoader());
    // analyse class links
    System.out.println("loading and linking same " +
      wHouseC.equals(orderC.getDeclaredField("wHouse").getType()));
    System.exit(0);
  }
  /** where the stock classes can be found */
  protected static URL getStockLocation() throws IOException {
    return new File("classes/stock/").toURL();
  }
  /** where the sales classes can be found */
  protected static URL getSalesLocation() throws IOException {
    return new File("classes/sales/").toURL();
  }
}
```

Whenever the virtual machine (VM) gets hold of your just produced sales.Order - maybe using the loadClass("sales.Order") statement in the preceding server.Loader code - it will also automatically ask the defining classloader of sales.Order to additionally load the linked class into memory:

```
salesLoader.loadClass("stock.Warehouse").
```

Since we have placed the byte code of stock.Warehouse into the classes\stock directory that is not under the hood of salesLoader, salesLoader will delegate the call to its parent stockLoader and, upon successful resolution, the respective class-representations in the VM are linked together. These issues will become apparent if you run Loader from the "classes\server" folder (using the attached "lazy developer" policy file which you should never use in production environments, you have been warned!).

```
javac -d classes\server\ server\*.java
java -Djava.security.policy=server.policy -classpath
  classes\server server.Loader
```

```
// server.policy
grant {
  // Allow everything for now [hk: *ouch*]
  permission java.security.AllPermission;
};
```

The output will most likely look similarly to this:

```
class sales.Order loaded from java.net.URLClassLoader@253498;
    defined by java.net.URLClassLoader@253498
interface stock.Warehouse defined by java.net.URLClassLoader@209f4e
loading versus linking: true
```

Note that java.exe operates a bit lazily since salesLoader.loadClass("stock.Warehouse") and hence stockLoader.loadClass("stock.Warehouse") will not happen until you begin to inspect or instantiate the imported Order class! In our Loader code, this is implicitly triggered by the getDeclaredField("wHouse") call. This way, you will not end up loading every class file that is under control by any classloader into memory when you were just asking for Order. You will really appreciate this feature if you have experienced the class inflation phenomenon typical in an OO project ;-)

## Hot-Deploy

As an educated audience [hk: i.e. you read all my newsletter ;-}] you did know all this before of course. This loading & linking is certainly not restricted to our Order and Warehouse, but is used to intern every class into the VM. It even does this with the main class and the Java(tm) 2 runtime representations that the main class depends upon. The big difference to our example is, however, that you usually specify a single classpath/ classloader ("classes\server\") and a single main class ("server.Loader") at startup of java.exe, hence a single java "application".

With our freshly acquired knowledge, we have revealed the opportunity to dynamically start/shutdown applications at runtime which is often called "deploying": java.exe starts with a minimal setup, e.g., the below server.ApplicationServer, and is then equipped with additional application specifications on-the-fly. An application specification that is processed by the server's deploy(...) and start(...) methods consists of the path to the byte-code, the name of the main class, and the set of initial arguments.

```
// server/ApplicationServer.java
package server;
import java.net.*;
import java.util.*;
import java.io.*;
/** A tiny "application server" */
public class ApplicationServer extends Loader {
  /** this map stores application urls->active-threads */
  final protected Map applications = new HashMap();
  /** the deploy method interns an application
   * @param url jar where the application is packed
   * @param main fully-qualified name of Main class
   * @param args arguments to main method
   * @param parent parent classloader
   * @throws ClassNotFoundException if application not found
   * @return the classloader that has been generated for that app
   */
  public synchronized ClassLoader deploy(final URL url,
      String main, final String[] args, ClassLoader parent)
      throws ClassNotFoundException {

    System.out.println("Deploying "+url);
```

```
          // is this a redeploy?
          if(applications.containsKey(url)) // yes: tear it down first
            undeploy(url);

          // generate a new classloader
          final ClassLoader loader = constructClassLoader(
            new URL[] {url}, parent);
          // load the mainclass
          final Class mainClass = loader.loadClass(main);

          // construct a new "main" thread
          Thread newThread = new Thread() {
            public void run() {
              try{
                // run the main method with the given args
                Class[] params = {String[].class}; // args types
                mainClass.getMethod("main", params).invoke(
                  mainClass, new Object[]{args});
                // keep application alive until teared down
                synchronized(this) { this.wait(); }
              } catch(java.lang.reflect.InvocationTargetException e) {
                e.getTargetException().printStackTrace();
              } catch(Exception e) {
              } finally {
                // we lock the appServer
                synchronized(ApplicationServer.this) {
                  try{
                    // in case that any application error occurs
                    // (or the application is to be undeployed)
                    System.out.println("Stopping " + url);
                    // remove entry if still there
                    applications.values().remove(this);
                    // call cleanup method
                    mainClass.getMethod("stop", new Class[0]).invoke(
                      mainClass, new Object[0]);
                  } catch(Exception _e) {}
                } // synchronized(appServer.this)
              } // finally
            }}; // method run(); class Thread()

          // set the thread context
          newThread.setContextClassLoader(loader);
          // register application x thread
          applications.put(url, newThread);
          // return classloader
          return loader;
        } // method deploy()

        /** starts an application that has already been deployed */
        public synchronized void start(URL url) {
          System.out.println("Starting " + url);
          ((Thread)applications.get(url)).start();
```

```
    }
  /** Undeploys a running application. Never, I repeat, NEVER, do
   *  this using Thread.stop() but use the various options that
   *  are proposed by your JDK documentation to gracefully notify
   *  a thread of shutdown.
   * @param url url where the application is packed
   * @throws Exception if the app could not be teared down
   */
  public synchronized void undeploy(URL url) {
    // uh uh. bastard. But for Heinz newsletter, its ok ;-)
    // [hk: gee thanks]
    ((Thread) applications.get(url)).stop(new Exception("stop"));
  }
  /** class loader factory method */
  protected ClassLoader constructClassLoader(URL[] urls,
      ClassLoader parent) {
    return new URLClassLoader(urls,parent);
  }

  /** example usage of the appServer */
  public static void main(String[] args) throws Exception {
    BufferedReader stdin = new BufferedReader(
      new InputStreamReader(System.in));
    ApplicationServer appServer = new ApplicationServer();
    ClassLoader stockLoader = appServer.deploy(
      getStockLocation(), "stock.Main",
      new String[] {"screwdriver", "stock.WarehouseImpl",
        "screwdriver", "200"},
      null);
    appServer.start(getStockLocation());
    stdin.readLine();
    appServer.deploy(getSalesLocation(),
      "sales.Main",
      new String[] {"screwdriver","50"},
      stockLoader);
    appServer.start(getSalesLocation());
    stdin.readLine();
    appServer.deploy(appServer.getSalesLocation(),
      "sales.Main",
      new String[] {"screwdriver","80"},
      stockLoader);
    appServer.start(appServer.getSalesLocation());
    stdin.readLine();
    appServer.undeploy(appServer.getSalesLocation());
    appServer.undeploy(appServer.getStockLocation());
    System.exit(0);
  }
}
```

As example applications that are to be deployed by the presented server logic, we sketch below an exemplary batch sales.Main that constructs and processes a set of sales.Order. And we introduce an exemplary stock application stock.Main that exports remote warehouse services such as implemented by stock.WarehouseImpl.

```java
// sales/Main.java
package sales;
/** An example batch application */
public class Main   {
  /** starts the batch */
  public static void main(String[] args) throws Exception {
    // analyse command-line
    for(int count=0; count<args.length; count++)
      // construct order and reserve
      new Order(args[count++],
        new Integer(args[count]).intValue()).reserve();
  }
}

// stock/WarehouseImpl.java
package stock;
import java.rmi.server.UnicastRemoteObject;
/** Example implementation of a remote service */
public class WarehouseImpl extends UnicastRemoteObject
    implements Warehouse {
  /** number of stored items */
  protected int quantity;
  /** constructs warehouse */
  public WarehouseImpl(String itemName, int quantity)
      throws java.rmi.RemoteException {
    this.quantity=quantity;
  }
  /** reserves items*/
  public void reserve(int amount) {
      System.out.println(this + " is about to reserve " +
        amount + " items.");
      if(quantity < amount) empty(amount - quantity);
      quantity -= amount;
  }
  /** what to do if the warehouse is empty */
  protected void empty(int underLoad)  {
      throw new IllegalArgumentException("warehouse empty");
  }
}

// stock/Main.java
package stock;
import java.rmi.*;
import java.util.*;
/**
 * Example service-publishing application
 */
public class Main {
  /** the services provided by this application */
  protected static Collection services = new Vector();

  /** create and export services */
```

```
      public static void main(String[] args) throws Exception {
        System.setSecurityManager(new RMISecurityManager());
        for(int count=0; count<args.length; count++) {
          services.add(args[count]);
          // use context classloader to resolve class names
          Naming.rebind("//localhost/"+args[count++],
            (Remote)Thread.currentThread().getContextClassLoader().
              loadClass(args[count++]).getConstructor(
                new Class[] {String.class,int.class}).newInstance(
                new Object[] {args[count++],
                  new Integer(args[count])}));
          // [hk: spend some minutes trying to understand the above
          // 6 lines - if you do manage to understand you should
          // probably be writing C code?]
        }
      }
      /** tearDown services means unPublish */
      public static void stop() {
        Iterator allServices=services.iterator();
        while(allServices.hasNext()) {
          try{
            Naming.unbind("//localhost/"+allServices.next());
          } catch(Exception e) {}
        }
      }
    }
```

If you follow the steps below, you should see these two applications interoperate via a dynamically created classloader hierarchy that is quite similar to the previous example (the RMI-codebase property is required to enable dynamic classloading through rmiregistry; it's value does not matter).

```
  javac -d classes\stock\ stock\*.java
  rmic -d classes\stock\ stock.WarehouseImpl
  javac -classpath classes\stock\ -d classes\sales\ sales\*.java
  javac -d classes\server\ server\*.java
  start rmiregistry
  pause // until rmiregistry is up
  java -Djava.security.policy=server.policy
    -Djava.rmi.server.codebase=infor -classpath classes\server
    server.ApplicationServer
```

Pressing <Enter> for the first time outputs:

```
Deploying file:/J:/misc/deployer/classes/stock/
Starting file:/J:/misc/deployer/classes/stock/

Deploying file:/J:/misc/deployer/classes/sales/
Starting file:/J:/misc/deployer/classes/sales/

stock.WarehouseImpl[RemoteStub [ref: [endpoint:[192.168.202.184:2053]
(local),objID:[0]]]] is about to reserve 5 items.
stock.WarehouseImpl[RemoteStub [ref: [endpoint:[192.168.202.184:2053]
(local),objID:[0]]]] is about to reserve 5 items.
stock.WarehouseImpl[RemoteStub [ref: [endpoint:[192.168.202.184:2053]
(local),objID:[0]]]] is about to reserve 5 items.
```

```
stock.WarehouseImpl[RemoteStub [ref: [endpoint:[192.168.202.184:2053]
(local),objID:[0]]]] is about to reserve 5 items.
stock.WarehouseImpl[RemoteStub [ref: [endpoint:[192.168.202.184:2053]
(local),objID:[0]]]] is about to reserve 5 items.
stock.WarehouseImpl[RemoteStub [ref: [endpoint:[192.168.202.184:2053]
(local),objID:[0]]]] is about to reserve 5 items.
... etc.
```

We can now experience how flexible our application server has become. After the first sales batch has been executed, please set the static "bunch" variable in sales.Order to "10". Recompile the class and press <Enter> for the second time just to see the changed byte-code running in place of the outdated sales representations. That we could realise this behaviour without having to cycle java.exe, even without affecting the referred stock logic in memory, is called "hot-redeploy" - a very useful feature when it comes to incrementally debug server-side logic or to update customer sites on the fly.

```
Deploying file:/J:/misc/deployer/classes/sales/
Stopping file:/J:/misc/deployer/classes/sales/
Starting file:/J:/misc/deployer/classes/sales/

stock.WarehouseImpl[RemoteStub [ref:
[endpoint:[192.168.202.184:2053](local),objID:[0]]]] is
about to reserve 10 items.
 stock.WarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2053](local),objID:[0]]]]
is about to reserve 10 items.
 stock.WarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2053](local),objID:[0]]]]
is about to reserve 10 items.
 stock.WarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2053](local),objID:[0]]]]
is about to reserve 10 items.
 stock.WarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2053](local),objID:[0]]]]
is about to reserve 10 items.
 stock.WarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2053](local),objID:[0]]]]
is about to reserve 10 items.
 stock.WarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2053](local),objID:[0]]]]
is about to reserve 10 items.
 stock.WarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2053](local),objID:[0]]]]
is about to reserve 10 items.
```

Warning: The deprecated Thread.stop() method used in the unDeploy() implementation has allowed me to write concise "newsletter code". Never, I repeat, NEVER use Thread.stop() because it is inherently unsafe: In our example, there is indeed the small opportunity that handling of an application error (an InvocationTargetException reported through reflecting the main method) has not yet obtained the lock to the ApplicationServer.this instance and a simultaneous unDeploy(...) will implant a shutdown exception on top of the cleanup logic. Please, consult your JDK documentation for safe ways to shutdown your threads! [hk: this actually happened once when I ran the tests!]

A second side note: With JDK/JRE 1.3, the apparent relationship between threads and classloaders has been made explicit by associating any thread with a so-called "context classloader" (Thread.setContextClassLoader(ClassLoader loader); Thread.getContextClassLoader()). They are however still unrelated according to 99% of the JDK code which still uses Class.forName(String name) - an awful static method that delegates to the "calling class" defining classloader. The correct way of loading classes in

the light of our new knowledge should be always (see ApplicationServer.deploy() and stock.Main.main()):

```
Thread.currentThread().getContextClassLoader().loadClass(name)
```

## Alternative Classloading Delegation

There is an unwritten law for middleware engineers that no matter how sophisticated your framework (and with exception of the Thread.stop(), we can be really proud of our ApplicationServer, couldn't we?), after two month of releasing the system basis, the application developers will cooperatively abuse it in ways that you have never imagined before.

Be sure that this st\*\*pid stock team will have (as pressed by the product managers and ... sic! ... customers with ... sicut! ... wishes) compiled another, undoubtfully useful, but technically devastating warehouse implementation that is able to delegate unsatisfied reservation calls to other (vendor) warehouses:

```java
// DelegatingWarehouseImpl.java
package stock;
import sales.Order;
/** Example service that links against sales.Order and introduces
 *  a mutual dependency */
public class DelegatingWarehouseImpl extends WarehouseImpl {
  /** name of the item at our vendor */
  protected String vendorItem;
  /** construct a new delegating warehouse */
  public DelegatingWarehouseImpl(String vendorItem, int quantity)
      throws java.rmi.RemoteException {
    super(vendorItem, quantity);
    this.vendorItem = vendorItem;
  }
  /** Overrides the underCapacity reaction to order at vendor */
  protected void empty(int underLoad) {
    try {
      new Order(vendorItem, underLoad).reserve();
      quantity += underLoad;
    } catch(Exception ex) {
      throw new IllegalArgumentException(
        "Could not place order " + ex);
    }
  }
}
```

```
  javac -classpath classes\sales\;classes\stock\
    -d classes\stock\ stock\DelegatingWarehouseImpl.java
```

Guess what ... they will phone you and tell you that there is an ugly NoClassDefFoundError: sales.Order thrown when trying to deploy and run the freshly compiled stock (with a voice indicating that this could not be much of a problem for these Java cracks in the tech department, knowhaddimean, nudge, nudge?).

Boom! Your whole nice server-side architecture collapses as SUN does not allow you to have the salesLoader (hosting DelegatingWarehouseImpl and the Warehouse that is needed by Order) both as parent AND child of the stockLoader (hosting Order that is needed by DelegatingWarehouseImpl). They would not call them children and parent, otherwise, would they?

Your options are now (leaving the Java(tm) platform is not an option, stupid! [hk: why actually not?!?]):

Yelling: "This is not the right project structure! We must extract a more abstract masterdata. Order that is deployed in a separate application and that is used by both sales and stock." But, with a tree-like application structure that follows the SUN classloader hierarchy, this cannot be done ad infinitum. You will most likely end up having most classes in masterdata and under the responsibility of a single person (which in any middle-sized company, sits on the same floor as the system basis programmer, hence this would be not a

good idea).

Crying: "Forget classloader delegation! We deploy everything into a single URLClassloader by exposing the protected addURL(URL url) method." But then, you loose the hot-deploy feature, again! In order to exchange a tiny class in a tiny module in the server, you will have to tear down and restart the whole logic. Now try to sell your customer these large extra coffee breaks at maintenance time!

Thinking: You sit down silently and release a new server. SmartApplicationServer which implements classloader delegation a bit differently to SUN without your colleagues even noticing except that they will no more detect any NoClassDefFoundErrors when implementing the next of their great ideas while you are having that large extra coffee break, a Cohiba Siglo V [hk: I'd opt for a Partagas Series D] and the new Linux Magazine:

```java
// SmartApplicationServer.java
package server;
import java.io.*;
import java.net.*;
import java.util.*;

/** An "application server" that copes with mutual application
 *   dependencies */
public class SmartApplicationServer extends ApplicationServer {
  /** A classloader extension that is able to delegate to other
   *   application's classloaders */
  protected class SmartClassLoader extends URLClassLoader {
    /** mirrors parent constructor */
    public SmartClassLoader(URL[] urls, ClassLoader parent) {
      super(urls,parent);
    }

    /** dispatch "normal" loadClass method to another name */
    protected Class loadClassNormal(String name, boolean resolve)
        throws ClassNotFoundException {
      return super.loadClass(name,resolve);
    }

    /** override "normal" loadClass method in order to delegate */
    protected Class loadClass(String name, boolean resolve)
        throws ClassNotFoundException {
      try{
        // first we try it traditionally
        return loadClassNormal(name,resolve);
      } catch(ClassNotFoundException e) {
        // if this doesnt help, we ask the other application
        // threads for help
        synchronized(SmartApplicationServer.this) {
          Iterator allThreads = applications.values().iterator();
          while(allThreads.hasNext()) {
            SmartClassLoader nextLoader = (SmartClassLoader)
              ((Thread)allThreads.next()).getContextClassLoader();
            if(nextLoader!=null && !nextLoader.equals(this)) {
              try{
                // try the context class loader of next thread
                return nextLoader.loadClassNormal(name,resolve);
              } catch(ClassNotFoundException _e) {
```

```
            }
          }
        }
        // they could not help us, hence we throw an exception
        throw new ClassNotFoundException(
          "class could not be found amongst applications.");
      } // synchronized(SmartApplicationServer.this)
    }  // catch
  } // method loadClass()
} // class

/** override parents factory method to construct dedicated
 *  classloaders */
public ClassLoader constructClassLoader(URL[] urls,
    ClassLoader parent) {
  return new SmartClassLoader(urls,parent);
}

/** example of the improved application server */
public static void main(String[] args) throws Exception {
  BufferedReader stdin = new BufferedReader(
    new InputStreamReader(System.in));
  ApplicationServer appServer=new SmartApplicationServer();
  appServer.deploy(getStockLocation(), "stock.Main",
    new String[] {"screwdriver", "stock.DelegatingWarehouseImpl",
    "vendorScrewer", "20", "vendorScrewer", "stock.WarehouseImpl",
    "vendorScrewer","200"},null);
  appServer.deploy(getSalesLocation(), "sales.Main",
    new String[] {"screwdriver","50"}, null);
  appServer.start(getStockLocation());
  stdin.readLine();
  appServer.start(getSalesLocation());
  stdin.readLine();
  appServer.undeploy(getSalesLocation());
  appServer.undeploy(getStockLocation());
  System.exit(0);
}
}
javac -d classes\server\ server\*.java
java -Djava.security.policy=server.policy
  -Djava.rmi.server.codebase=infor -classpath classes\server
  server.SmartApplicationServer
```

Let me elaborate a bit on the steps that we need to take to realise this third, comfortably sounding scenario. The key ingredient is the introduction of a SmartApplication.SmartClassLoader inner class derived from java.net.URLClassLoader that is intimately coupled to the SmartApplicationServer, more specifically, to its "applications" map that stores the running deployments.

In java.lang.ClassLoader, loadClass(String name) is fixedly and simply implemented to call loadClass(name,false) whose task is to not only to locate and intern the class, but also to immediately resolve linked classes if the second boolean parameter is set to true. We can now easily implement a different delegation semantics by overriding the second method:

First we try to call the "standard" classloading semantics which we have re-exposed as loadClassNormal(String name, boolean resolve). If this succeeds we return the found class (at the minimum

"overhead" of a try/catch block).

Else (in the above example in which we have removed any classloading hierarchy, this happens both if sales.Order tries to link stock.Warehouse as well as if stock.WarehouseImpl tries to link sales.Order) we catch the ClassNotFoundException and iterate through all other deployed applications to also try loadClassNormal(...) through their main thread's context classloader until the right resource has been found.

Only if the class could not be found in all deployed applications, we throw the final ClassNotFoundException.

By these "minimally invasive" extensions, we finally produce the desired output:

```
Deploying file:/J:/misc/deployer/classes/stock/
Deploying file:/J:/misc/deployer/classes/sales/
Starting file:/J:/misc/deployer/classes/stock/

Starting file:/J:/misc/deployer/classes/sales/

stock.DelegatingWarehouseImpl[RemoteStub [ref:
[endpoint:[192.168.202.184:2269](local),objID:[0]]]]
is about to reserve 10 items.
stock.DelegatingWarehouseImpl[RemoteStub [ref:
[endpoint:[192.168.202.184:2269](local),objID:[0]]]] is about
to reserve 10 items.
 stock.DelegatingWarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2269](local),objID:[0]]]] is
about to reserve 10 items.
 stock.WarehouseImpl[RemoteStub [ref:
[endpoint:[192.168.202.184:2269](local),objID:[1]]]] is about
to reserve 10 items.
 stock.DelegatingWarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2269](local),objID:[0]]]] is
about to reserve 10 items.
 stock.WarehouseImpl[RemoteStub [ref:
[endpoint:[192.168.202.184:2269](local),objID:[1]]]] is about
to reserve 10 items.
 stock.DelegatingWarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2269](local),objID:[0]]]]
is about to reserve 10 items.
 stock.WarehouseImpl[RemoteStub
[ref: [endpoint:[192.168.202.184:2269](local),objID:[1]]]]
is about to reserve 10 items.

Stopping file:/J:/misc/deployer/classes/sales/
Stopping file:/J:/misc/deployer/classes/stock/
```

Ideally, this has been just the beginning of a wonderful exploration into the world of "how can I tweak these suboptimal JDK-classes to fit my practical needs". For example, you will most likely ask how the hell an application programmer or even a customer deployer should be able to manage all those dependencies between applications (usually: jar-files, such as stock.jar and sales.jar) once your system has evolved to a particular size. The solution to this is quite straightforward: Jar dependencies are usually encoded in the Manifest.mf class-path entry and that is where your application server can obtain them in order to identify deployment needs and deployment order.

Similarly, the loadClass calls delegated from one application to another (in a full-blown implementation, also calls to getResource(...) must be delegated) introduce runtime dependencies: If you cycle the byte code of the "defining application" you must also cycle the dependent byte-code of the consuming application. I leave this (together with caching of class locations) as an exercise for you.

Alternatively, if you are a bit lazy, nevertheless curious, i.e., your are a system programmer, you can have a look at the latest org.jboss.deployment.scope.* sources of the JBoss Open Source application server (http://www.jboss.org) whose ingenious mastermind, Marc Fleury, has been the originator of the presented classloading semantics. Besides clustering support, the goal for the next major release of Jboss (3.0 - Project Rabbit Hole) is to have a deployer that can manage multiple scopes, i.e., "virtual applications" such as we have today constructed out of stock.* and sales.*

He, he!

[hk: hihi]

---

Thanks for taking the time to read this newsletter, please also take the time to give us feedback, we always appreciate it.

Heinz

---

Receive our newsletter with new tips! More than 8,500 subscribers (by December 2001) can not be wrong! They read our tips every week!
To subscribe to "The Java FAQ Daily Tips" weekly edition newsletter send e-mail wuth "subscribe" word in the header and the body (write just subscribe without ""!!!) to:
javafaqtips-request@javafaq.nu
or on the web:
http://javafaq.nu/mailman/listinfo/javafaqtips_javafaq.nu

## Writing GUI Layout Managers

## Author: Dr. Heinz M. Kabutz

## Copyright 2001 Maxkab Solutions CC - South Africa
## Reproduced as part of the JavaTips with permission.

If you are not already subscribed to this newsletter, you can either subscribe via email or subscribe via the web. Be warned that if you are a beginner in Java, you will at times struggle to keep up.

---

Welcome to the 10th issue of "The Java(tm) Specialists' Newsletter". It has been a lot of fun writing these newsletters, especially since there is soooo much to write about in Java. Today I heard about some cases where in the JDK 1.3 the GUI threads get started which were not started in JDK 1.2.2. The effect is that command-prompt driven server code suddenly did not shut down anymore. An example is at the end of this newsletter thanks to James Pereira, Technical Director of Foundation Technologies, one of very few directors who still finds time to actually write Java code. He promised me a newsletter on class versioning with multiple ClassLoaders which should be very interesting. These differences in Java VMs harks of the days when we used to use C macros (#ifdef) to distinguish between versions of a 8087 FP chip used on a specific machine. At the rate Java gets changed, we have to be VERY careful to not rely on the existence of certain bugs.

But now, let's go back a few years, when JBuilder 2.0 had just come out. In those days, the GridBagLayout was not supported as well as in the later JBuilder versions, so we tried to avoid using it where possible. If you want to age quickly, try and maintain some GridBagLayout code written by someone else ...

A colleague of mine wanted to do a screen layout similar to BorderLayout but with different rules. The rules for BorderLayout are that the layout manager allocates preferred height for North and South components, then preferred width for East and West components, and the rest goes to Center. He wanted to have Left, Middle and Right components where the Middle took on its preferred width and Left and Right took on the remaining width shared equally. It would look like this:

```
OOOOOOOOOO       OOOOOOOOOO
OOOOOOOOOO       OOOOOOOOOO
OOOOOOOOOOaaaaOOOOOOOOOO
OOOOOOOOOOaaaaOOOOOOOOOO
OOOOOOOOOOaaaaOOOOOOOOOO
OOOOOOOOOO       OOOOOOOOOO
OOOOOOOOOO       OOOOOOOOOO
```

if we resized the screen, it would look like this:

```
OOOOOOOOOOOOOO     OOOOOOOOOOOOOO
OOOOOOOOOOOOOO     OOOOOOOOOOOOOO
OOOOOOOOOOOOOO     OOOOOOOOOOOOOO
```

```
OOOOOOOOOOOOOOaaaaOOOOOOOOOOOOOO
OOOOOOOOOOOOOOaaaaOOOOOOOOOOOOOO
OOOOOOOOOOOOOOaaaaOOOOOOOOOOOOOO
OOOOOOOOOOOOOO     OOOOOOOOOOOOOO
OOOOOOOOOOOOOO     OOOOOOOOOOOOOO
OOOOOOOOOOOOOO     OOOOOOOOOOOOOO
```

or

```
OOOOOOO     OOOOOOO
OOOOOOOaaaaOOOOOOO
OOOOOOOaaaaOOOOOOO
OOOOOOOaaaaOOOOOOO
OOOOOOO     OOOOOOO
```

We tried the standard way of combining layout managers with panels but were unable to do this layout. In those days without decent GridBagLayout tool support we were very hesitant to hack around with that monster layout. (Incidentally, the same type of layout is used by the jGuru crew to demonstrate that you need the GridBagLayout, so they also did not seem to get it right with normal layout managers.)

This caused me to try and write my own Layout Manager, which I called the WildLayoutManager, named after my ex-colleague, who is.

## Writing your own GUI Layout Manager

It is actually increadibly straightforward writing your own layout manager. This layout manager took me one lunch time, i.e. less than 1 hour. Permanent employees at that company get free lunches, but as a contractor I had to pay, so I programmed instead of ate in those days. Nowadays I simply talk about work during lunch and charge the time ;-)

Seriously, it really is easy. When I tell people that I've written my own LayoutManager to do the layout for me they get all boggle-eyed and start holding up crucifixes, garlic or search for the silver bullet. (Those in the know, know that there is no silver bullet for software development.)

```java
//: WildLayoutManager.java
import java.awt.*;

public class WildLayoutManager implements LayoutManager {
  // these are the constraints possible with the WildLayoutManager
  public static final String LEFT = "Left";
  public static final String RIGHT = "Right";
  public static final String MIDDLE = "Middle";

  // We keep handles to three components, left, right and middle
  private Component left;
  private Component right;
  private Component middle;

  // we need to be able to add components.  if two components are added
  // with the same constraint we keep the last one
  public void addLayoutComponent(String name, Component comp) {
    if (LEFT.equals(name)) {
      left = comp;
    } else if (RIGHT.equals(name)) {
      right = comp;
    } else if (MIDDLE.equals(name)) {
      middle = comp;
    } else {
      throw new IllegalArgumentException(
        "cannot add to layout: unknown constraint: " + name);
    }
```

```java
    }

    // here we remove the component - first find it!
    public void removeLayoutComponent(Component comp) {
        if (comp == left) {
            left = null;
        } else if (comp == right) {
            right = null;
        } else if (comp == middle) {
            middle = null;
        }
    }

    // The minimum dimension we're happy with is the preferred size
    // this could be more fancy by using the minimum sizes of each component
    public Dimension minimumLayoutSize(Container parent) {
        return preferredLayoutSize(parent);
    }

    // Here we work out the preferred size of the component, which is used
    // by methods such as pack() to work out how big the window should be
    public Dimension preferredLayoutSize(Container parent) {
        Dimension dim = new Dimension(0, 0);
        // get widest preferred width for left && right
        // get highest preferred height for left && right
        // add preferred width of middle
        int widestWidth = 0;
        int highestHeight = 0;
        if ((left != null) && left.isVisible()) {
            widestWidth = Math.max(widestWidth, left.getPreferredSize().width);
            highestHeight =
                Math.max(highestHeight, left.getPreferredSize().height);
        }
        if ((right != null) && right.isVisible()) {
            widestWidth = Math.max(widestWidth, right.getPreferredSize().width);
            highestHeight =
                Math.max(highestHeight, right.getPreferredSize().height);
        }
        dim.width = widestWidth * 2;
        dim.height = highestHeight;
        if ((middle != null) && middle.isVisible()) {
            dim.width += middle.getPreferredSize().width;
            dim.height = Math.max(dim.height, middle.getPreferredSize().height);
        }

        Insets insets = parent.getInsets();
        dim.width += insets.left + insets.right;
        dim.height += insets.top + insets.bottom;

        return dim;
    }

    // this is the brain of the layout manager, albeit rather small.
    // I told you this is straightforward...
    public void layoutContainer(Container target) {
```

```java
      // these variables hold the position where we can draw components
      // taking into account insets
      Insets insets = target.getInsets();
      int north = insets.top;
      int south = target.getSize().height - insets.bottom;
      int west = insets.left;
      int east = target.getSize().width - insets.right;

      // we first find the width of the left and right components
      int widestWidth = 0;
      if ((left != null) && left.isVisible()) {
        widestWidth = Math.max(widestWidth, left.getPreferredSize().width);
      }
      if ((right != null) && right.isVisible()) {
        widestWidth = Math.max(widestWidth, right.getPreferredSize().width);
      }
      if ((middle != null) && middle.isVisible()) {
        widestWidth = Math.max(widestWidth,
          (east - west - middle.getPreferredSize().width) / 2);
      }

      // next we set the size of the left component equal to the widest width
      // and whole height, and we set the bounds from North-West corner
      if ((left != null) && left.isVisible()) {
        left.setSize(widestWidth, south - north);
        left.setBounds(west, north, widestWidth, south - north);
      }
      // next we set the size of right component equal to the widest width
      // and whole height, and we set the bounds from North-East corner
      if ((right != null) && right.isVisible()) {
        right.setSize(widestWidth, south - north);
        right.setBounds(east-widestWidth, north, widestWidth, south - north);
      }
      // lastly we set the size of the middle component equals to the
      // remaining width, which should be equal to the middle object's
      // preferred width and we set the height equal to the middle object's
      // preferred height
      if ((middle != null) && middle.isVisible()) {
        middle.setSize(east - west - widestWidth * 2,
          middle.getPreferredSize().height);
        middle.setBounds(
          west+widestWidth,
          north + (south - north - middle.getPreferredSize().height)/2,
          east - west - widestWidth * 2,
          middle.getPreferredSize().height);
      }
    }
  }
```

You see, it really was quite simple. Here is an example frame that tries out the new layout manager:

```java
//: WildLayoutExample.java
import java.awt.*;
import javax.swing.*;

public class WildLayoutExample extends JFrame {
```

```java
    public WildLayoutExample() {
      super("WildLayoutExample");
      setSize(new Dimension(400, 300));
      getContentPane().setLayout(new WildLayoutManager());
      // construct the left panel
      JPanel leftPanel = new JPanel(new BorderLayout());
      leftPanel.add(new JLabel("Left Label"), BorderLayout.NORTH);
      leftPanel.add(new JTree(), BorderLayout.CENTER);
      // construct the middle panel
      JPanel middlePanel = new JPanel(new GridLayout(0,1,5,5));
      middlePanel.add(new JButton("Add >"), null);
      middlePanel.add(new JButton("<< Remove All"), null);
      // construct the right panel
      JPanel rightPanel = new JPanel(new BorderLayout());
      rightPanel.add(new JLabel("Right Label"), BorderLayout.NORTH);
      rightPanel.add(new JTextArea("jTextArea1"), BorderLayout.CENTER);
      // add the panels to the content pane using our new layout manager
      getContentPane().add(leftPanel, WildLayoutManager.LEFT);
      getContentPane().add(middlePanel, WildLayoutManager.MIDDLE);
      getContentPane().add(rightPanel, WildLayoutManager.RIGHT);
    }
    public static void main(String[] args) {
      WildLayoutExample frame = new WildLayoutExample();
      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); // JDK 1.3 !
      frame.setVisible(true);
    }
  }
```

You could use the idea of custom layout managers to create all sorts of interesting layouts, such as a special form layout for designing forms for your business application. I do not see any particular problems with writing your own layout manager, especially if there is some layout that you want to use quite often. Just don't use absolute layouts, whatever you do!!!

And remember: let's be careful out there!

Heinz

---
*Warning Advanced:*
*The following code exits normally under JDK 1.2.2 but "hangs" under JDK 1.3. Many thanks to James Pereira for pointing this out and sending me the code:*

```java
  import java.awt.Cursor;
  public class J23D {
    // This cursor class when loaded now starts the GUI event threads!
    private static Cursor s_waitCursor =
      Cursor.getPredefinedCursor(Cursor.WAIT_CURSOR);
    public void log(String _msg) {
      System.out.println(_msg);
    }
    public static void main ( String[] _args ) {
      new J23D().log("That's all folks!");
    }
  }
```

---

(c)1999, 2000, 2001, 2002. JavaFAQ.nu. All rights reserved worldwide.
This document is not free for distribution!
This document can not be changed, either in whole or in part
without the express written permission of the publisher.
All questions please mailto:info@javafaq.nu