



the
POWER
of
JAVA™



JavaOne
Part of the Network for Business Success

Deployment Tips and Tricks for Using Java™ Web Start and Java Plug-in Software

Andy Herrick

Thomas Ng

Java Web Start Engineering
Sun Microsystems

TS-1319

Copyright © 2006, Sun Microsystems, Inc., All rights reserved.

2006 JavaOneSM Conference | Session TS-1319 |

java.sun.com/javaone/sf

Goal

Learn to solve common deployment issues with Java™ Web Start and Java Plug-in software

Agenda

Get Java platform

Use script to detect and deploy the Java platform

Get Deployed

Use Java Web Start and Java Plug-in software

Get Jazzed

Use Java Network Launching Protocol API to enhance and empower your app

Get Working

Use trace output, NetBeans™ IDE, and JConsole

Get Fast

Use jardiff, preload, lazy loading, and Pack200

Agenda

Get Java platform

Use script to detect and deploy the Java platform

Get Deployed

Use Java Web Start and Java Plug-in software

Get Jazzed

Use Java Network Launching Protocol API to enhance and empower your app

Get Working

Use trace output, NetBeans IDE, and JConsole

Get Fast

Use jardiff, preload, lazy loading, and Pack200

Get Java Platform

Ubiquity of Java Technology Problem

- Java technology is very common on most PCs, but not on **all** machines
- Even if Java technology exists, it might be one of many versions
- Developers (**you**) need to deploy Java applications/applets
- How to deploy if you don't know which Java platform to target, or whether that Java platform is available?

Get Java Platform

Ubiquity of Java Technology Problem

- Need: Launch Java application or applet from the browser with a specified minimum version of Java platform
 - Detect whether the Java platform exists on the system
 - If it exists, detect what version
 - If it does not exist, or does not meet minimum version requirement, install the latest version
 - After the Java platform is installed, launch the application or applet
- You can use Java Platform, Standard Edition 5 (Java SE) to solve many of these problems today!

Get Java Platform

Java Platform Detection or Installation in 5.0

- JavaScript™ technology detection logic
- Family detection—5.0 family or 1.4.2 family
- Internet Explorer on Windows:
 - Java Web Start software ActiveX detection
 - Java platform auto-install with OBJECT tag
- Firefox and Mozilla family
 - navigator.mimeTypes for detection
 - GetJava download site for Java install software

Agenda

Get Java platform

Use script to detect and deploy the Java platform

Get Deployed

Use Java Web Start and Java Plug-in software

Get Jazzed

Use Java Network Launching Protocol API to enhance and empower your app

Get Working

Use trace output, NetBeans IDE, and JConsole

Get Fast

Use jardiff, preload, lazy loading, and Pack200

Get Deployed

- Combine the deployment of the Java platform software with the deployment of your application or applet
- Example
 - Deploy Notepad as a Java Web Start application
 - Deploy SwingSet as a Java Plug-in applet

Notepad App with Java SE 5+

```
<input type=button value="Notepad" onclick="launch()">
// Javascript
function launch() {
    if (isJava5IE() || isJava6IE() ||
        isJava5FF() || isJava6FF()) {
        // Launch directly if Java 5+ is available
        window.location="http://foo.com/notepad.jnlp";
    } else if (isWindowsIE()) {
        // Auto-Install Java, then launch Notepad
        launchJNLPJavaAutoInstall();
    } else {
        // Redirect to GetJava and launch Notepad
        launchJNLPGetJava();
    }
}
```

Detect Java SE 5 Platform on IE

```
function isJava5IE() {  
    try {  
        // Detect Java 5 platform  
        var obj = new ActiveXObject(  
            "JavaWebStart.isInstalled.1.5.0.0");  
        if (obj != null) {  
            return 1;  
        }  
    } catch (exception) {  
        return 0;  
    }  
}
```

- For Java 6 platform, use `JavaWebStart.isInstalled.1.6.0.0`

Detect Java SE 5 on Firefox and Mozilla™

```
// Java Detection for Firefox, Mozilla and Netscape
function isJava5FF() {
    for (var i = 0; i < navigator.mimeTypes.length; ++i) {
        s = navigator.mimeTypes[i].type;
        // Detect Java 5 platform
        if (s == "application/x-java-applet;version=1.5") {
            return 1;
        }
    }
    return 0;
}
```

- For Java 6 platform, use version=1.6

Java Auto-Install with IE

```
function launchJNLPJavaAutoInstall() {
    // Auto-Install latest Java platform; after install
    // is complete, the Java Web Start ActiveX object will
    // launch the Notepad application
    document.write("<OBJECT
        classid=clsid:5852F5ED-8BF4-11D4-A245-0080C6F74284
        codebase=http://java.sun.com/update/1.5.0/
        jinstall-1_5-windows-i586.cab#Version=5,0,0,99
        height=0 width=0>");
    document.write("<PARAM name=app
        value=http://foo.com/notepad.jnlp>");
    document.write("</OBJECT>");
}
```

Redirect to GetJava page for FireFox

```
function launchJNLPGetJava() {
    // Open new browser window to GetJava site
    window.open("http://java.com/getjava", "newWindow");
    // Spin and wait for install complete to launch app
    // Infinite loop possible here if user bail out on
    // install, a better version should use timeout
    launchTID = setInterval(
        'launchApp("http://foo.com/notepad.jnlp")', 100);
}
function launchApp(app) {
    navigator.plugins.refresh(true);
    x = navigator.mimeTypes['application/x-java-
        applet;version=1.5'];
    if (x) {
        window.location = app;
        clearInterval(launchTID);
    }
}
```

SwingSet Applet with Java SE 5+

```
<input type=button value="SwingSet" onclick="launch()">
```

```
function launch() {  
    if (isWindowsIE()) {  
        // Install latest Java Platform if needed, and  
        // launch SwingSet  
        generateSwingSetObjectTag();  
    } else {  
        // Launch SwingSet with EMBED tag  
        generateSwingSetEmbedTag();  
    }  
}
```

Java Auto Install with IE

```
function generateSwingSetObjectTag() {
    // Launch SwingSet Applet with Java Platform 5.0+
    // If not available, install latest Java Platform
    document.write("<OBJECT
        classid=clsid:8AD9C840-044E-11D1-B3E9-00805F499D93
        codebase=http://java.sun.com/update/1.5.0/
        jinstall-1_5-windows-i586.cab#Version=5,0,0,99
        width=695 height=525>");
    document.write("<PARAM name=archive
        value=SwingSet.jar>");
    document.write("<PARAM name=code
        value=SwingSetApplet>");
    document.write("</OBJECT>");
}
```


Redirect to GetJava Page for Firefox

```
function generateSwingSetEmbedTag() {  
    // Use Java 5+ to launch applet; redirect to GetJava  
    // page for install if needed  
    document.write("<EMBED  
        type=application/x-java-applet;version=1.5  
        code=SwingSetApplet archive=SwingSet.jar  
        width=695 height=525  
        pluginspage=http://java.com/getjava />");  
}
```

DEMO

Get Deployed with JavaScript Technology

Use Deployment Toolkit

Java Platform Detection or Installation in Mustang

- Deployment Toolkit
 - ActiveX object for Internet Explorer
 - Plug-in for Firefox
- JavaScript APIs
- Richer detection and installation support
- Same user experience for Internet Explorer and Firefox!

Notepad App with Java SE 5.0_07+

```
<input type=button value="Notepad" onclick="launch()">
```

```
function launch() {  
    if (isDeploymentToolkitAvailable()) {  
        launchNotepad();  
    } else {  
        // Previous Java 5+ detection code  
        ...  
    }  
}
```

Deployment Toolkit Detection

```
function isDeploymentToolkitAvailable() {
    if (isDtkAvailableIE() || isDtkAvailableFF()) {
        return 1
    }
    return 0;
}
function isDtkAvailableFF() {
    x = navigator.mimeTypes['application/npruntime-
        scriptable-plugin;DeploymentToolkit'];
    return x ? 1 : 0;
}
function isDtkAvailableIE() {
    return isJava6IE();
}
```

Initialize the Toolkit Object

```
// Get the DeploymentToolkit Object
<OBJECT
  classid="clsid:CAFEEFAC-DEC7-0000-0000-ABCDEFEDCBA"
  id="deployToolkit" width=1 height=1>
</OBJECT>
<COMMENT>
<EMBED id="npDeployToolkit" type="application/npruntime-
  scriptable-plugin;DeploymentToolkit" width=1 height=1>
</EMBED>
</COMMENT>

<SCRIPT language="javascript">
if (isWindowsIE() == false) {
  deployToolkit = npDeployToolkit;
}
// continue on next slide
```

Deployment Toolkit Example

```
function launchNotepad() {
    var ret = 1;
    // Check and see if Java Platform 5.0u7+ is installed
    if (!isJava5u7PlusAvailable()) {
        // Install the latest Java platform
        ret = deployToolkit.installLatest();
    } else {
        ret = 0;
    }
    // if we have 5.0u7+ installed, launch the application
    if (ret == 0) {
        // launch the jnlp application
        deployToolkit.launch("http://foo.com/notepad.jnlp");
    }
}
```

Detect 5.0u7+ with Deployment Toolkit

```
function isJava5u7PlusAvailable() {
    // get the list of JVMs installed in the machine
    var jvms = deployToolkit.jvms;
    var listlen = jvms.getLength();
    // Go thru the list of JVMs installed
    for (i=0; i<listlen; i++) {
        var version = jvms.get(i).version;
        // make sure a 5.0u7+ JVM is installed
        var compRes = deployToolkit.compareVersion(version,
            "1.5.0_07");

        if (compRes >= 0) {
            return 1;
        }
    }
    return 0;
}
```


SwingSet Applet with 5.0u7+

```
<input type=button value="SwingSet" onclick="launch()">
```

```
function launch() {  
    if (isDeploymentToolkitAvailable()) {  
        launchSwingSet();  
    } else {  
        // Previous Java 5+ detection code  
        ...  
    }  
}
```

Deployment Toolkit Example

```
function launchSwingSet() {
    var ret = 1;
    // Check if the browser default Java is 5.0u7+
    if (!isJava5u7PlusBrowserDefault()) {
        // Install the latest Java platform
        ret = deployToolkit.installLatest();
    } else {
        ret = 0;
    }
    // If 5.0u7+ is browser default JVM, launch the applet
    if (ret == 0) {
        // launch the SwingSet applet
        generateSwingSetAppletTag();
    }
}
```

Detect Browser Java Technology with Deployment Toolkit

```
function isJava5u7PlusBrowserDefault() {  
    // get the default Java version registered with the  
    // browser  
    var version = deployToolkit.browserJVM.version;  
  
    // make sure it is a 5.0u7+ JVM  
    var compRes = deployToolkit.compareVersion(version,  
        "1.5.0_07");  
  
    if (compRes >= 0) {  
        return 1;  
    }  
    return 0;  
}
```

Launch Applet with Latest Java Platform

```
function generateSwingSetAppletTag() {  
    // Use latest Java Platform to launch applet  
    document.write("<APPLET code=SwingSetApplet  
        archive=SwingSet.jar width=695 height=525>");  
    document.write("</APPLET>");  
}
```

DEMO

Get Deployed with Deployment Toolkit

Agenda

Get Java platform

Use script to detect and deploy the Java platform

Get Deployed

Use Java Web Start and Java Plug-in software

Get Jazzed

Use Java Network Launching Protocol API to enhance and empower your app

Get Working

Use trace output, NetBeans IDE, and JConsole

Get Fast

Use jardiff, preload, lazy loading, and Pack200

Use Launch File Hints

- Associations

```
// Register your JNLP application as default handler  
// of specific file extensions and mime-type  
<association extensions="note"  
    mime-type="application/x-notepad-note"/>
```

- Related Content

```
// Include a help file with your application  
<related-content href="helpfile.note" />
```

- Shortcuts

```
// Create a shortcut for your application in a specific  
// start-menu folder  
<shortcut> <menu submenu="notepad"/> </shortcut>
```

Single Instance Service

- Client will run only one instance of application
- Each subsequent attempt to launch application will trigger call to `newActivation()` in the original instance
- Works well in conjunction with associations
 - Launching via associations now grants the application file permissions to the file being opened or printed

DEMO

SingleInstanceService and Associations in
Notepad

Runtime Check for Update

- JNLPClassLoader now extends URLClassLoader
- Use `URLClassLoader.getURLs()` to get Array of the urls that make up the application
- Use `URLConnection.getLastModified()` with HEAD request to test each url to see if it has been updated
- If there has been an update you can dynamically restart the application

Check for Update

```
final HashMap map = new HashMap();
private void startUpdateCheck() {
    final URLClassLoader cl = (URLClassLoader)
        this.getClass().getClassLoader();
    new Thread(new Runnable() {
        public void run() {
            for (;;) {
                if (updated(cl, map)) {
                    restartApplication();
                }
                try { // check for update every 5 seconds
                    Thread.currentThread().sleep(5000);
                } catch (Exception exception) { }
            }
        }
    }).start();
}
```

Check for Update

```
private boolean updated(URLClassLoader cl, HashMap map) {
    URL [] urls = cl.getURLs();
    try {
        for (int i=0; i<urls.length; i++) {
            if (urlUpdated(urls[i], map)) {
                return true;
            }
        }
    } catch (IOException ioe) {
        System.err.println(ioe.toString());
    }
    return false;
}
```

Check for Update

```
private boolean urlUpdated(URL url, HashMap map)
    throws IOException{
    URLConnection conn = url.openConnection();
    if (conn instanceof HttpURLConnection) {
        ((HttpURLConnection) conn).setRequestMethod("HEAD");
        long time = conn.getLastModified();
        ((HttpURLConnection) conn).disconnect();
        if (map.containsKey(url.toString())) {
            Object o = map.get(url.toString());
            return (((Long)o).longValue() < time);
        } else {
            map.put(url.toString(), new Long(time));
        }
    }
    return false;
}
```

Restart Application

- Use `DownloadService` to get the codebase
- Use `BasicService` to launch the application using `showDocument`
- Prompt user to save work
 - You can use new modality modes in AWT
- Don't forget to remove `SingleInstanceListener`

Restart Application Code

```
private void restartApplication() {
    try {
        promptForSave();
        if (sis != null) {
            sis.removeSingleInstanceListener(this);
        }
        URL codebase = basicService.getCodeBase();
        URL app = new URL(codebase, "notepad.jnlp");
        basicService.showDocument(app);
        System.exit(0);
    } catch (IOException ioe) {
        ioe.printStackTrace();
    }
}
```

DEMO

Check for Update and Restart

Dynamically Downloading Resources

- DownloadService methods: `isResourceCached`, `loadResource`, and `removeResource` no longer require listing in the launch file
 - Can be used to load resource jar only containing the locale in use
 - Must either be called by trusted code, or refer to resources from the same codebase
- These methods can now be used with a reference to a launch file as the resource
 - Can be used to remove yourself from cache

DEMO

Self Removal



Agenda

Get Java platform

Use script to detect and deploy the Java platform

Get Deployed

Use Java Web Start and Java Plug-in software

Get Jazzed

Use Java Network Launching Protocol API to enhance and empower your app

Get Working

Use trace output, NetBeans IDE, and JConsole

Get Fast

Use jardiff, preload, lazy loading, and Pack200

Tracing

- Turn on trace file output in the advanced panel of Java Control Panel
- Trace files will (by default) go into the `<deployment.user.home>/log/` directory
- Dynamically change trace level using the Java Console
- Set the initial (startup) trace level by adding deployment property
 - `deployment.trace.level=basic|cache|net|security|temp`
 - `deployment.trace.level=all`

Debugging

- To prevent double launching, always launch javaws with a file argument
 - “javaws -<options> <jnlp-file>”
- Use “-J” arg to pass additional arguments to Java technology
 - “javaws -J-Xrunjdwp:transport=dt_socket,server=y,address=8000 -J-Xdebug notepad.jnlp”
- This is the same as setting the environment variable JAVAWS_VM_ARGS to
 - “-Xrunjdwp:transport=dt_socket,server=y,address=8000 -Xdebug”

NetBeans Module for Java Web Start Software

- Download and install NetBeans 5.0 IDE
- Download Java Web Start software module from the NetBeans Update Center Beta
- See instructions at:
 - <http://www.netbeans.org/kb/articles/matisse-jaws.html>

Monitor Applications Using JConsole

- No longer “experimental” in JDK™ 6 software
- Use JConsole to monitor you application
 - Monitor memory usage
 - Monitor garbage collection activities
 - Find out the number of objects pending for finalization
 - Obtain heap histogram to help determine the leak suspect
 - Obtain heap dump snapshots and use that for heap analysis

DEMO

Notepad with NetBeans IDE

Agenda

Get Java platform

Use script to detect and deploy the Java platform

Get Deployed

Use Java Web Start and Java Plug-in software

Get Jazzed

Use Java Network Launching Protocol API to enhance and empower your app

Get Working

Use trace output, NetBeans IDE, and JConsole

Get Fast

Use jardiff, preload, lazy loading, and Pack200

Minimize Download Size

- Use JNLPDownloadServlet
- Use lazy download
- Use application preload
- Use Pack200

Use the JNLPServlet

- Implement the version-based download protocol
 - Protocol is defined in the Java Network Launching Protocol and API specification
 - Allows you to use the best available version for an ordered range of versions given
- Implement jardiff for minimal update downloads
 - Download only the incremental updates to versioned resources
- One way to enable Pack200

Use Lazy or Preloaded Resources

- Use Lazy Download
 - Download components only when you need them
 - Use the DownloadService API, or just allow the ClassLoader to download on demand
 - Jar Indexing now works in Java Web Start software
- Use Application Preload
 - Download when network is not busy
 - Mostly for enterprise/intranet

Use Pack200

- Hyper compression for JAR file
- No change required in launch file or html file
- Create packed jars using tools from JDK software
 - pack200 and unpack200
- Required server side changes
 - Both packed and original jars must be hosted
 - Servlet or JSP™ technology is required to serve the packed jars
 - JNLPServlet provided in the JDK software

Use Pack200

How Do I Know Packed Jars Are Really Used?

- Client side
 - Enable Java console and tracing
 - basic: Downloading `http://foo.com/SwingSet.jar` to cache
 - basic: encoding = pack200-gzip for `http://foo.com/SwingSet.jar`
- Server side
 - Check JNLPDownloadServlet log
 - JnlpDownloadServlet(4): DownloadRequest[path=/SwingSet.jar encoding=pack200-gzip,gzip isPlatformRequest=false]
 - JnlpDownloadServlet(4): Real resource returned: JnlpResource[WAR Path: /SwingSet.jar.pack.gz lastModified=Wed Mar 01 10:57:43 PST 2006]]

Pack200 Without a Servlet

- Possible on Apache web server
 - Tested on Apache HTTP Server 2.0.52
- Use of .htaccess file and “type map file”
 - Server must allow overrides with .htaccess
- Remove servlet requirement for Pack200 support
 - Just drop these files to your application directory
- No versioning or jardiff support as in JNLPDownloadServlet

Pack200 Without Servlet Example

- For test.jnlp and file.jar with pack support, you need the following files:

.htaccess

test.jnlp // points to file.jar

file.jar.var // type map for file.jar

file.jar.pack.gz // pack200 compressed file.jar

file.jar // original file.jar

Pack200 Without Servlet Example

```
# .htaccess
# Return the right mime type for JARs
AddType application/x-java-archive .jar
# Enable type maps
AddHandler application/x-type-map .var
Options +MultiViews
# Tweak MultiViews - this line is for
# APACHE 2.0 ONLY!
MultiViewsMatch Any
<Files *.ppack.gz>
    # Enable Content-Encoding header for .jar.pack.gz files
    AddEncoding pack200-gzip .jar
    # Stop mod_gzip from messing with the Content-Encoding
    # response for these files
    RemoveEncoding .gz
</Files>
```

Pack200 Without Servlet Example

```
# file.jar.var  
# Type map file - one for each jar
```

```
URI: file.jar.pack.gz  
Content-Type: x-java-archive  
Content-Encoding: pack200-gzip
```

```
URI: file.jar  
Content-Type: x-java-archive
```

Summary

Get Java Platform

Get Deployed

Get Jazzed

Get Working

Get Fast

Put these tips and tricks to work today!

For More Information

- **GetJava:**
 - <http://java.com/getjava>
- **Java SE 5 deployment documentation:**
 - <http://java.sun.com/j2se/1.5.0/docs/guide/deployment>
 - <http://java.sun.com/j2se/1.5.0/docs/guide/javaws>
 - <http://java.sun.com/j2se/1.5.0/docs/guide/plugin>
- **Deployment products pages:**
 - <http://java.sun.com/products/javawebstart/>
 - <http://java.sun.com/products/plugin/>
- **NetBeans Java Web Start Module:**
 - <http://www.netbeans.org/kb/articles/matisse-jaws.html>

Q&A

<code />



the
POWER
of
JAVA™



JavaOne
Sun and Network on Demand Software

Deployment Tips and Tricks for Using Java™ Web Start and Java Plug-in Software

Andy Herrick

Thomas Ng

Java Web Start Engineering
Sun Microsystems

TS-1319