



# Is there a place for Java applets in Web 2.0?

Jim Crossley & Gary Thompson

TS-5453



# Agenda

- Introduction (GT)
- The history of rich web applications (GT)
  - What is a Web 2.0 app?
  - Why aren't applets used now?
- Examples (JC)
- Why do Java™ 6 technology and JavaFX™ software make a difference? (JC)
- Why applets are good for Web 2.0 (GT)

# The history of rich web applications

## ➤ What is a Web 2.0 app?

- Offers a high degree of interactivity to the user
- Displays new data updates immediately – no page refreshing
- Typically implemented using AJAX or Flash, or a combination

## ➤ Why aren't applets used now?

- JVM™ software issues
  - Large download
  - Slow startup
  - Deployment issues (Heap size)
- Poor historical perception of applets
- Poor multimedia support

# Possible topics

Signed  
vs  
Unsigned Applets

JVM Issues

Browser and  
OS issues

Applets and WebStart

Server-side  
integration

Applet UI  
AWT / Swing / JavaFX

Interacting with  
the web page.

Applet debugging  
and profiling

Deployment options

Applet  
development tools

Thin and Thick clients

Reusing Applets  
as  
Applications or Midlets

# Topics

Signed  
vs  
Unsigned Applets

JVM Issues

Browser and  
OS issues

Applets and WebStart

Server-side  
integration

Applet UI  
AWT / Swing / JavaFX

Interacting with  
the web page.

Applet debugging  
and profiling

Deployment options

Applet  
development tools

Thin and Thick clients

Reusing Applets  
as  
Applications or Midlets

# Topic: Interacting with the web page

- Applets often live in isolation on web pages
- Example 1 – JavaScript™ language to Applet
  - Code example
  - Live example application
- Example 2 – Applet to JavaScript
  - Code example
  - Live example application
- Example 3 – Interacting with Flash?
  - Code example
  - Live example application

## Example 1 - JavaScript to Applet

- You can use JavaScript language to access the public methods of an Applet like any other element in an HTML page

```
function setAppletValue() {  
    var applet = document.getElementById('applet');  
    applet.setText("Hello");  
}
```

- May need SCRIPTABLE applet attribute for IE

# Example 1 – JavaScript to Applet

- Live example of JavaScript language calling Applet

Not set

## Example 1 - Javascript to Applets



## Example 2 – Applet to JavaScript

- Step 1. Define a JavaScript method in the HTML page:

```
function appletClick(param1, param2) {  
    alert("Applet sent " + param1 + ", " + param2);  
}
```

- Add the MAYSCRIPT attribute to your applet

```
<applet  
  code="com.altio.examples.applets.JavascriptSwing.cl  
  ass" archive="exampleapplets.jar"  
  id='exampleApplet' width=400 height=75 MAYSCRIPT>
```

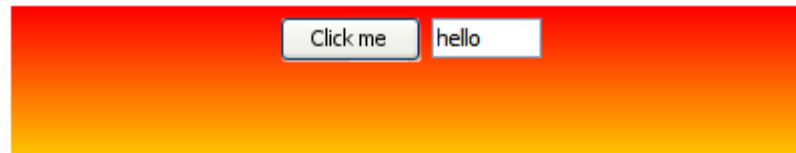
## Example 2 – Applet to JavaScript

- Step 2. Use LiveConnect to execute JavaScript methods:

```
import netscape.javascript.JSObject;  
  
JSObject window = JSObject.getWindow(this);  
String[] params = new String[] {  
    "Value 1",  
    "Value 2" };  
window.call("appletClick", params);
```

# Example 2 – Applet to JavaScript

- Live Example of Applet calling JavaScript language



## Applets and AJAX

hello

hello

The value of JTextField1 is Some text

## Example 2 – Applet to JavaScript

- Newer (1.5+) plugins can use the Common DOM classes to access browser DOM from applets

```
DOMService service = DOMService.getService(this);
final Applet myApp = this;
String title = (String) service.invokeAndWait(new
DOMAction() {
    public Object run(DOMAccessor accessor) {
        HTMLDocument doc = (HTMLDocument)
accessor.getDocument(myApp);
        return doc.getTitle();
    }
});
```

- New Java 6 platform plugin will replace LiveConnect with an improved Java technology/JavaScript bridge

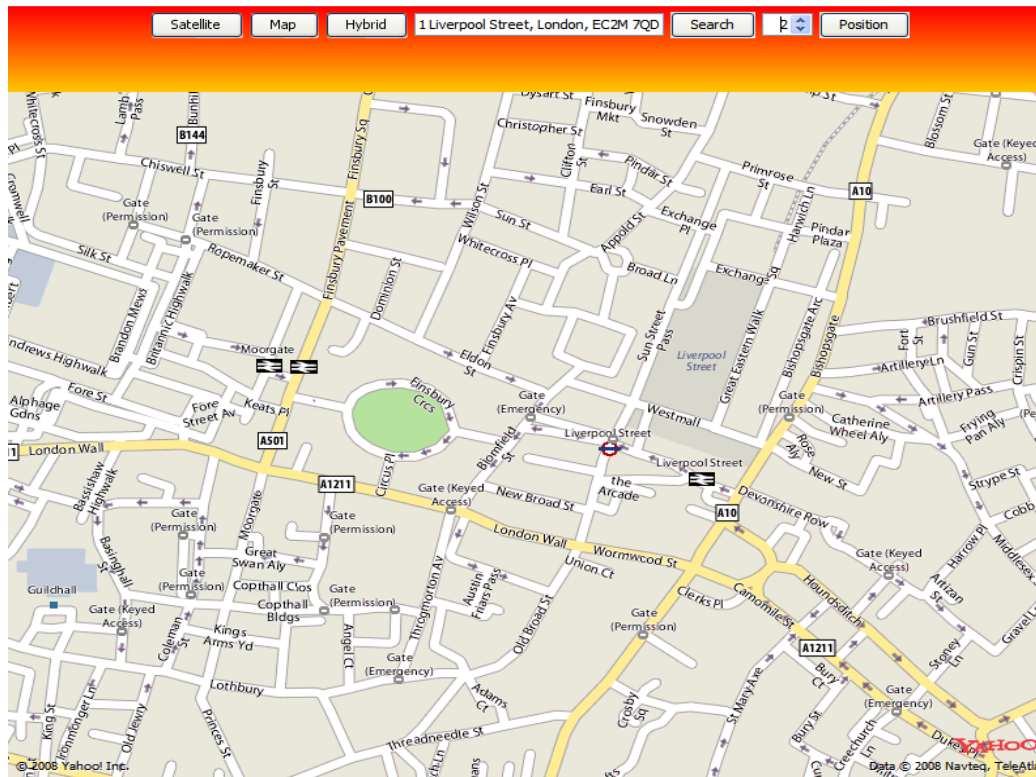
## Example 3 – Applet to Flash

- You can use the LiveConnect classes to call methods on Flash objects too:

```
JSObject w= JSObject.getWindow(this);  
JSObject map = (JSObject) w.getMember("map");  
Object[] param = new Object[] {"1 Liverpool Street,  
    London", "0"};  
map.call("setCenterByAddress", param);
```

# Example 3 – Applet to Flash

## ► Live Example of Applet to Flash (Yahoo Maps)



# Topic: Server-side integration

- Web 2.0 applications typically use:
  - Asynchronous data updates
  - Updating individual GUI components instead of a whole page
  
- Example 4 – Asynchronous data updates
  - Code example
  - Live example
  
- Example 5 – Altio list updates
  - Live example

## Example 4 – Asynchronous data

- Reproducing AJAX's asynchronous data in applets
- Use a separate thread in the applet to either:
  - A) Periodically open a connection to poll the server for data
    - Polling has an overhead of opening a connection, but is useful in environments with limited resources / threads
  - B) Open a persistent connection to receive data updates
    - Better for frequent data updates, as the connection is always available
- Persistent connection:
  - Server writes blocks of data but leaves connection open
  - Client reads blocks of data then updates UI



## Example 4 – Server code

```
public void doGet(HttpServletRequest request,
    HttpServletResponse response) throws IOException,
    ServletException {
    while (true) {
        String sResponse = "<exampleservlet start='" +
            new SimpleDateFormat().format(new Date()) + "'>";
        for (int x=0;x<5;x++) {
            sResponse = sResponse + "<block id='" + x + "'
val='block" + x + "'/>";
        }
        sResponse = sResponse + "</exampleservlet>";
        out.print(sResponse.length() + ",");
        out.print(sResponse);
        out.flush();
        Thread.sleep(3000);
    }
}
```

## Example 4 – Client code

```
while (l==1) {
    StringBuffer l = new StringBuffer();
    int ch = isr.read();
    while (ch != ',') {
        l.append((char)ch);
        ch = isr.read();
    }
    // read X bytes of content
    final int expectedLength =
Integer.valueOf(l.toString()).intValue();
    StringBuffer sb = new StringBuffer(expectedLength);
    for (int x=0;x<expectedLength;x++) {
        sb.append((char)isr.read());
    }
    // Update list contents
    callback.dataArrived(sb.toString());
}
```

## Example 4 – Asynchronous data

- Live example of asynchronous jTable and HTML updates from server

### Example 4 - Applet connections

## Applet

Start updates Updates received: 32

[illegible]

## HTML

Received: 32


```
<exampleservlet start='27/04/08 12:55'><block id='0' val='block0'/><block id='1'  
val='block1'/><block id='2' val='block2'/><block id='3' val='block3'/><block id='4'  
val='block4'/></exampleservlet>  
<exampleservlet start='27/04/08 12:55'><block id='0' val='block0'/><block id='1'  
val='block1'/><block id='2' val='block2'/><block id='3' val='block3'/><block id='4'  
val='block4'/></exampleservlet>  
<exampleservlet start='27/04/08 12:55'><block id='0' val='block0'/><block id='1'  
val='block1'/><block id='2' val='block2'/><block id='3' val='block3'/><block id='4'  
val='block4'/></exampleservlet>  
<exampleservlet start='27/04/08 12:55'><block id='0' val='block0'/><block id='1'  
val='block1'/><block id='2' val='block2'/><block id='3' val='block3'/><block id='4'  
val='block4'/></exampleservlet>  
<exampleservlet start='27/04/08 12:55'><block id='0' val='block0'/><block id='1'  
val='block1'/><block id='2' val='block2'/><block id='3' val='block3'/><block id='4'  
val='block4'/></exampleservlet>  
<exampleservlet start='27/04/08 12:55'><block id='0' val='block0'/><block id='1'  
val='block1'/><block id='2' val='block2'/><block id='3' val='block3'/><block id='4'  
val='block4'/></exampleservlet>  
<exampleservlet start='27/04/08 12:55'><block id='0' val='block0'/><block id='1'  
val='block1'/><block id='2' val='block2'/><block id='3' val='block3'/><block id='4'  
val='block4'/></exampleservlet>
```

## Example 5 – Altio list updates

- Altio has built-in support for polling or streaming data connections
- Data updates are sent to the client applet without any user action
- Individual controls can subscribe to data elements and receive notifications when their data changes
- Individual cells in a list can change font/colour to indicate an update to their data item

# Example 5 – Altio list updates

Conference Schedule



## JavaOne Conference

Room

All

Session

All

Day

All

Time

All

Session ID	Title	Room	Date	Time	Attendees
TS-5453	Is there a place for Applets in Web 2.0?	101	06-May-2008	13:20	50
TS-4817	The Java Platform Portlet Specification 2.0 (JSR...	101	06-May-2008	10:50	300
TS-102	A JavaFX Script Programming Language Tutorial	108	07-May-2008	12:10	120
TS-4794	Glassfish	101	06-May-2008	13:20	100
TS-5425	JAX-RS: The Java API for RESTful Web Services	103	08-May-2008	12:10	500
TS-6609	Using JavaFX Script To Build Swing Applications	104	08-May-2008	13:20	360
TS-5212	GUI Testing Made Easy	104	06-May-2008	18:00	500
TS-7866	Server Side Java™ Technology: What's Next?	105	06-May-2008	18:00	500

# Topic: Applet UI

- Example 6 – Swing controls in an applet
  - Simple code JApplet code example
- Example 7 – JavaFX software in an applet
  - Code example + discussion of issues
- Live example of Swing and JavaFX software in an applet

## Example 6 – Swing in an applet

- Use JApplet then Swing-based applets are simple
  - Add your Swing controls to the applet's root pane

```
public class SimpleSwing extends JApplet {  
    public void start() {  
        super.start();  
        JButton jb = new JButton("Hello");  
        this.getContentPane().add(jb);  
    }  
}
```

- Extra libraries need to be specified in the applet's **ARCHIVE** attribute

## Example 7 – JavaFX Software in an applet

```
ScriptEngineFactory sef = new JavaFXScriptEngineFactory();  
final ScriptEngine engine = sef.getScriptEngine();  
  
// Bind the feed reader to the script's bindings...  
Bindings bindings = engine.createBindings();  
  
// Bind a JComponent ... JavaFX creates a Canvas and adds  
bindings.put("MY_CONTAINER:javafx.swing.JComponent",  
    realControl);  
  
ScriptContext context = new SimpleScriptContext();  
context.setBindings(bindings, ScriptContext.GLOBAL_SCOPE);  
context.setBindings(bindings, ScriptContext.ENGINE_SCOPE);  
engine.setContext(context);  
Object result = engine.eval(toExec);
```



## Example 7 – JavaFX Software in an applet

- Place a JavaFX Canvas into a JComponent to show it:

```
var canvas = Canvas {  
    height: 300  
    width: 300  
    content:  
        Clock {  
            ticking: true  
        }  
};
```

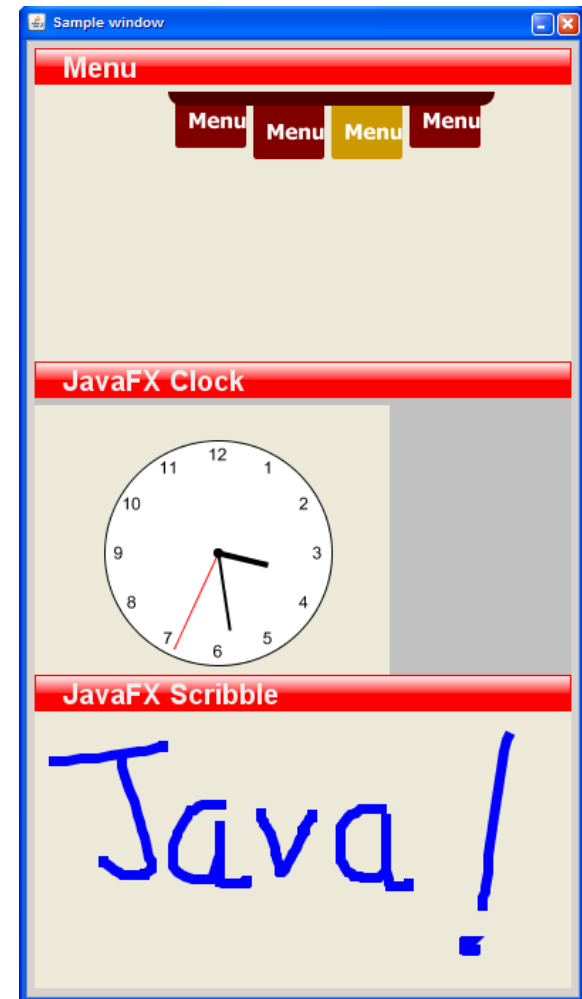
```
MY_CONTAINER: JComponent.add(canvas.getComponent()) ;
```

# Example 7 – JavaFX Software in an applet

- Currently not simple to use JavaFX software in an applet
  - Applet + JavaFX software runtime need to be signed
  - Sets Look and Feel to system default
- Easy use of JavaFX software in applets will be addressed in future Java 6 technology plugin updates
- JavaFX software runtime is a large download, but can be easily included by using JNLP deployment

# Example 6 & 7 – Live demo

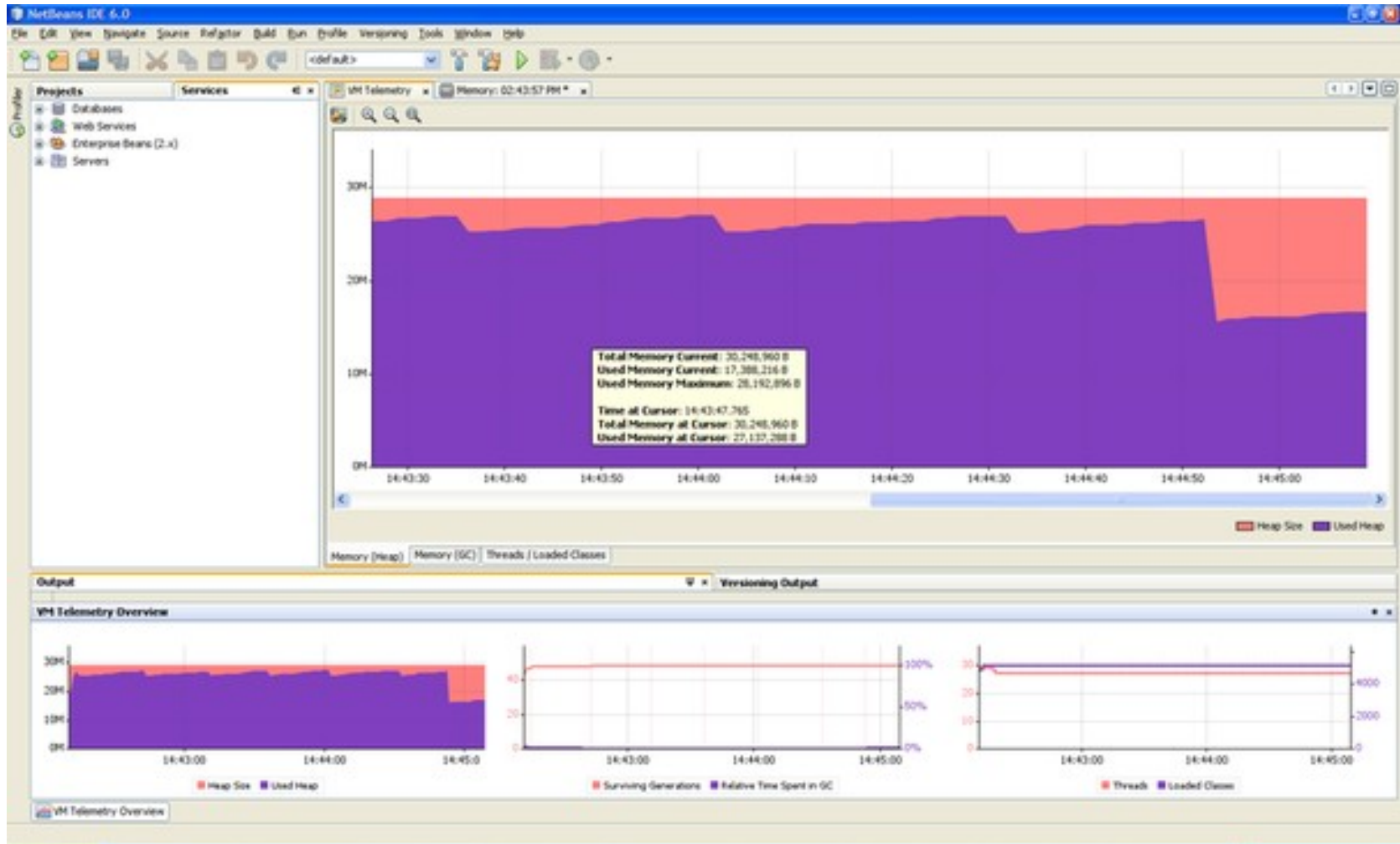
- An Altio application using both Swing and JavaFX software custom controls



# Topic: Applet debugging/profiling

- You can use an development tool to write applets, but debugging/profiling has been neglected in the past
- Need two simple steps:
  - Applet parameters:
    - `-Djava.compiler=NONE -Xnoagent -Xdebug -Xrunjdpw  
:transport=dt_socket,server=y,suspend=n,address=2502`
  - Debug from Eclipse/NetBeans™ software as a remote application on port specified above
- Profiling applets
  - Lots of profiling tools insist on running applets in AppletViewer
  - NetBeans software profiler can be used to easily profile applets

# Netbeans Software Profiler



# Why do Java 6 Technology and JavaFX Software make a difference?

## ➤ Java 6 technology Update 10 Plugin

- Java Kernel for faster download of required components
- Faster startup (Java QuickStart) + graphics performance
- Applets run in separate process to browser, and can run in separate JVMs
- Configurable per-applet JRE version and heap size
- Deployment toolkit can deploy applets with 1 line of JavaScript code

## ➤ JavaFX software

- Script layer on top of Java platform and Swing
- Easier to write graphically rich user interfaces
- Applets make it easy to deploy over the web

# Conclusion

- Previous Applet issues have been addressed:
  - Applets can communicate with servers and other client objects
  - Java 6 technology Update 10 means that applets now have a mature deployment mechanism, not just a delivery mechanism
  - Applets can be as dynamic and rich as any other web application
  
- Future of Applets looks bright:
  - Desktop Swing improvements can be used by Applets
  - JavaFX software makes it simple to create rich Java-based web applications
  - Future JRE updates should add to the plugin enhancements
  - JavaFX software + Applets + cross-platform = compelling RIA solution

# Cool Applets

- <http://coolapplets.blogspot.com/>
- Just2Easy (Applet Word Processing):
  - <http://www.j2e.com/>
- Iris (AJAX + Applets for Flickr UI):
  - <http://swinglabs.java.sun.com/iris/>
- Map 24 (Interactive maps):
  - <http://www.uk.map24.com/>
- UpNext (NYC Map/Social):
  - <http://www.upnext.com>



# For More Information

➤ Altio:

- <http://www.altio.com>

➤ Applets/JavaScript:

- <http://java.sun.com/javase/6/docs/technotes/guides/plugin/develop>

➤ JavaFX Community:

- <https://openjfx.dev.java.net/>

➤ New Java plugin:

- <https://jdk6.dev.java.net/plugin2/>

# THANK YOU

Jim Crossley & Gary Thompson, Altio  
Ltd

TS-54533

