

Technologies For Creating Rich Internet Applications

Presenter's name
emailaddress@Sun.com

Agenda

- **Introduction**
- RIA Technologies
- Using AJAX for RIA
 - jMaki
 - Wicket
 - GWT (Google Web Toolkit)
 - JavaServer Faces
- Using JavaFX for RIA
- Conclusion

Introduction

- **We have been here before**
 - Browser interface = “Dumb” terminal
 - RIA = X11
- **What is new this time?**
 - The browser paradigm – simplicity
 - Word Wide Web
- **Pushing the web to the next level - Web 2.0**

Introduction



Introduction

- **Typical RIA characteristics:**
 - Rich UI experience
 - Provide desktop application like functionality
 - Do not require software installation
 - Execute on the client-side launched from the browser (Javascript, Applet, Flash, etc.)

Introduction

- <http://www.sys-con.com>
- <http://www.official-linerider.com/pla>

Introduction

- **Advantages of RIA**

- Better end-user experience
 - UI more responsive
 - UI techniques easier to work with
- Processing offloaded to the client
- Less network load

Introduction

- **RIA Pitfalls**

- Affect on client machine
- Security
- Search Engine Optimization
- Simplicity made complex
- Difficult to implement

Agenda

- Introduction
- **RIA Technologies**
- Using AJAX for RIA
 - jMaki
 - Wicket
 - GWT (Google Web Toolkit)
 - JavaServer Faces
- Using JavaFX for RIA
- Conclusion

RIA Technologies

Currently two main RIA categories

RIA Technologies

- **Browser centric**
 - Live “IN” the browser
 - Standards based, Javascript, XML, HTTP
 - Developers use current skill set
 - Limited by browser capabilities

RIA Technologies

- **Browser launched**
 - Container based (e.g. Flash)
 - Typically proprietary
 - Requires specialized skill set
 - Not limited by browser capabilities

RIA Technologies

- **Browser Centric**
 - ActiveX Controls
 - Internet Explorer centric
 - Security issues
 - Provides seamless rich content

RIA Technologies

- **Browser Centric**

- Asynchronous Javascript and XML (AJAX)
 - New use of existing technologies
 - Client-side frameworks:
 - jMaki
 - Dojo
 - Prototype
 - Script.aculo.us
 - ...

RIA Technologies

- **Browser Centric**

- Asynchronous Javascript and XML (AJAX)
 - Server-side frameworks:
 - DWR – Direct Web Remoting
 - Openlaszo – multi-runtime framework
 - GWT -Google Web Toolkit

RIA Technologies

- **Browser Centric**

- Asynchronous Javascript and XML (AJAX)

- Existing frameworks adopting AJAX:
 - JavaServer Faces (JSF) – Dynamic Faces
 - Rails – (Ruby on Rails)
 - Wicket
 - ...

RIA Technologies

- **Browser Launched**
 - Adobe Flash/Flex
 - Flash is the runtime environment
 - Flex is a RIA framework
 - Actionscript and MXML (Magic XML)

RIA Technologies

- **Browser Launched**
 - Java Applets
 - Original vision was RIA
 - Java Applications
 - Java Web Start

RIA Technologies

- **Browser Launched**
 - JavaFX
 - JavaFX scripting
 - Compiled to byte code
 - Consumer Runtime Environment
 - Microsoft Silverlight
 - Based on Windows Presentation Framework (.Net 3.0)

Agenda

- Introduction
- RIA Technologies
- **Using AJAX for RIA**
 - jMaki
 - Wicket
 - GWT (Google Web Toolkit)
 - JavaServer Faces
- Using JavaFX for RIA
- Conclusion

Using AJAX For RIA

- **jMaki**
 - “AJAX in a tag”
 - Widget Model – leverages existing widgets
 - Layouts
 - Server Model

Demo

- jMaki demo

Using AJAX For RIA

- **Wicket**

- AJAX without Javascript and XML
- Makes web app development easy for Java Developer
- Easy to separate web app responsibilities
- POJO data model

Demo

- Wicket demo

Using AJAX For RIA

- **Google Web Toolkit (GWT)**
 - Takes the “J” out of AJAX
 - Manage “Back” button history
 - Compiles to Javascript
 - Supports Java source compatible with J2SE 1.4.2 or earlier
 - Google API Support: Google Gears

Demo

- GWT demo using GWT4NB plugin

Using AJAX For RIA

- **JavaServer Faces**
 - Component Based
 - Project Woodstock (Sun components)
 - AJAX details hidden
 - Dynamic Faces (DynaFaces)

Using AJAX For RIA

- **DynaFaces**

- Incremental improvement to JSF 1.2
- Extends JSF life cycle
- Developer entry points:
 - Page – AJAX components, ajaxZone, Javascript library
 - Components – embed ajaxZones and Javascripting

Demo

- JavaServer Faces and DynaFaces demo

Agenda

- Introduction
- RIA Technologies
- Using AJAX for RIA
 - jMaki
 - GWT – Google Web Toolkit
 - JavaServer Faces
- **Using JavaFX for RIA**
- Conclusion

Using JavaFX For RIA

- **JavaFX Platform**
 - JavaFX script
 - JavaFX Mobile

Using JavaFX For RIA

- **JavaFX Script**

- For creating rich UIs using:
 - Swing
 - Java 2D
 - Java 3D
- Statically typed
- Java APIs available
- Declarative UI development

Using JavaFX For RIA



Using JavaFX For RIA

- **Currently uses Java Web Start**
- **Future – Consumer Java Runtime Environment**
 - Faster startup
 - Easier install
 - Smaller footprint
 - Goal is to be “Flash-like” experience

Using JavaFX For RIA

- **Currently uses Java Web Start**
- **Future – Java SE 6 Update N**
(<https://jdk6.dev.java.net/6uNea.html>)
 - Faster startup
 - Easier install
 - Hardware acceleration
 - New “Nimbus” look and feel
 - Goal is to be “Flash-like” experience

Demo

- JavaFX script demo using JavaFX plugin

Agenda

- Introduction
- RIA Technologies
- Using AJAX for RIA
 - jMaki
 - GWT – Google Web Toolkit
 - JavaServer Faces
- Using JavaFX for RIA
- **Conclusion**

Conclusion

- **Adobe and Microsoft – \$\$\$, proprietary, good tooling**
- **JavaFX – great potential, look for more tooling**
- **AJAX – clearly has the most momentum, utilizes browser-based technologies, great tool support**

Technologies For Creating Rich Internet Applications

Presenter's name
emailaddress@Sun.com