



# JavaFX: The Second Coming of Java to the Web

Simon Rlitter  
Technology Evangelist  
Sun Microsystems



# Java Technology Momentum



Java™



## Java Everywhere

---

4.5 Billion Java-Enabled Devices

---

1.8 Billion Java-Enabled Phones

---

7 Million Java Set-top Boxes

---

800 Million Java Desktops

---

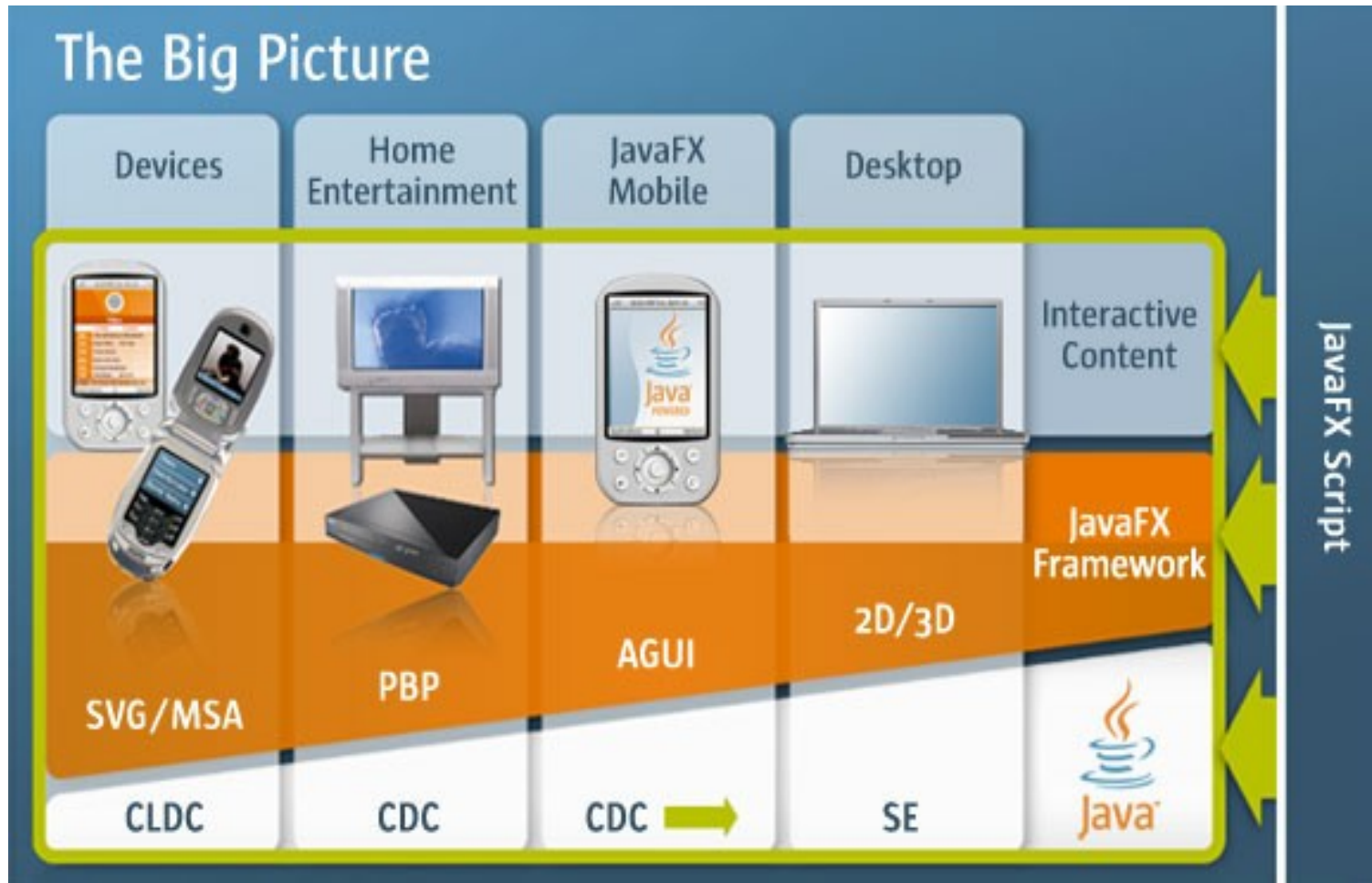
180 Operators Deploying  
Java Content

---

5 Million Developers

---

# JavaFX Overview



# JavaFX Components

- JavaFX script
- JavaFX Mobile
- JavaFX TV
- Java SE 6 update n (Consumer JRE)
- Designer tools
- Developer tools

# Java Mobile Edition (ME) Challenges

- Device fragmentation
  - > MSA attempts to address this
- Limited native device access
  - > Security is a major consideration
- APIs are different to Java SE
  - > By virtue of different device format
- Original design targeted low power devices
  - > Phones today more like PCs five years ago

# JavaFX Mobile Motivation

- Personal and mobile computing are converging
  - > Moore's law
  - > Higher speed networks, Wi-Fi
- The platform is becoming more important
  - > Consistency for development
  - > Reduced development time
- The industry is looking for an open source software stack

# The Network in Your Hand

## JavaFX Mobile



- To enable consumer interaction wherever, whenever and however
- Advanced phone OS built with:
  - > The power of Java SE desktop software
  - > The acceptance of the linux kernel
  - > Tuned for Web and Network Services integration
  - > Compelling User Experience
  - > Rich authoring tools
  - > Highly portable architecture

# JavaFX Mobile Architecture

## FRAMEWORKS

- Application APIs
- User Interface Toolkit
- Application Manager
- Advanced Graphics Engine
- Telephony Framework
- Multimedia Framework
- Software Update
- Security Framework
- System Libraries
- Java Virtual Machine



## PHONE APPLICATIONS

- Messaging
- Browser
- PIM and Phone Apps
- Media Player

## NATIVE OS

- Low-Level Services and Libraries
- linux kernel

# The JavaFX Mobile OS Advantage

## Advantages

- Portability
- Time to market
- Quality
- Single Application Community
- Feature rich
- User Experience
- High Performance
- Security



## Operator Opportunities

- Resident Apps
- User Experience
- Operator Dynamic Update

# Optimize and pre-integrate key services

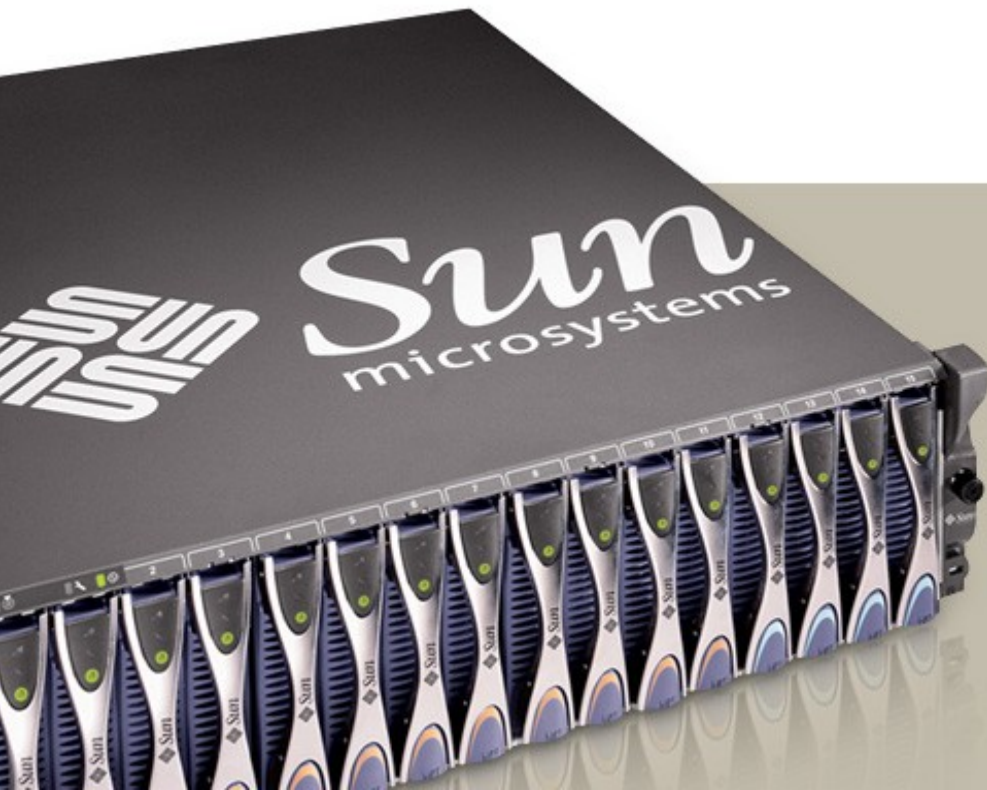


Resident Content Drives ARPU

# JavaFX Mobile and JavaME

- JavaFX Mobile is a complete desktop scale OS for mobile computing devices
- Java ME does not disappear
  - > JavaFX mobile will run Java ME code
- New apps can be written using Java ME, Java SE or JavaFX script

# JavaFX Script



# JavaFX Script: Design Questions

- Why does it take a long time to write GUI programs?
- How can we avoid the “Ugly Java GUI” stereotype?
- Why do Flash programs look different than Java programs?
- Why does it seem easier to write web-apps than Swing programs?
- And how can I avoid having an enormous, writhing mass of listener patterns?

# A Basic Java GUI: Not Very Pretty



# The “Ugly Java GUI” Stereotype

- AWT/Swing Container/Component Hierarchy
  - > A tree of rectangular (mostly grey) boxes
  - > If all you do is compose Swing components together, the result is typically “the Ugly Java GUI”
  - > Same problem exists with other toolkits, e.g., GTK, VB
- UI Designers and Swing programmers are using different building blocks
  - > UI Designers compose designs in tools like Photoshop and Illustrator
  - > The building blocks they use have direct analogs in Java 2D, but not always directly in Swing

# Java 2D API

- To match the designs of UI designers requires using Java 2D API
- Java 2D API doesn't have compositional behavior
  - > Makes it too complex for many programmers to use efficiently
- In addition to Swing Components, JavaFX includes SVG-like interfaces to Java 2D API as first-class elements which can be composed together into higher-level components
- FX allows declarative expression of this composition

# JavaFX Script: Why Scripting?

- Java Language != Java Platform
  - > VM runs “language-neutral” bytecode
  - > Rich set of Class libraries are “language-neutral”
  - > “**Write once run anywhere**” applies to Platform
  - > Leverage programmer skills and advantages of particular languages.
- Time-tested technologies
  - > Open-source projects for various languages
  - > Jakarta BSF

# Yet Another Scripting Language?

- Leverage the power of Java's 'write once, run anywhere' capability
  - > Cross browser and cross device
- Optimize the creative process of building rich and compelling UIs
  - > Swing, Java 2D and Java 3D
  - > Suitable for developers and content authors
- Structure of JavaFX code closely matches the actual layout of the GUI
  - > Enhanced readability and maintainability

# Developer Productivity

- Streamline the Edit-Compile-Debug Cycle
  - > Hot swapping (Lisp), Dynamic Typing, Type Interference and so on
  - > Instant feedback is a requirement, not a “nice to have”
  - > More Expressive and more flexible
- However
  - > Static type checking and compilation errors are good
  - > Strong typing leads to safe code

# JavaFX Script Language Basics

- A proper subset of core Java APIs
  - > Programming Language for the Java platform
  - > Object-oriented
  - > Declarative Syntax
  - > Statically-typed with type-inference
  - > Automatic data binding
  - > Extensive Widget library encompassing Swing and Java 2D API
  - > Development tools including NetBeans and Eclipse IDE plugins

# Static V. Dynamic Typing

- High-quality, compile-time error reporting
- High-quality IDE support
  - > Code-completion
  - > Searching
  - > Refactoring
- Efficient compilation

# JavaFX Data Binding

- Cause and Effect - Responding to change
- bind operator
  - > Allows dynamic content to be expressed declaratively
- Dependency-based evaluation of any expression
- Automated by the system
  - > Rather than manually wired by the programmer
- Simply declare dependencies
  - > The JavaFX runtime takes care of performing updates when things change
- Eliminates listener patterns

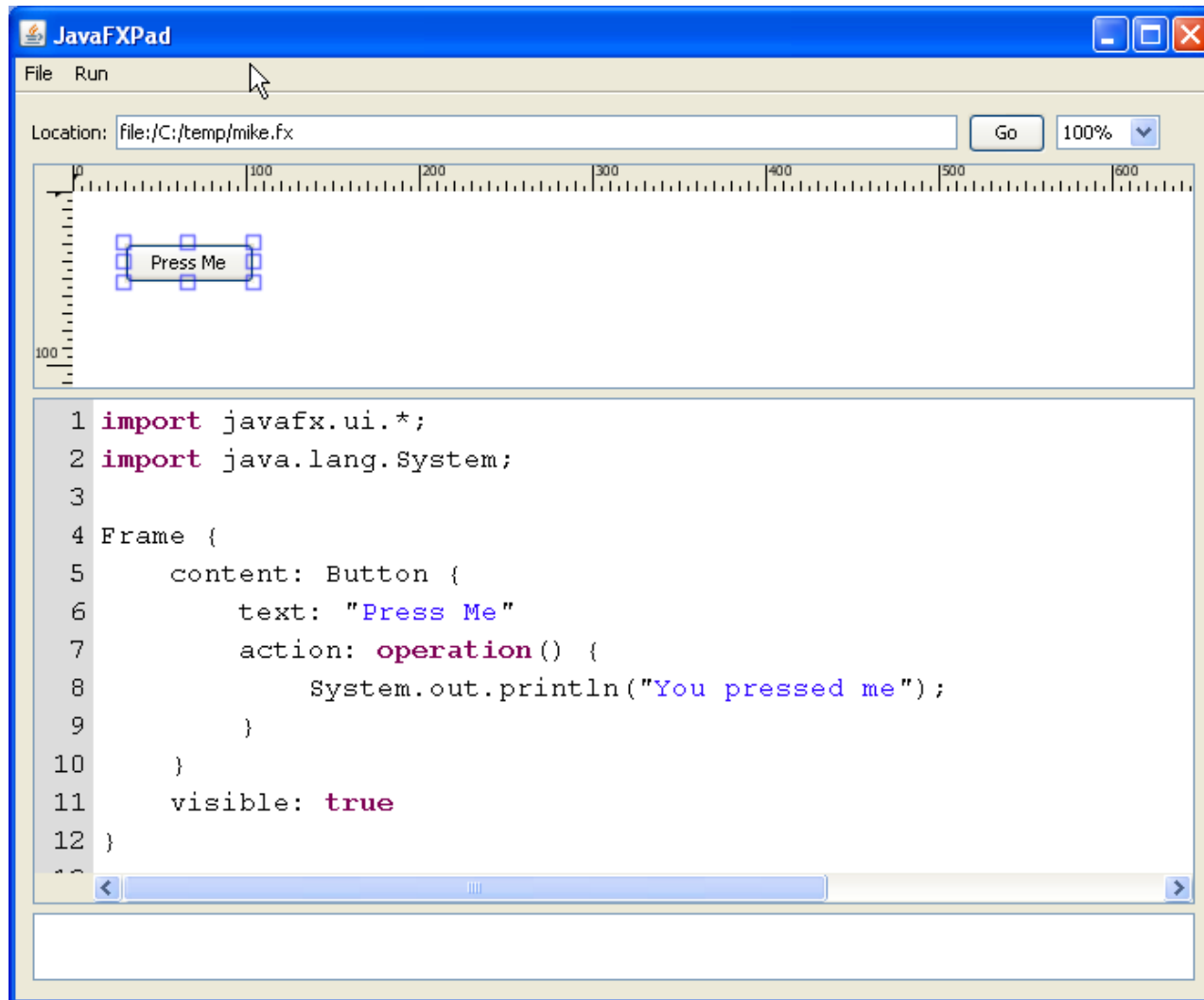
# Benefits of Declarative Syntax

- You can see it in Web applications
- For example, ease of composing styled text
  - > HTML versus JTextPane
- HTML Table using JSTL versus JTable
- JavaFX brings that same ease of use to Swing and Java 2D programming

# Tooling: NetBeans Plugin

- Understand the Edit-Compile-Debug lifecycle for JavaFX programs on NetBeans
- Run JavaFX programs in NetBeans easily with the JavaFX shell

# Tooling: JavaFXPad



# JavaFX Script Deployment

- JavaFX Script Runtime
  - > 1.5 MB jars (700 kb with pack200)
- JavaFX Script Deployment the same as Java
  - > JavaFX Script files are archived in Jar files and loaded via the Java class loader
  - > Standalone Java Application
  - > Java Web Start
  - > Applet

# Invoking JavaFX From Java Apps

- Use FXShell
  - > Use simple wrapper to call the main() method of the FXShell class
  - > FXShell is part of the JavaFX jars
  - > Not officially supported
- Use JSR-223
  - > Invoke JavaFX scripting engine through standard APIs
  - > Requires JDK 6 or JSR-223 support
  - > More flexible

# Future

- JavaFX Script compiler
- Tools to support JavaFX development. Plug-ins are available for NetBeans 5.5 and 6.0, and Eclipse 3.2
- Making JavaFX Script runnable on Java based mobile devices and set-top boxes
- Integrate with consumer JRE
- A JavaFX Script painter is available from ReportMill, <http://www.reportmill.com/jfx>

# Consumer JRE

- Quickstarter
- Java Kernel
- Deployment toolkit
- Installer improvements
- Graphics performance
  - > Windows
- Nimbus
  - > New cross-platform look-and-feel

# Summary

- JavaFX is a family of products and technologies aimed at content creators
- JavaFX Mobile
  - > Linux based full stack using Java SE
- JavaFX script simplifies GUI programming
  - > Let the graphic artists do the hard work
- More coming, watch this space
  - > Better tools (more drag'n'drop)
  - > Ease of deployment
  - > Consumer, modular JRE

# Further Information

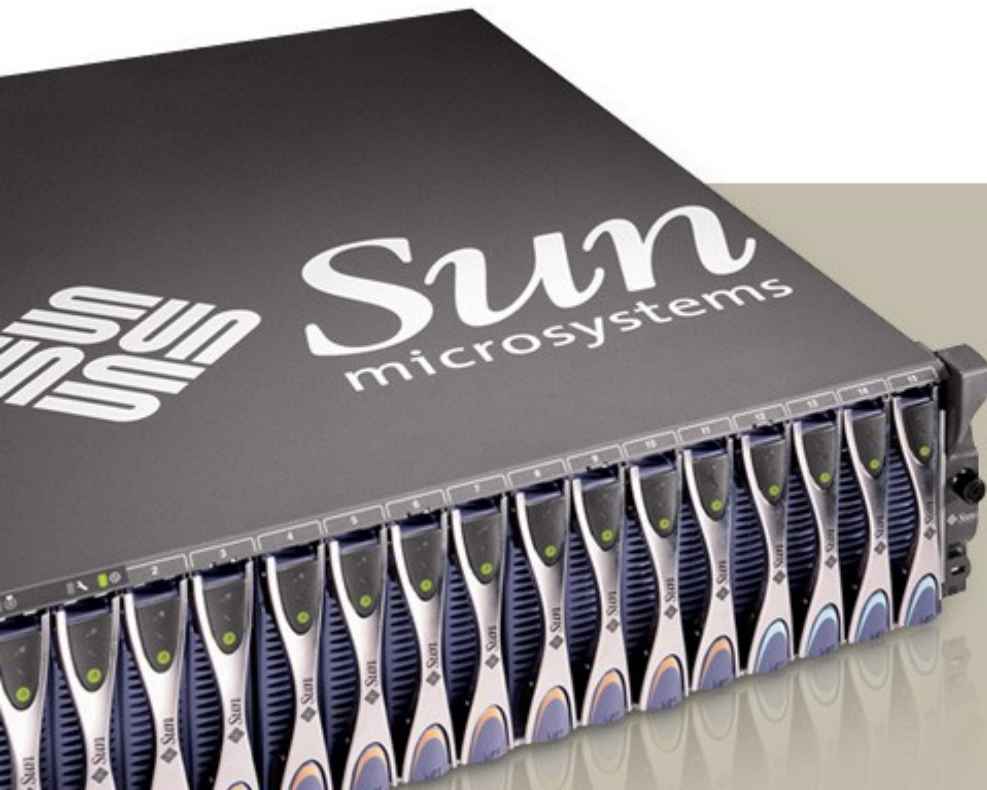
<http://www.sun.com/javafx>

<http://openjfx.org>

<http://blogs.sun.com/chrisoliver>

<http://evc-cit.info/jfx/makeapi/api/index.html>

# Demos





# JavaFX: The Second Coming of Java to the Web

Simon Rlitter  
Technology Evangelist  
Sun Microsystems

