

# What's in Java SE 7

Or... Exciting Stuff coming in Java SE

Danny Coward  
Java SE Platform Lead  
Sun Microsystems  
2101

**JAZOON07**

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY  
JUNE 24 - 28, 2007 ZURICH



# What I will cover

A broad (but not always deep) view of what's in the future of Java SE

# AGENDA

Java SE today

Creating Java SE

JavaFX and consumer content

Supporting multiple languages

Modularity in the platform

Some important upgrades

# AGENDA

Java SE today

Creating Java SE

JavaFX and consumer content

Supporting multiple languages

Modularity in the platform

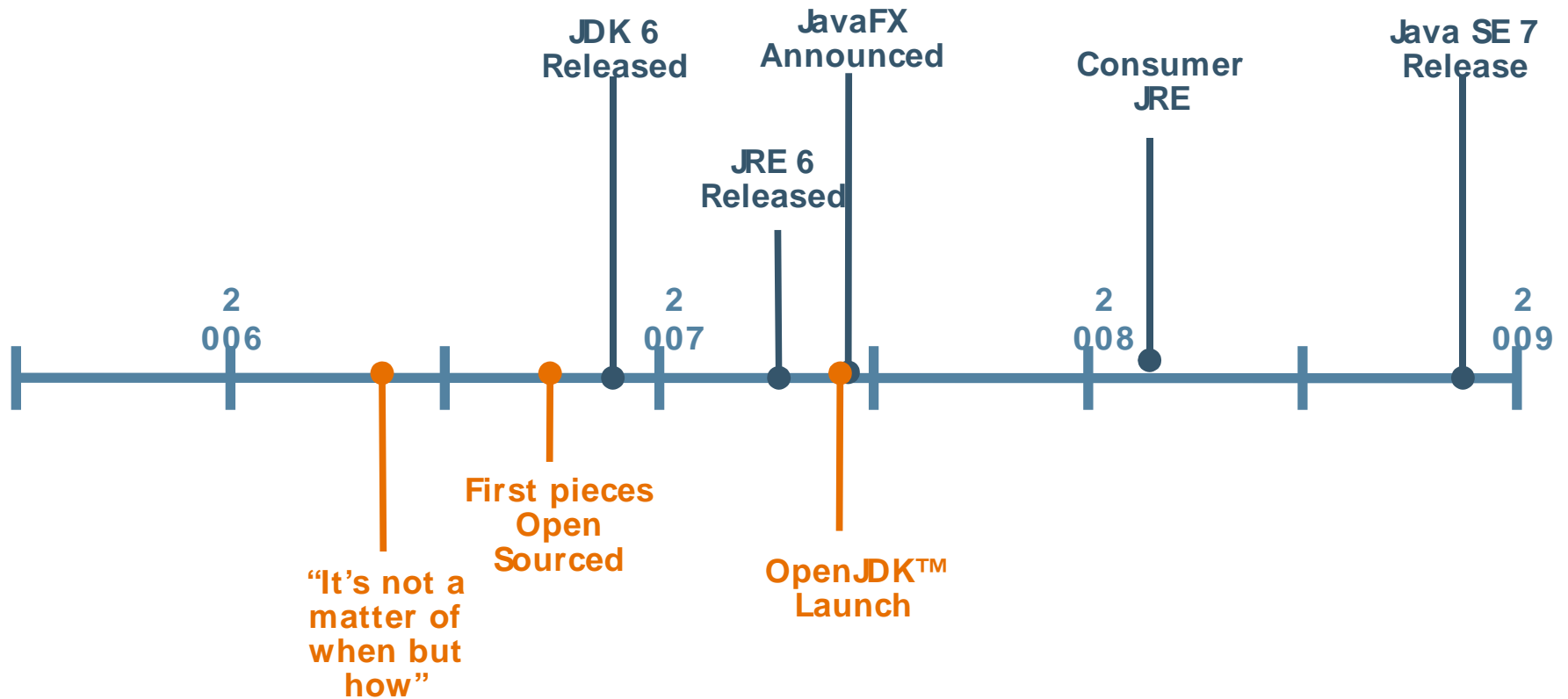
Some important upgrades

**JAZOON07**

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY  
JUNE 24 - 28, 2007 ZURICH



# Java SE Platform Roadmap



**JAZZ00N07**

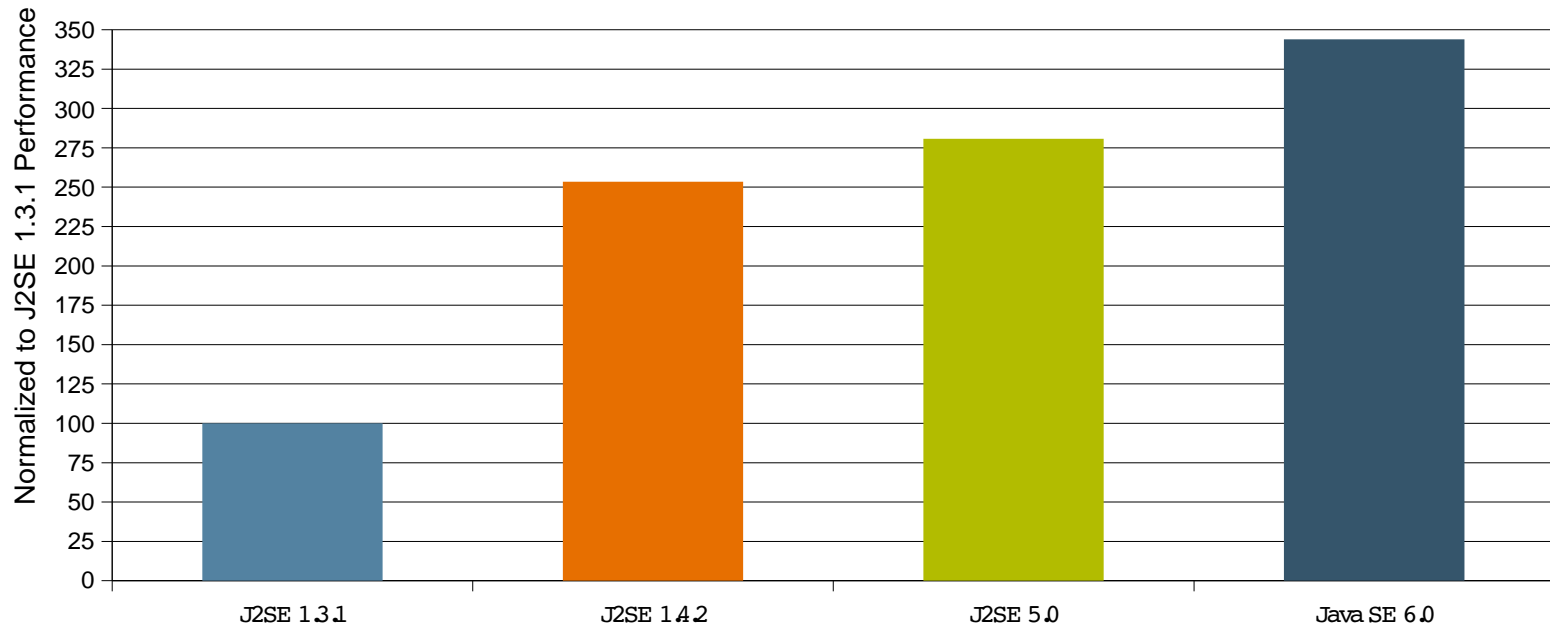
JDK = Java Development Kit (JDK™) | JRE = Java Runtime Environment (JRE™)  
 THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY  
 JUNE 24 - 28, 2007 ZÜRICH



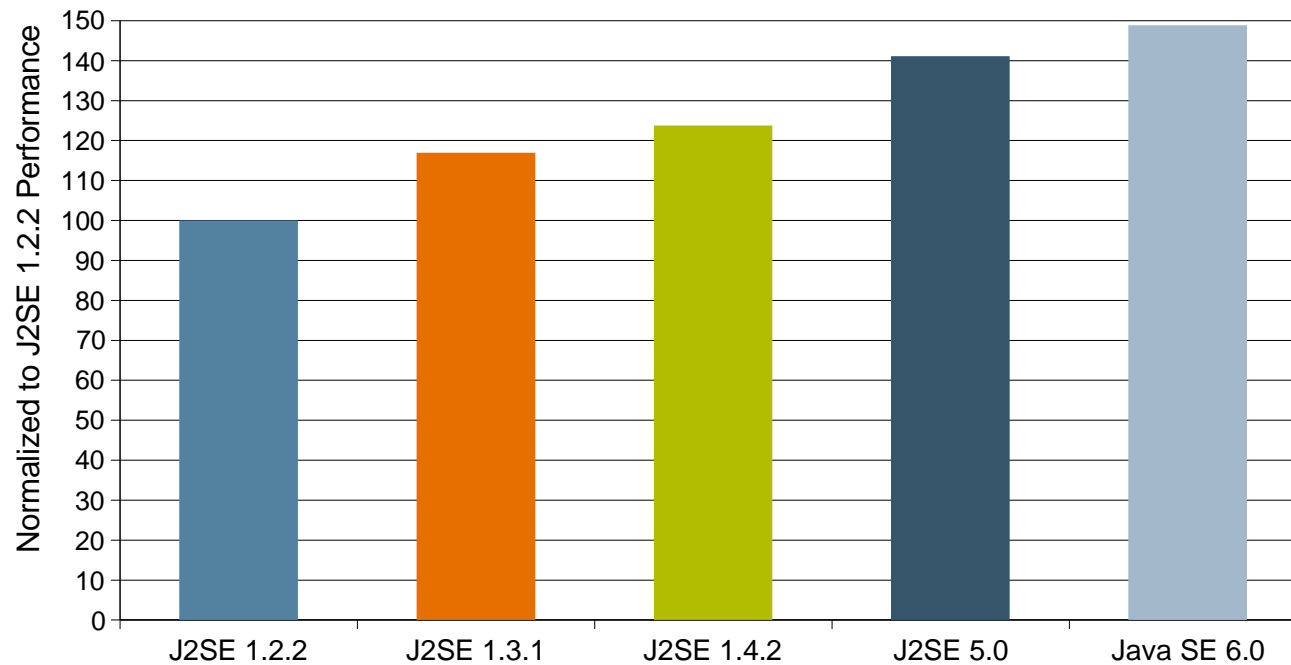
# Java SE 6 Platform: Top 10 Features

<b>Web Services</b>	<b>Easy to use Web Service APIs</b>
<b>Scripting</b>	<b>Ability to mix JavaScript™ technology with Java c</b>
<b>Database</b>	<b>Updated JDBC APIs, all-Java database in JDK</b>
<b>More Desktop APIs</b>	<b>Swingworker, JTable sorting and filtering, GroupLayout and more</b>
<b>Monitoring and Management</b>	<b>Attach on demand</b>
<b>Compiler Access</b>	<b>APIs to control the compiler</b>
<b>Pluggable Annotations</b>	<b>Define your own annotations</b>
<b>Desktop Deployment</b>	<b>Better Swing fidelity, tuned for Vista</b>
<b>Security</b>	<b>Integration with native services</b>
<b>The Ilities: Quality, Compatibility, Stability</b>	<b>Faster more reliable and backwards compatible</b>

# Better Performing on the Server

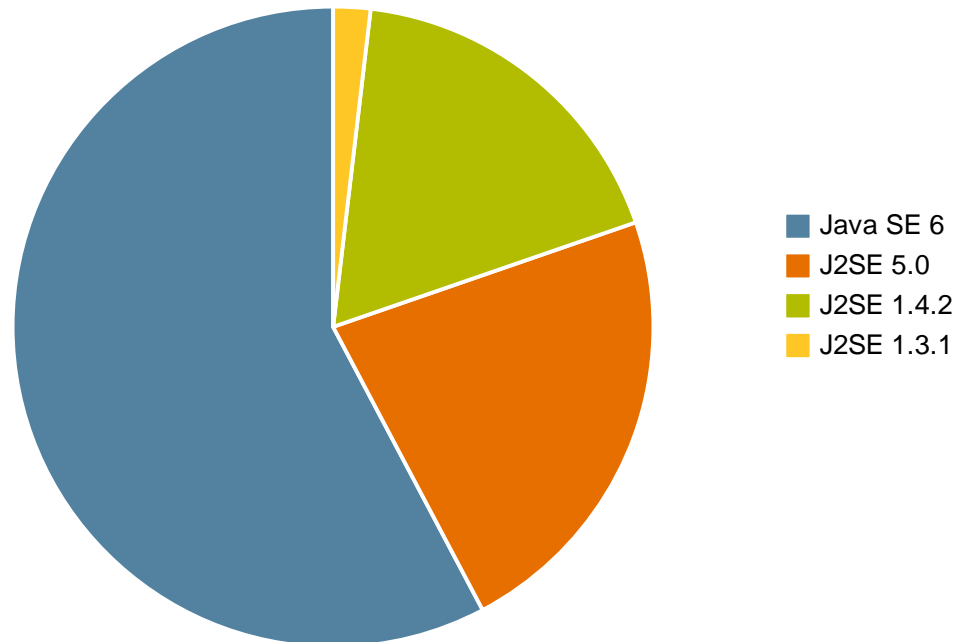


# Better Performing on the Desktop



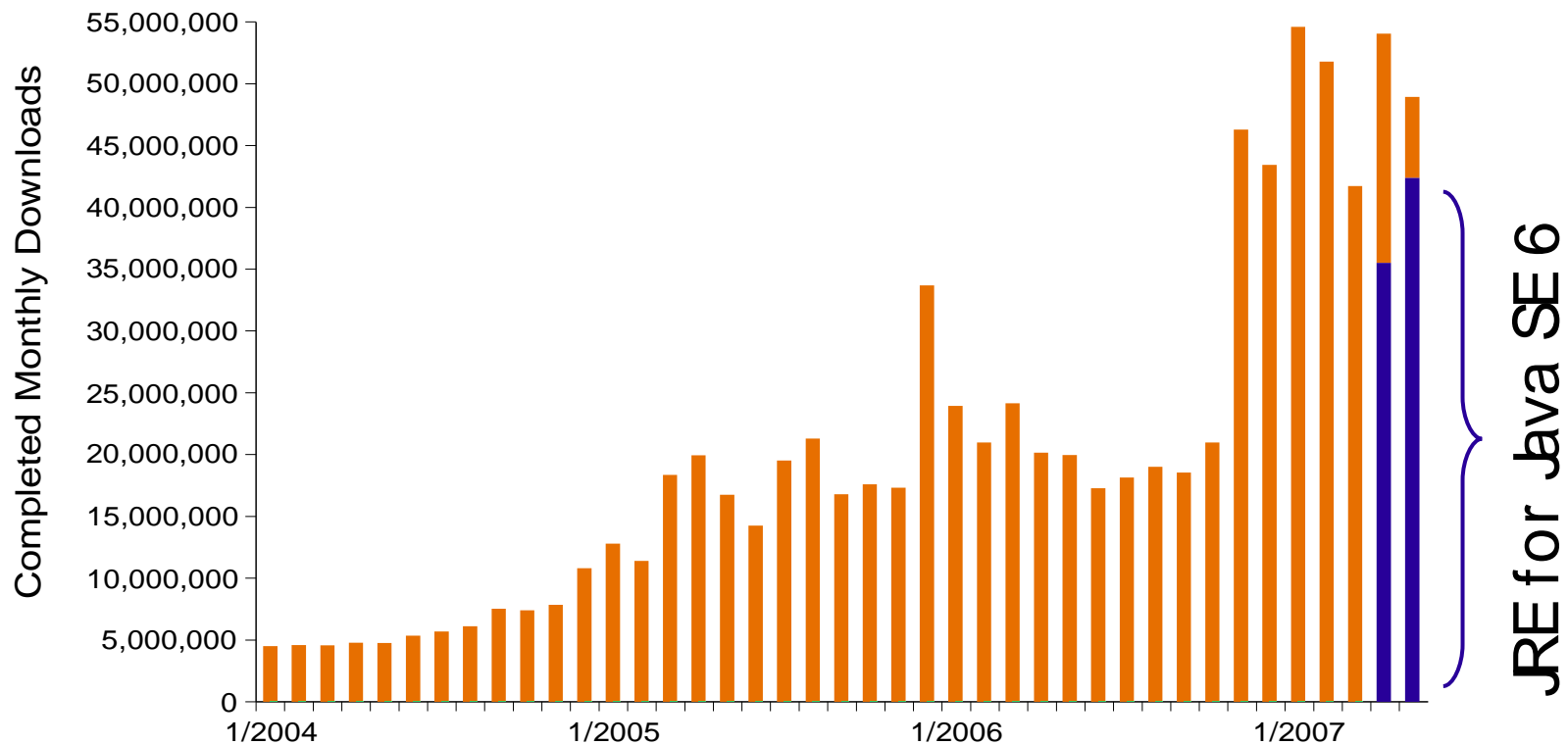
# JDK Version 6 Adoption

2,746,523 downloads of the JDK Version 6. Already!



# JRE Software Adoption

78M downloads of the JRE Version 6. Already!



## Java SE Platform on the Desktop

- > 60% of new PCs will have Java SE platform pre-installed
- > 91% of all PCs run Java platform
- > That's 540 million PCs worldwide
- > 78 million of them upgraded to the Java SE 6 platform since the release in April

**Have you upgraded to the Java SE 6 platform yet?**

# AGENDA

Java SE today

Creating Java SE

JavaFX and consumer content

Supporting multiple languages

Modularity in the platform

Some important upgrades

**JAZOON07**

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY  
JUNE 24 - 28, 2007 ZURICH



## Java SE Platform Is Open Sourced

- > Fully buildable open source JDK software
- > Extension of the Peabody experience
- > GPLv2 license, encouraging
  - Compatible innovations
  - Wider distribution
- > OpenJDK community
  - [openjdk.java.net](http://openjdk.java.net)
- > Interim governance board
  - Charter to build OpenJDK constitution

# How Java SE Platform Gets Developed



## OpenJDK

What	API Specifications	JDK Implementation
Where	<a href="http://jcp.org">jcp.org</a>	<a href="http://openjdk.java.net">openjdk.java.net</a>
How	Expert Groups	OpenJDK projects
When	Now	Now !

# Java SE and Consumer Content

- > Why refocus on consumer content now ?
  - More & more businesses want direct relationships with their consumers (through their access devices)
  - Others have noticed...
  
- > General observations
  - Consumer content is not just created by Java developers
  - Swing is powerful but complex
  - The runtime environment is too heavy

# AGENDA

Java SE today

Creating Java SE

JavaFX and consumer content

Supporting multiple languages

Modularity in the platform

Some important upgrades

**JAZOON07**

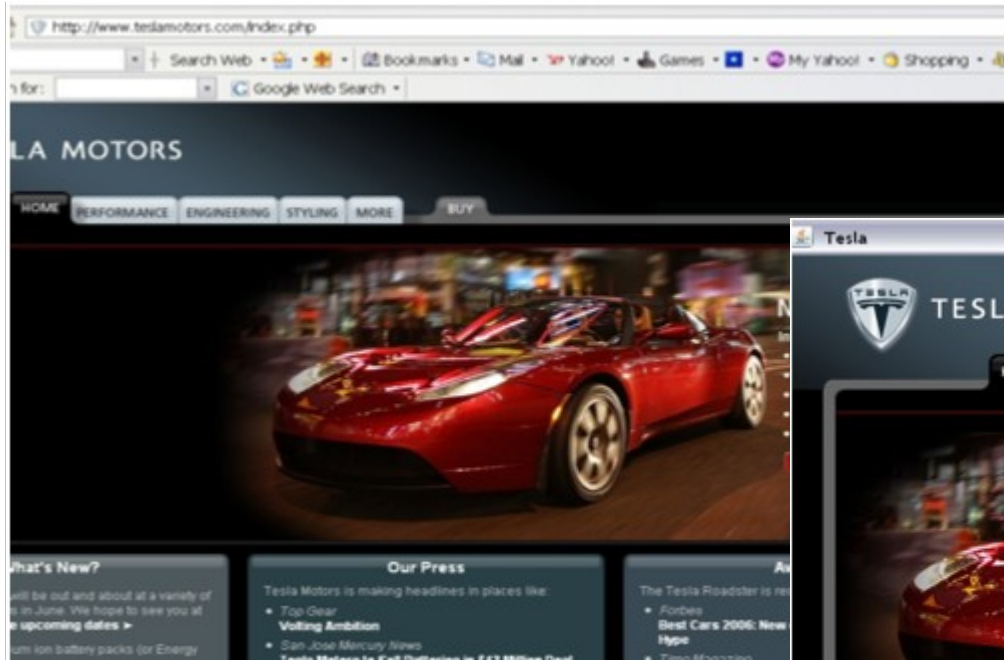
THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY  
JUNE 24 - 28, 2007 ZURICH



# JavaFX Script

- > New programming language for the Java platform
  - Aimed at content creators / mashups
  - Object-oriented
  - Declarative Syntax
  - Statically-typing and type-inference
  - Extensive widget library encompassing Swing and Java 2D API

# JavaFX Script Applications



teslamotors.com ...

... written in JavaFX



# Hello JavaFX

```
import javafx.ui.*;
Frame {
    title: "Hello World JavaFX"
    width: 200
    height: 50
    content: Label {
        text: "Hello World"
    }
    visible: true
}
```



# Using Swing and Java

## StringOpThread class

```

.....
public void run() {
    opacityBegin = 0.01;
    opacityEnd = 1.0;
    opacityIncreStep = 0.02;
    opacitySleep = 2;
    for(currOpacity = opacityBegin;
        currOpacity < opacityEnd;
        currOpacity+=opacityIncreStep) {
        setPickOpacity();
        repaint();
        try {
            thread.sleep(opacitySleep);
        } catch (InterruptedException
e) {}
    }
}

```

## Main guitar class ...

```

guitarAnimationThread = new StringOpThread();
.....
.....
if (guitarAnimationThread != null) {
    guitarAnimationThread.run();
}
.....

```

# Using JavaFX Script

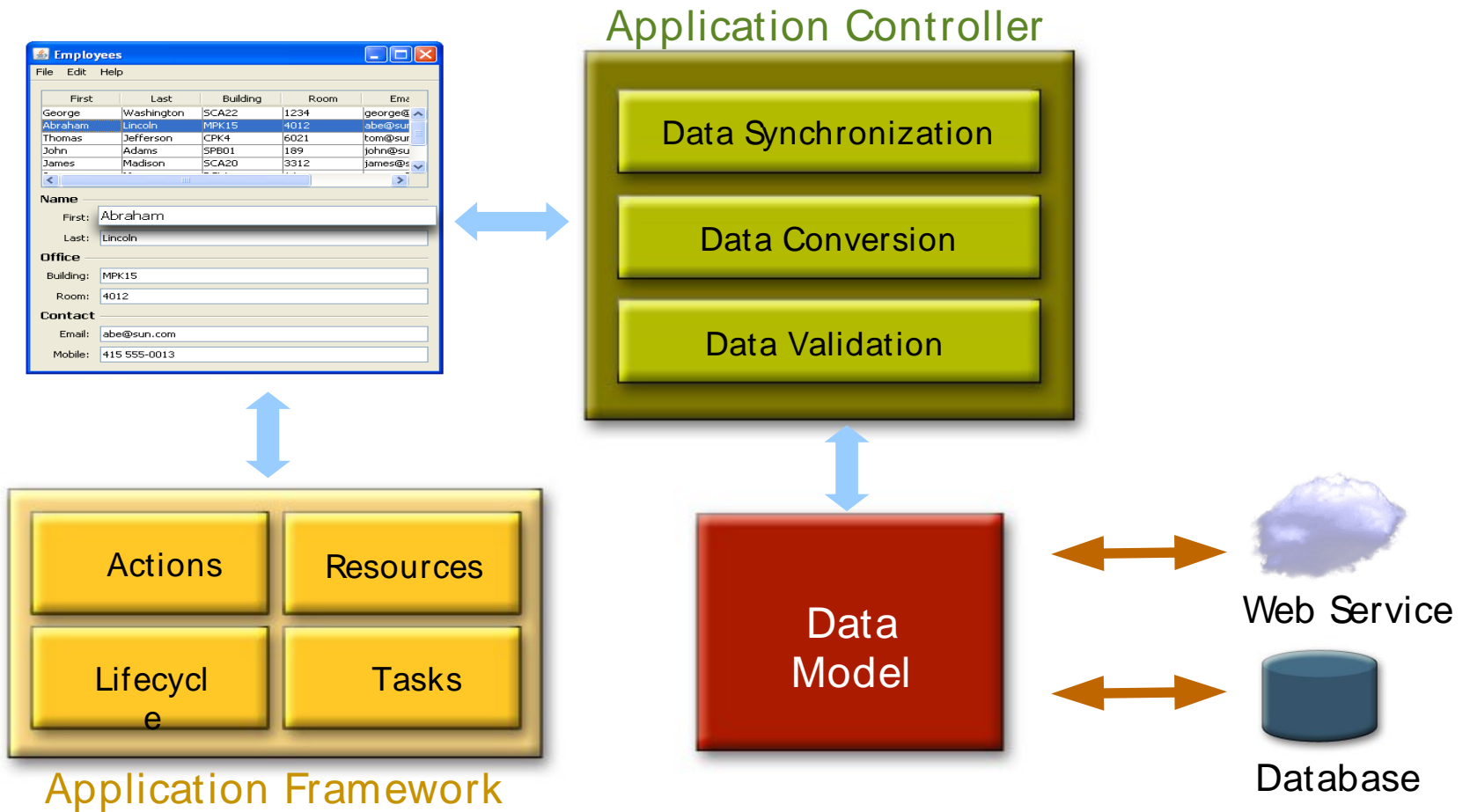
```
picks.opacity = [0, .01 .. 1 ] dur 1000 linear
```

## JavaFX Script Today

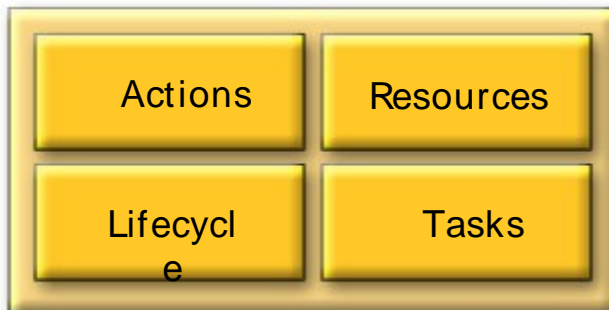
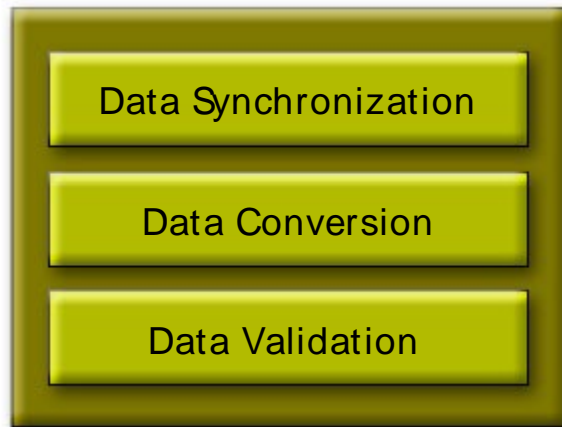
- > Go to [openjfx.dev.java.net](http://openjfx.dev.java.net)
  - Download runtime
  - Download EA plugins for NetBeans and Eclipse
- > Lots of documentation
  - [jfx.wikia.com](http://jfx.wikia.com)

Much, much more to come !

# A Swing Application

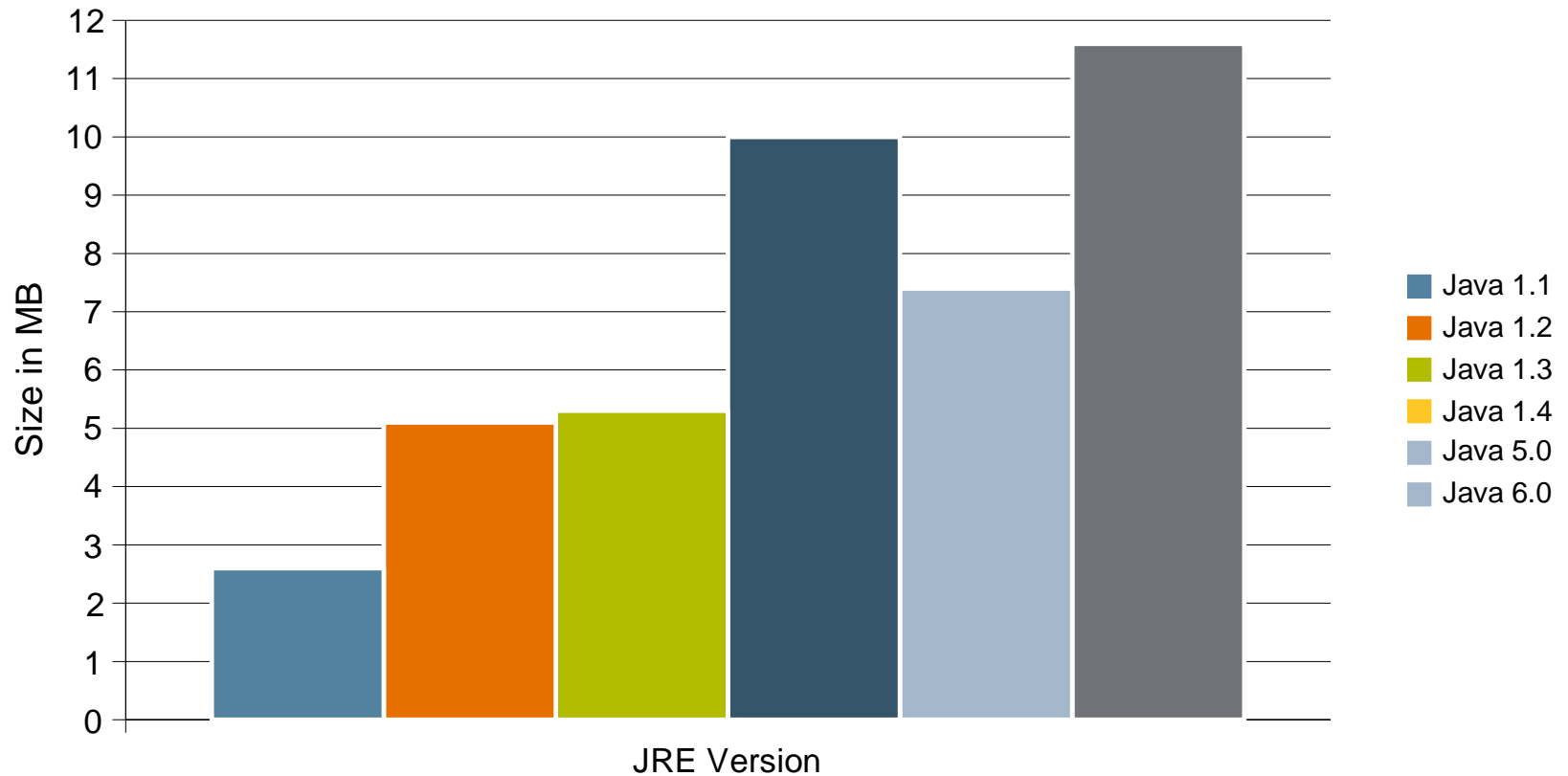


# A Swing Application



- **JSR 295: Beans Binding**  
Formalizing the Swing controller API support for synchronizing JavaBeans
- **JSR 303: Beans Validation**  
Metadata model for expressing validation constraints
- **JSR 296: Swing Application Framework**  
Formalizing support functions

# JRE Software Download Size



## JRE as Consumer Runtime

- > Startup time is poor
  - Especially starting from cold
- > Installing is slow and complicated
  - The JRE software has become very large
  
- > Irritating for enterprise applications, worse for consumer applications

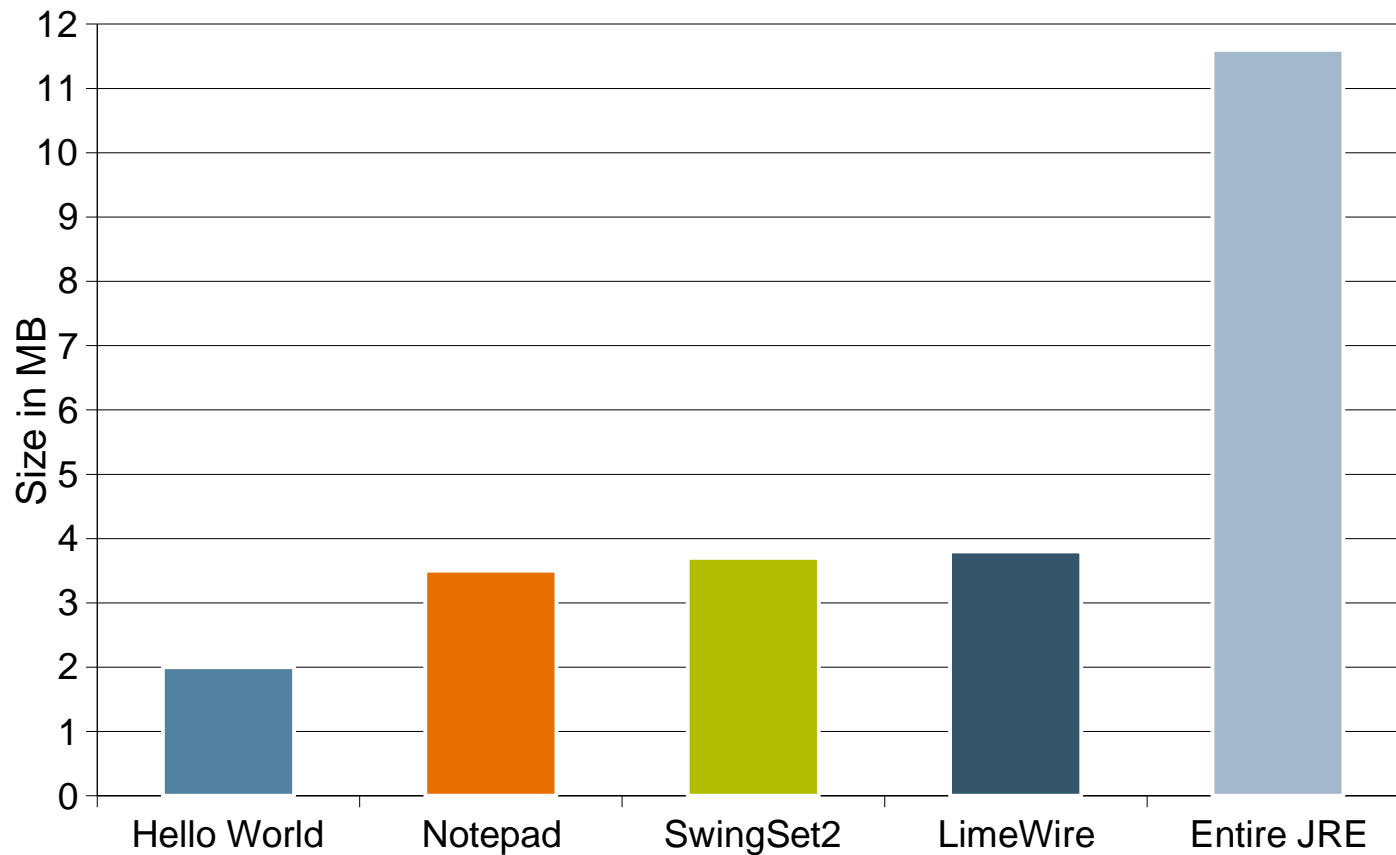
# The Consumer JRE Release

- > Quickstarter
  - Pre-load the cache, before launch
  - Not the same as having a running VM
  - Cooperates with the OS
- > A radically improved install experience
- > A **Modular** JRE version

# Java Technology- Based Kernel' : Modularizing the JRE Software

- > Just enough to run 'Hello World'
- > Install the rest in the background
  - Referencing a class
  - Class.getResource() or equivalent
  - System.loadLibrary() or equivalent
- > Custom JRE for applications that can't pause

# Promising Results



# AGENDA

Java SE today

Creating Java SE

JavaFX and consumer content

Supporting multiple languages

Modularity in the platform

Some important upgrades

**JAZOON07**

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY  
JUNE 24 - 28, 2007 ZURICH



# Languages on the JVM

“Any customer can have a car painted any color that he wants so long as it is black.”

– Henry Ford

## Why go multi-lingual ?

- > Java programming language is the best general purpose language!
- > Many other languages, many other virtues
  - Rapid prototyping and experimentation
  - Particular styles of programming
  - Mixing different types of developers
  - Or just for fun

## Java SE 6 Platform already supports Scripting

- > Scripting for the Java Platform (JSR 223)
  - Developer APIs to mix script fragments in
  - Framework APIs for adding script engines
- > Collecting conforming scripting engines
  - See [scripting.java.net](http://scripting.java.net)
- > We added a JavaScript technology engine
  - JavaScript technology works out of the box

## Multiple Languages in JDK Version 7

- > Turbo-charging scripting engines
  - New bytecode for dynamic method dispatch
  - ‘Supporting Dynamically Typed Languages on the Java Platform’ (JSR 292)
  - Investigate hotswapping
- > Bundling more dynamic language engines
  - ‘The Beanshell Scripting Language’ (JSR 274)
  - JRuby, Jython, BeanShell, Groovy
  - JavaFX™ Script

## Evolving the Java Programming Language

- > Reading is more important than writing
  - > One language same meaning; everywhere
  - > Simplicity matters
- 
- > Seeking a small number of changes for Java SE 7 platform

# Evolving the Java Programming Language

- > 'superpackages'
- > Extensions to the annotation syntax (JSR 308)
- > Language support for Java technology properties
- > Control abstraction constructs
  - Closures
  - Concise instance creation expressions
  - First-class methods
- > Operator overloading
- > 'Rough edges'
  - Shorter variable declaration, strings in switch statements, Enum comparisons

# Modularity in Java SE Platform

## > Development time

- Interfaces and implementation classes
- Information hiding
  - public**, **private** and **protected**, class level and below
- Assertions

## > Deployment time

- Java ARchives (JAR)
- Resources framework

# AGENDA

Java SE today

Creating Java SE

JavaFX and consumer content

Supporting multiple languages

**Modularity in the platform**

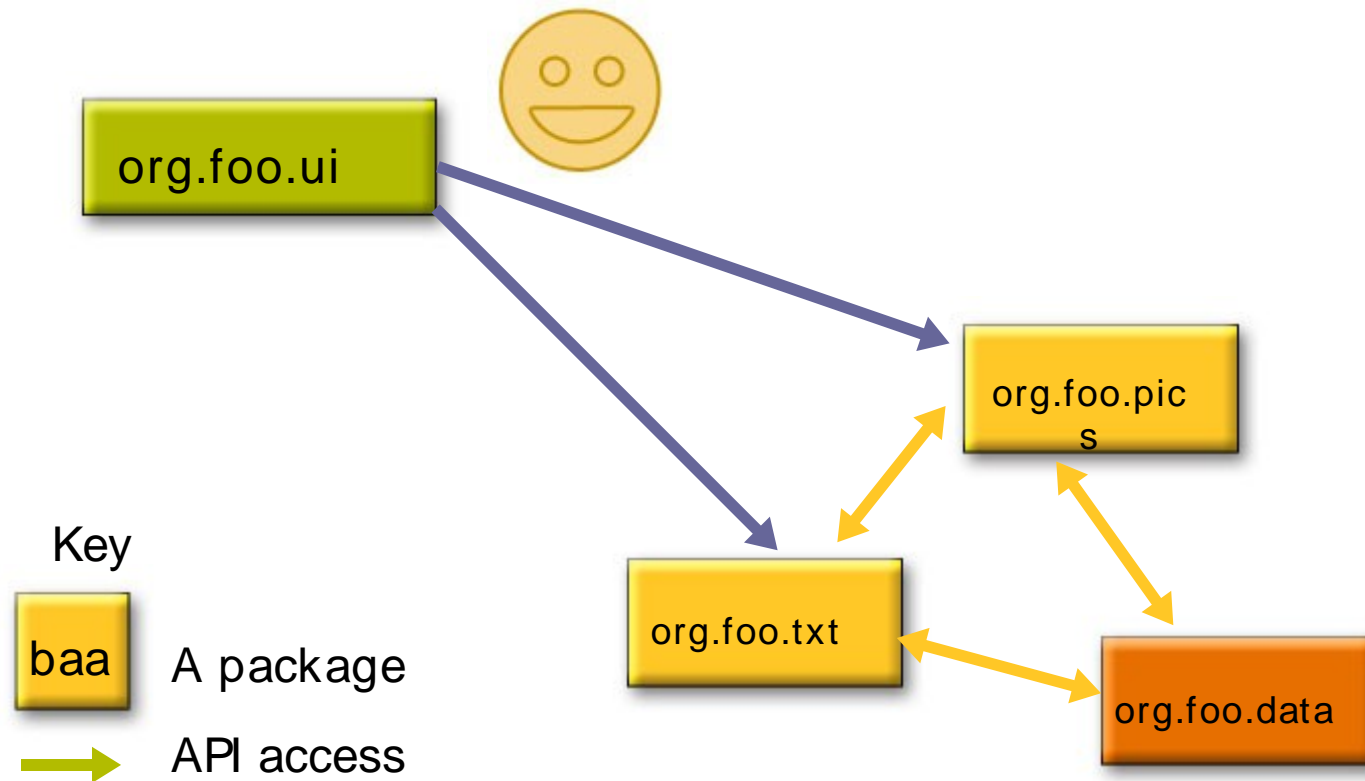
Some important upgrades

**JAZOON07**

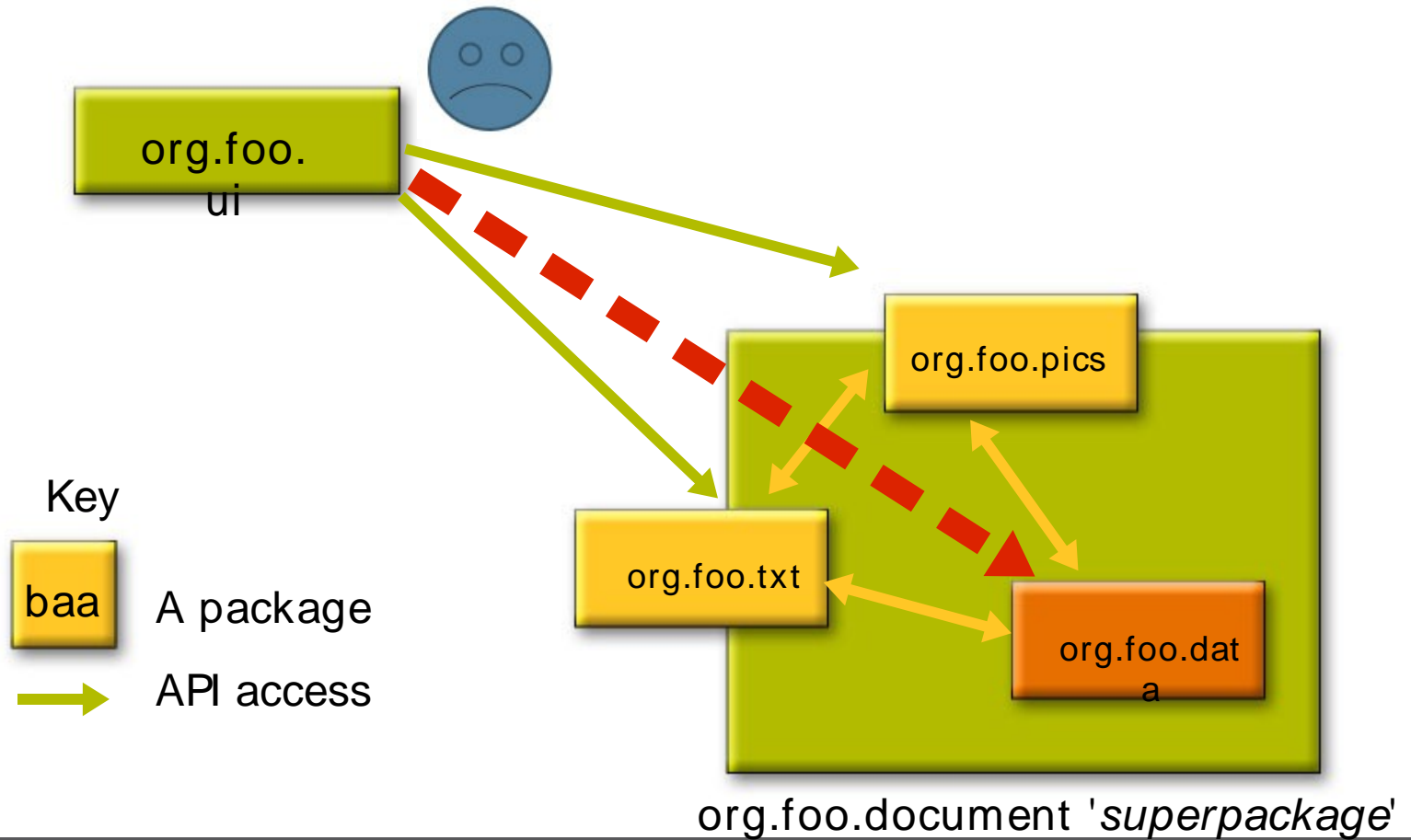
THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY  
JUNE 24 - 28, 2007 ZURICH



# Java Technology Packages Dilemma



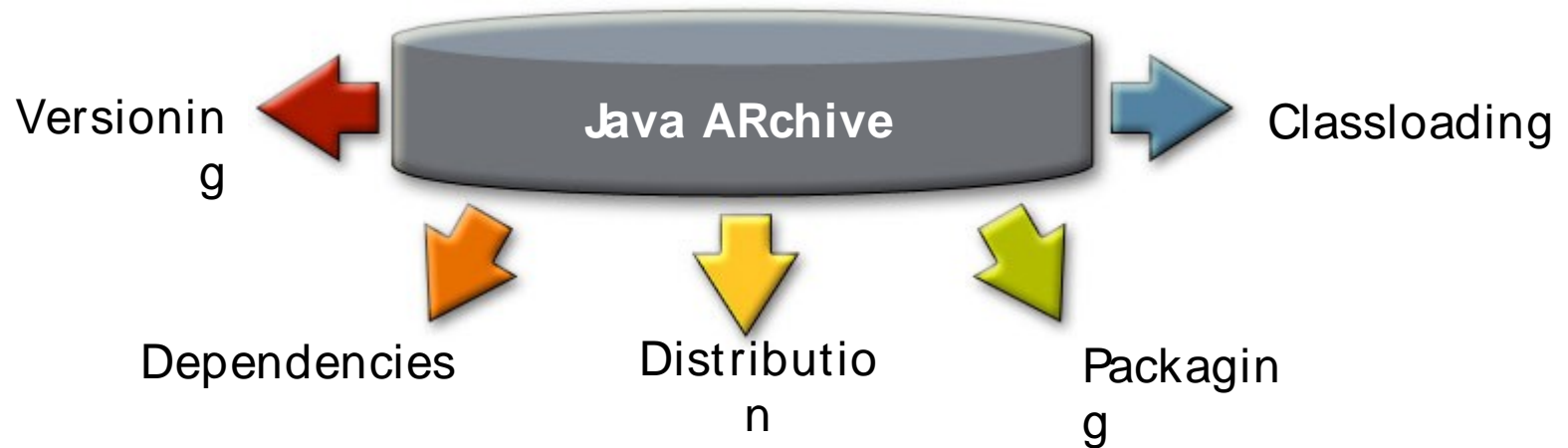
# Dilemma Resolved by superpackage



# Draft Syntax for superpackages

```
super package org.foo.document {  
    // super-package exports:  
    export  
        org.foo.pics.MyImageInterface;  
    export org.foo.txt.*;  
  
    // super-package members:  
    org.foo.pics.MyImageInterface;  
    org.foo.txt.*  
    org.foo.data.*;  
}
```

# Packaging Applications



- > JAR file format is stretched to its limit
  - Hard to version
  - Difficult to manage
  - Can't express dependencies
- > Due for major upgrade in Java SE 7 platform
  - Java Application Modules (JAM)

# Anatomy of a JAM File

```
org.foo.document- 1.2.3.jam:  
  / META- INF/ MANIFEST.MF  
  / MODULE- INF/ METADATA.module  
  / MODULE- INF/ bin/ doc- windows.dll  
  / MODULE- INF/ bin/ doc- linux.so  
  / MODULE- INF/ lib/ helper.jar  
  / org/ foo/ pics/ MyImageInterface  
  / org/ foo/ txt/ Word.class  
  / org/ foo/ txt/ Sentence.class  
  / org/ foo/ data/ Letter.class  
  / org/ foo/ data/ Number.class  
  / icon/ graphics.jpg
```

**Metadata—under  
/ MODULE- INF directory**

**Native libraries—under  
/ MODULE- INF/ bin  
directory**

**Embedded JAR files—  
under / MODULE- INF/ lib  
directory**

**Resources—e.g. classes,  
images, etc.**

## Summary: Modularity in Java SE Platform

- > Development time: superpackages
  - JSR 294: Improved Modularity Support
- > Deployment time: superJARs
  - JSR 277: Java Module System
  - Aiming for interoperability with OSGi and other systems

## Management Updates

- > JSR 255: Java Management Extensions (JMX™) Specification, version 2.0
  - Namespaces, federated JMX technology servers
  - [opendmk.dev.java.net](http://opendmk.dev.java.net)
- > JSR 262: Web Services Connector for JMX Agents
  - Based on ws management standards
  - Early Draft Available
  - [ws-jmx-connector.dev.java.net](http://ws-jmx-connector.dev.java.net)

# AGENDA

Java SE today

Creating Java SE

JavaFX and consumer content

Supporting multiple languages

Modularity in the platform

Some important upgrades

**JAZOON07**

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY  
JUNE 24 - 28, 2007 ZURICH



## More New IO APIs for Java SE Platform

- > JSR 203: In early draft
- > New file system API `java.nio.filesystem.Filesystem`
  - Listen for filesystem changes
  - Security attributes (file permissions, ACLs,...)
  - Symbolic links
  - Adds operations: copy, move, walk file tree,...
  - Scalable access to directories

# Summary Proposed JSRs for Java SE 7 Platform

JSR 277: Java Module System

JSR 294: Improved modularity support in the Java programming language

JSR 295: Beans binding

JSR 303: Bean validation

JSR 292: Supporting Dynamically Typed Languages

JSR 296: Swing application framework

JSR 203: More new I/O APIs for the Java Platform (NIO.2)

JSR 220: Enterprise JavaBeans™ 3.0

JSR 255: JMX specification, version 2.0

JSR 262: Web services connector for JMX agents

**JSR 260: Javadoc™ Tag Technology Update**

*JSR(s) TBD Java Language changes*

JSR 308: Annotations on Java Types

**JSR 310: Date and Time API**

# Proposed Technologies for Java SE 7 Platform

## Specification

JSR 277: Java Module System

JSR 294: Improved Modularity Support in the Java Programming Language

JSR 295: Beans Binding

JSR 303: Bean Validation

JSR 296: Swing Application Framework

JSR 203: NIO.2

JSR 292: Supporting Dynamically Typed Languages

JSR 220: Java Persistence Architecture

JSR 255: JMX Specification, version 2.0

JSR 262: Web Services Connector for JMX

JSR 260: Javadoc Tag Technology Update

*JSR(s) TBD Java Language changes*

JSR 308: Annotations on Java Types

JSR 310: Date and Time API

## Draft ?

Early Draft

-

V 0.6 API

-

-

Early Draft

-

Final Spec

-

Early Draft

-

-

-

-

## Implementation

[openjdk.java.net/projects/modules](http://openjdk.java.net/projects/modules)

[openjdk.java.net/projects/modules](http://openjdk.java.net/projects/modules)

[beansbinding.dev.java.net](http://beansbinding.dev.java.net)

-

[appframework.dev.java.net/](http://appframework.dev.java.net/)

-

-

[glassfish.dev.java.net](http://glassfish.dev.java.net)

[opendmk.dev.java.net](http://opendmk.dev.java.net)

[ws-jmx-connector.dev.java.net/](http://ws-jmx-connector.dev.java.net/)

-

[ksl.dev.java.net](http://ksl.dev.java.net)

[pag.csail.mit.edu/jsr308/](http://pag.csail.mit.edu/jsr308/)

[jsr-310.dev.java.net/](http://jsr-310.dev.java.net/)

## Summary

Java SE 6 platform

> **Use it now** - [java.sun.com/javase](http://java.sun.com/javase)

OpenJDK

> **Join us in building it** - [openjdk.java.net](http://openjdk.java.net)

JavaFX

> **Try it out** - [openjfx.java.net](http://openjfx.java.net)

Java SE 7 platform

> **Join us in defining it!**

Visit us at [planetjdk.org](http://planetjdk.org)

**JAZOON07**

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY  
JUNE 24 - 28, 2007 ZURICH



Danny Coward  
[blogs.sun.com/ dannycoward](http://blogs.sun.com/dannycoward)  
Sun Microsystems  
[danny.coward@sun.com](mailto:danny.coward@sun.com)

**JAZOON07**

THE INTERNATIONAL CONFERENCE ON JAVA TECHNOLOGY  
JUNE 24 - 28, 2007 ZURICH

