



What's coming in Java SE 7

Danny Coward
Java SE Platform Lead



Děkuji

Thankyou for having such good
english !

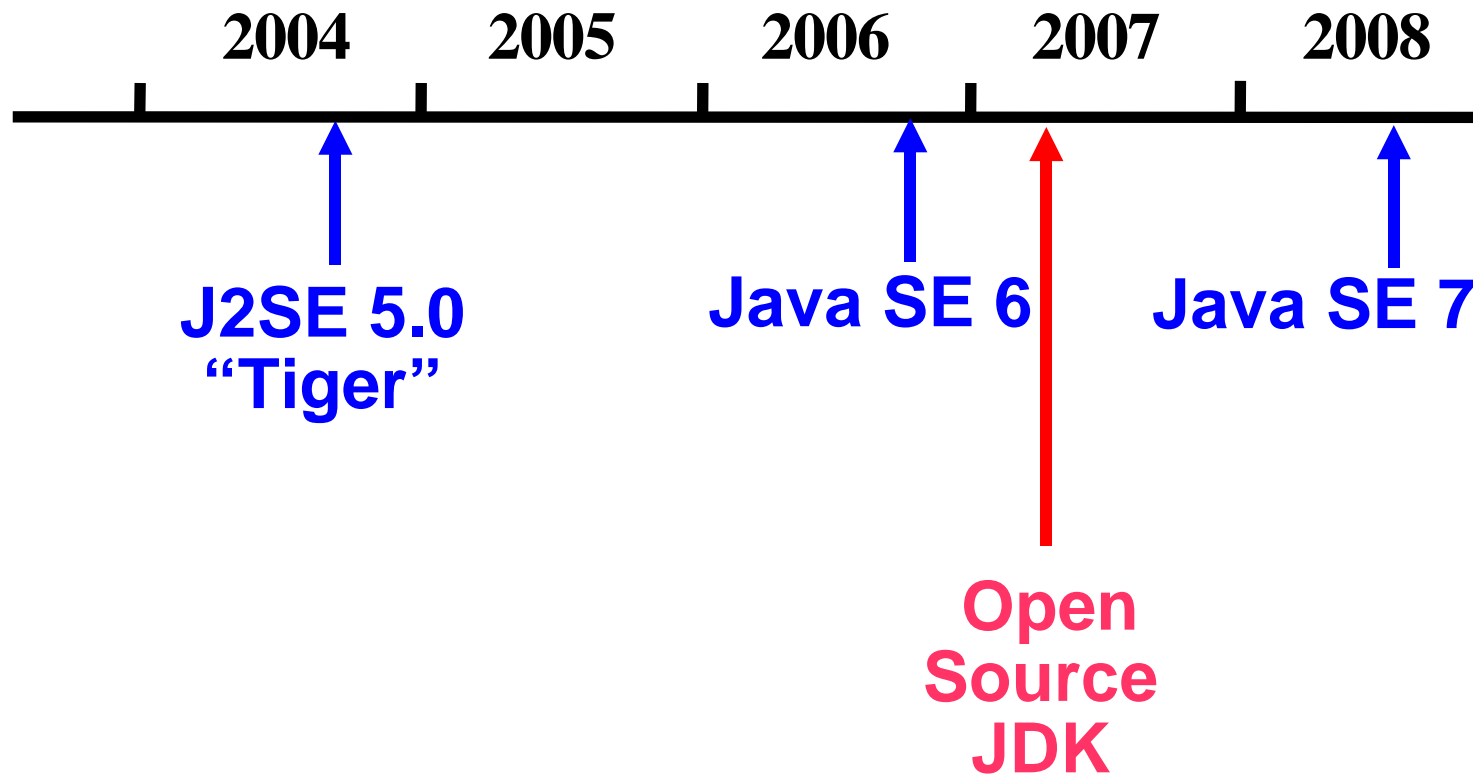
Java SE 7 Update

- The stage we are at
- The APIs & formats we plan on including
- The language features we're considering
- An interesting non-API feature

Java SE 7 Update

- The stage we are at
- The APIs & formats we plan on including
- The language features we're considering
- An interesting non-API feature

Java SE Timeline



We're (finally) Open Sourcing the JDK

- November 2006
 - > Compiler, Hotspot
- Mid-2007
 - > Fully buildable JDK
 - > Modulo a few components
- Get the latest
 - > <http://community.java.net/jdk/opensource/>

Java SE 7 Update

- The stage we are at
- The APIs & formats we plan on including
- The language features we're considering
- An interesting non-API feature

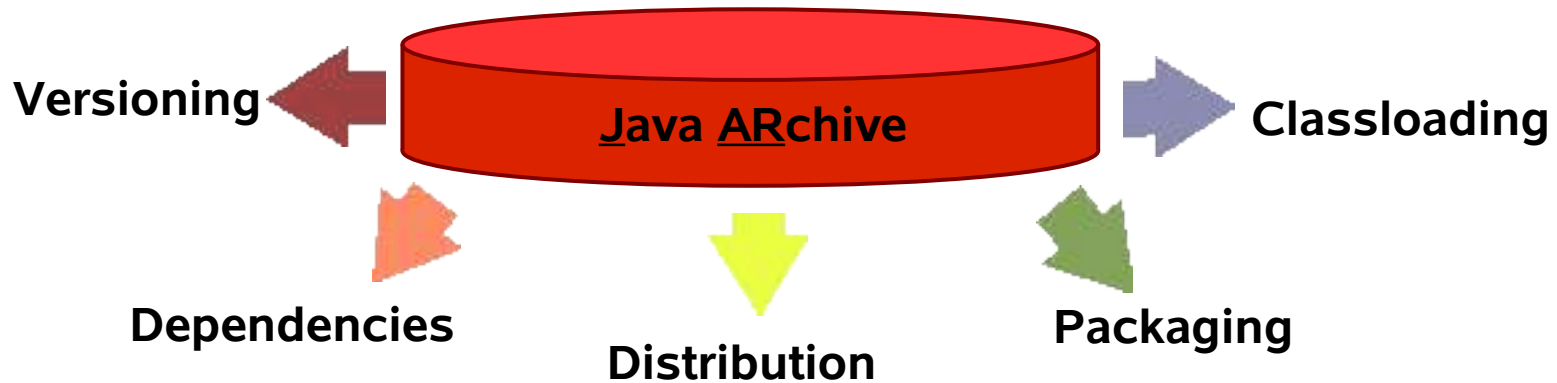
Java Community Process (jcp.org)

Many, **many** discussions to come



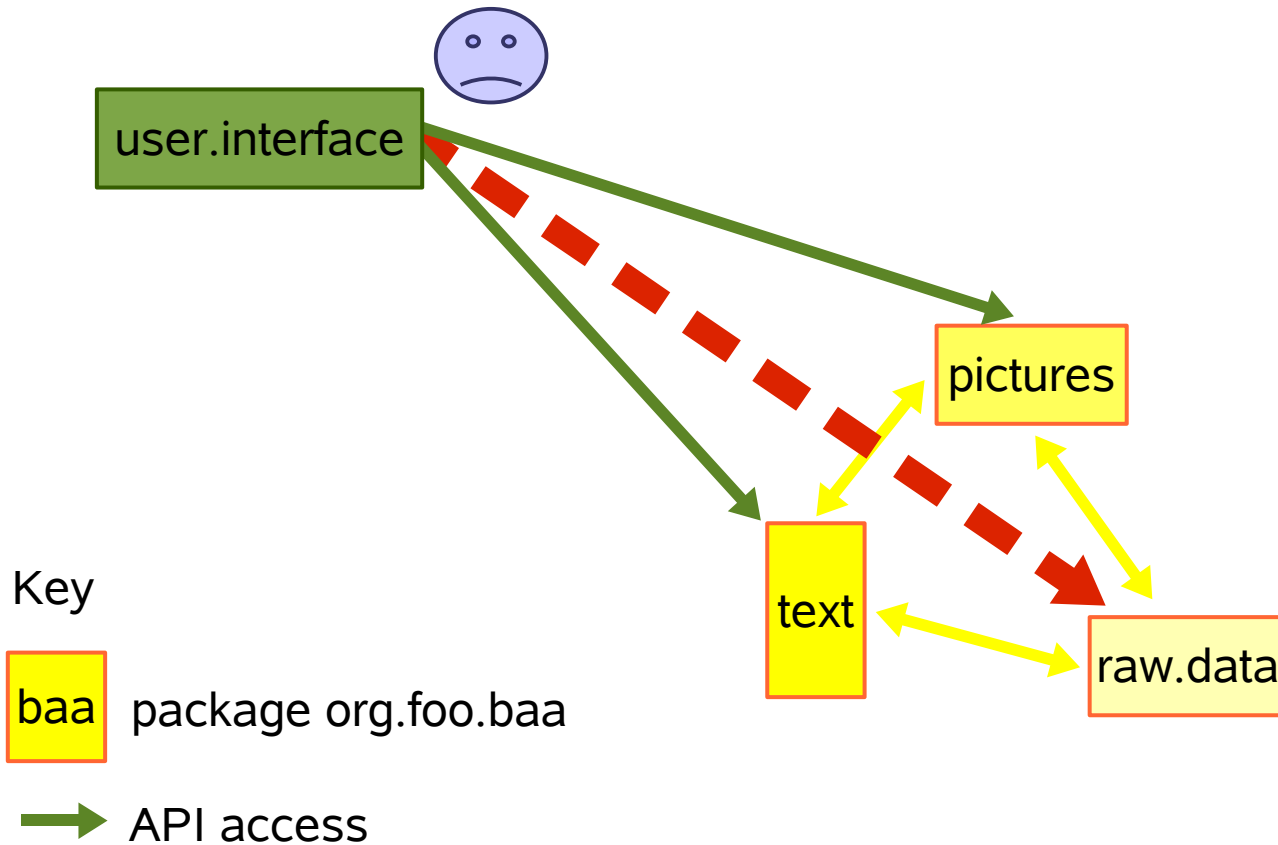
many...

SuperJARs / JAva Module (JAM)

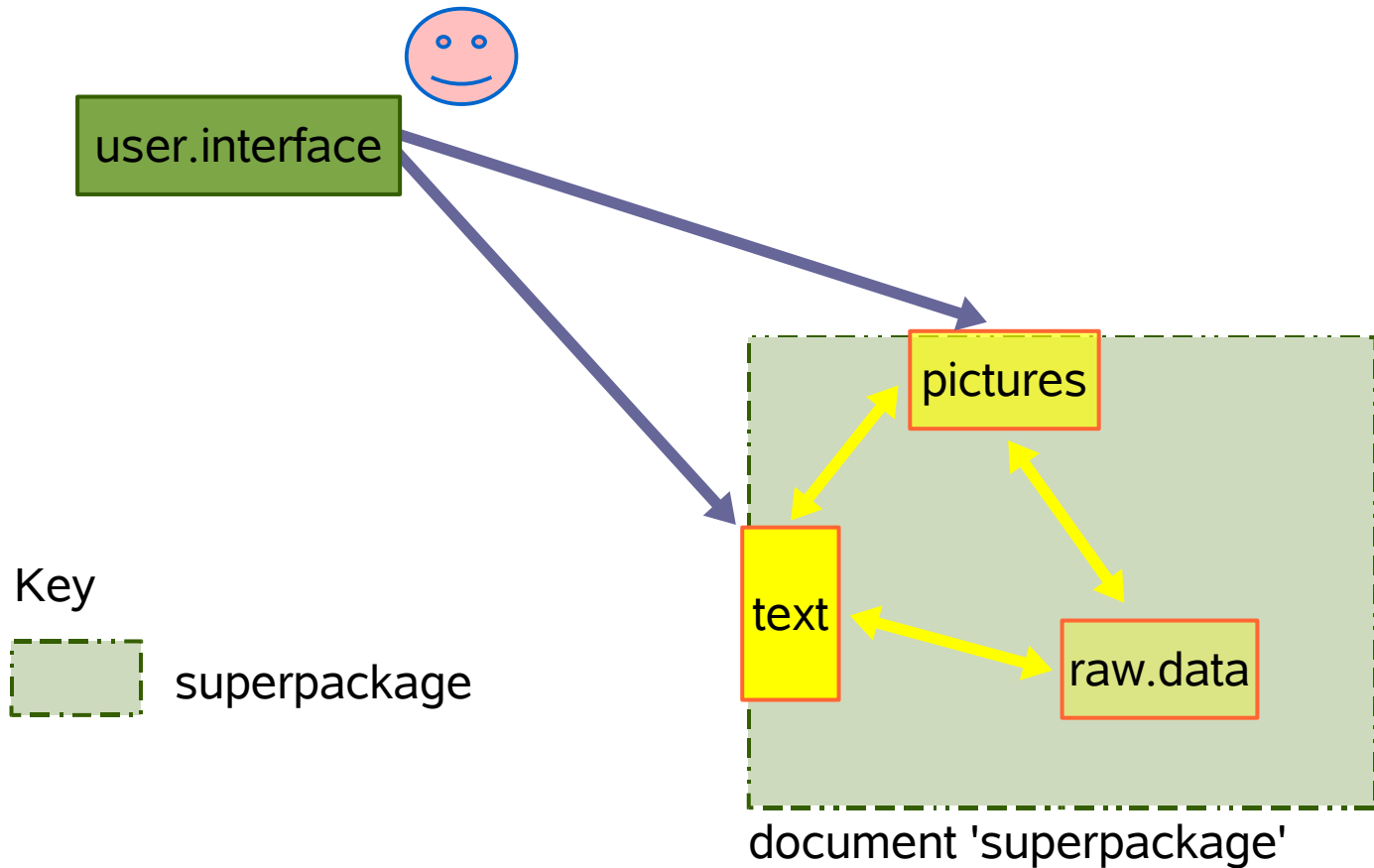


- JSR 277: Java Module System
- New deployment and distribution format
- Friendly to JARs
- There is an Expert Draft to review
 - > You may be packaging your applications as .JAMs in the future !

Publishing APIs: All or nothing



JSR 294 Superpackages



A superpackage syntax proposal

```
super package org.foo.document{  
    // super-package exports:  
    export org.foo.pictures.MyImageInterface;  
    export org.foo.text.*;  
    // super-package members:  
    org.foo.pictures.MyImageInterface;  
    org.foo.text.*  
    org.foo.raw.data.*;  
}
```

Language: Freedom of Choice

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

Henry Ford

Dynamically Typed Languages

```
// Collection has grown; figure out the
// next increment in size
newSize(c) {
    return c.size() * c.growthFactor();
}
```

Dynamically Typed Languages

```
// Collection has grown; figure out the
// next increment in size
newSize(c) {
    return ((Interface91) ((Interface256) c).size()) *
           ((Interface91) ((Interface42) c).growthFactor());
}
```

Supporting Dynamic Languages

- Many dynamic language interpreters
- Hard to write, brittle and slow
- Dynamic typed method calls are the problem
- JSR 292 New bytecode – *invokedynamic*

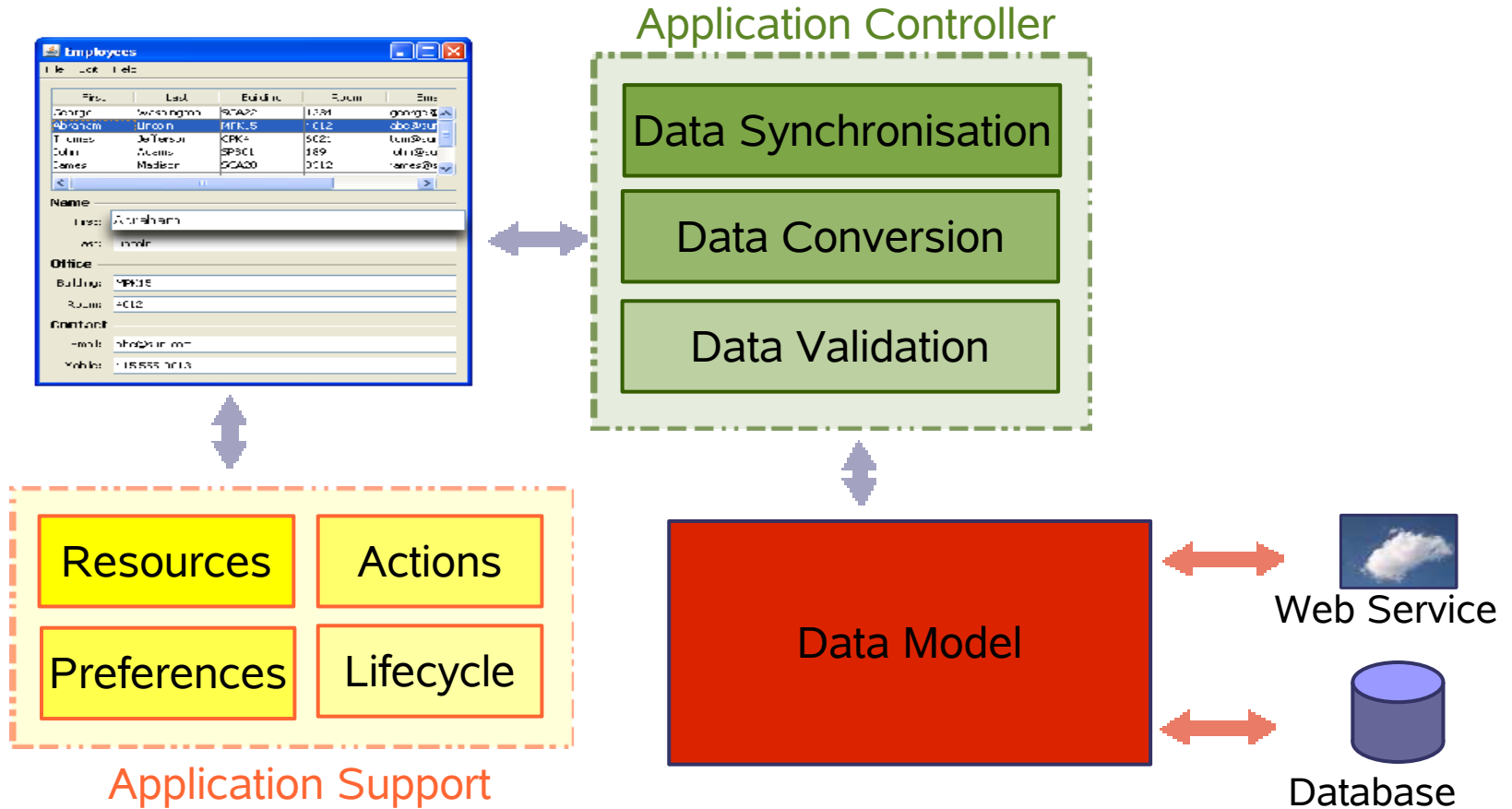
Also planning to co-bundle other language interpreters with JDK 7

- > JRuby, Jython, Beanshell (JSR 274)...
- > See scripting.dev.java.net (22+ engines)

Swing Development

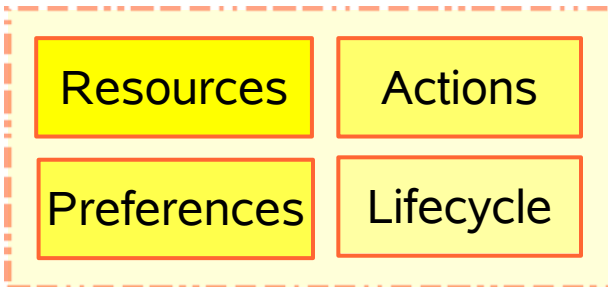
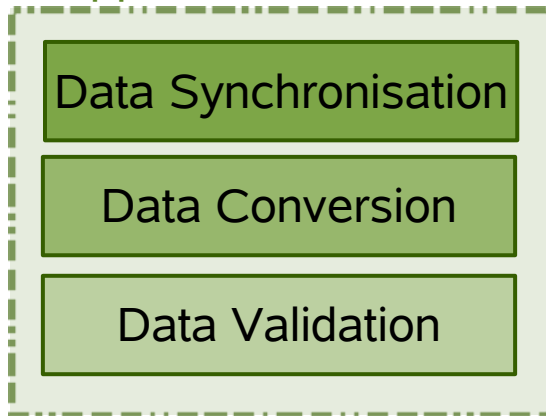
- Swing is a powerful toolkit
- Some developers are put off
 - > 'Routine things should be easier'
 - > 'There's too much choice'
 - > 'I end up writing too much code'
 - > 'Its difficult to teach'
- Time to make it easier !

A Typical Desktop Application



Swing Ease of Use JSRs

Application Controller



- “Beans Binding” (JSR 295)
 - Formalising the Swing Controller
 - API for connecting JavaBeans
 - “Bean Validation” (JSR 303)
 - Meta data model to express validation constraints
 - “Swing Application Framework” (JSR 296)
 - Formalising support functions
- All these JSRs have just got started

JSRs for Java SE 7

- JSR 203 More New I/O APIs (NIO 2)
 - > New filesystem API
 - > Asynchronous I/O
 - > Early draft soon
- Java Persistence API from JSR 220
 - > Two way mapping between Java objects and relational data
 - > ..with focus on Ease of Use
 - > Already part of Java EE 5

Java SE 7 JSRs

- JSR 255 JMX 2.0
 - > Ease of development annotations
 - > Enable federated JMX servers
 - > Array of incremental improvements
- JSR 262 Web Services Connector for JMX Agents
 - > Web service standards for remote management
 - > Promote standard management tools
 - > In early draft stage

Java SE 7 JSRs

- JSR 260: Javadoc Tag Technology Update
 - > Define new tags and documentation mechanisms
 - > Categorisation of methods
 - > Embedded examples, common usages
 - > Information views
 - > Dropped for Java SE 6 :-(
 - > On the list for Java SE 7 :-)

Summary of JSRs

- JSR 277 Java Module System
- JSR 294 Improved Modularity Support in the Java Language
- JSR 295 Beans Binding
- JSR 303 Beans Validation
- JSR 296 Swing Application Framework
- JSR 203 NIO2
- JSR 220 Java Persistence APIs
- JSR 255 JMX 2.0
- JSR 262 Web Services Connector for JMX
- JSR 260 Javadoc Technology Update
- *JSR(s) TBD Java Language changes*

Java SE 7 Update

- The stage we are at
- The APIs & formats we plan on including
- The language features we're considering
- An interesting non-API feature

Java SE Personals Ad

“Popular, mature platform seeks small package of language additions for fun productive times, easy reading, and with similar sense of style. Must have Good Sense of Humor.”

We have had many replies !

Language Proposals

- Superpackages
 - > see above
- Language level Xml Support

DOM/SAX manipulation of XML fragments

```
elt.appendChild(<present>  
    <who>currentSpeaker</who>  
    <day>Calendar.TUESDAY</day>  
    </present>);
```

Language support for Xpath queries

Language Proposals

- Writing Java Beans properties

```
private String foo;  
public String getFoo() {...}  
public void setFoo(String foo){...}
```

```
public property String foo;
```

- Reading JavaBeans properties

```
a.setFoo(b.getFoo());
```

```
a->Foo = b->Foo;
```

Language Proposals

- BigDecimal

```
interestBd.multiply(percentageBd.divide(100));
```

```
interestBd * ( percentageBd / 100 )
```

- Strings in switch statements
- Comparisons for Enums

```
boolean isRoyalty(Rank r) {
    return rank >= Rank.JACK && rank != Rank.ACE;
}
```

- Shorter Instance creation

```
Map<String, List<Integer>> m = new HashMap<String, List<Integer>>();
Map<String, List<Integer>> m = new (); ??
Map<String, List<Integer>> m = new Hashmap (); ??
```

Language Proposals

- Closures, Lambdas and friends
 - > **Method References/Tidying up IC creation**

```
addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent ae) {
        updateList(ae);
    }
});
```

```
addActionListener(updateList);
```

```
addActionListener(new ActionListener() {
    updateList(ae);
});
```

- > Full, Formal Closure Support

Java SE 7 Update

- The stage we are at
- The APIs & formats we plan on including
- The language features we're considering
- An interesting non-API feature

Java Kernel

- Radically improvement in:
 - > installation experience
 - > hot and cold startup times
- Define Java Kernel
 - > install rest of platform on demand / in parallel

Could be pretty Flash-y !

Things to remember

- Java SE Homepage
 - > java.sun.com/javase/index.jsp
- JDK Blogs
 - > planetjdk.org
- Me
 - > blogs.sun.com/dannycoward
 - > danny.coward@sun.com
- Open source rollout
 - > community.java.net/jdk/opensource/

End

