Java Technology Community Development: Peabody & Glassfish

Simon Ritter
Java Technology Evangelist
Sun Microsystems, Inc.

Agenda

- Sun and Open * Computing
- The Future of Java SE - Mustang
- The Future of Java EE – GlassFish
- Resources & Summary

Sun and Open * computing

Open * Computing

"Innovation Happens Elsewhere"

-Bill Joy

Massive Community Growth

- Java.sun.com
- Sun Developer Network (SDN)
- NetBeans.org
- Java.net

Sun Developer Network

www.sun.com/developers

- Content
  - Developer portal
  - Cool stuff
  - White papers
  - Code samples
- Training
  - Classes
  - Seminars
  - Webinars
- Support
  - Knowledge base
  - Forums (5000+ active)
  - User Groups (537 and growing)
- Software
  - Electronic download
  - Quarterly freeware mailing

- **769,000 unique registrations**
- **112,000 members of SDN Mobility Program**

NetBeans.org

- Center for Innovation, News and Support
- 3.5 Million Downloads of NetBeans tool
- 100 add-ons and extensions
- 27,000 registered users
- Visitors from 160 countries
- Active Community supporting developers in English, French, Russian, Japanese, and simplified Chinese

java.net Building Open Communities

- Shared vision, goals
- Agreed sharing (license)
- Agreed relationships (governance)
- Committed members

Not Just Open Source; One Size Doesn't Fit All!

Java.net

- Goal is to stimulate the expansion of Java by building a web-based community where developers, educators, researchers, and others to learn, share, and collaborate
- The Single Source for Java Technology Collaboration
  - > Unifying Community of other Java communities
  - > 100K+ members
  - > Nearly 1000 Projects

Key java.net Features

Collaboration

- Weblogs
- Wikis – Collaborative Web Pages
- Mailing Lists
- Forums – Project/Community Discussion Forums

Information

- RSS – Content Syndication Newsfeeds
- News – Relevant Project/Community Information
- Javapedia (an encyclopedia of everything Java related)

Development

- CVS Source Trees – Source Repository, Versioning
- Issuezilla – Bugs & Issues Tracking
- Help Wanted – to match projects and developers

Communities on java.net

- Java Web Services & XML
- Java Desktop
- Java Tools
- Java Patterns
- ...

- Mustang (Java SE 6.0)
- GlassFish (Java EE 5.0)
- Many more ...

- Java Communications
- JAIN
- Java Games
- Robotics
- ...

- NetBeans
- Jini
- Project JXTA
- Linux
- ...

The Future of Java SE
Mustang

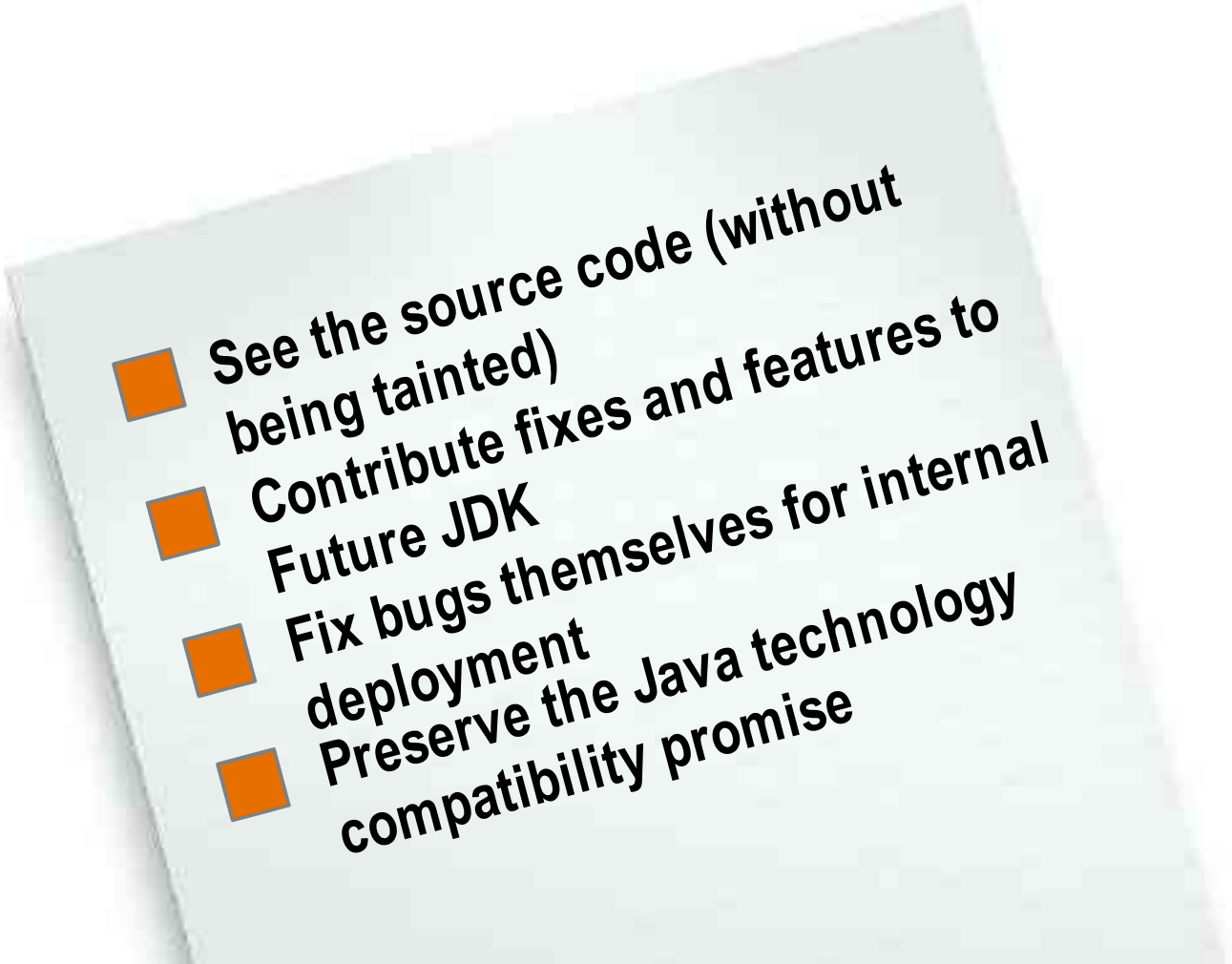
1.4.0	**Merlin**	**2002/2/13**
1.4.1	Hopper	2002/10/16
1.4.2	Mantis	2003/5/29
5.0	**Tiger**	**2004/9/30**
6	**Mustang**	**2006/Q3**
7	**Dolphin**	**2008/Q1**

With Mustang on java.net, Future Java Technology Development is More Open and Transparent

- We listened
- Developer mindshare is vital
- Build grassroots support
- Channel enthusiasm for community development

...all while retaining compatibility

What Java Developers Ask For:



Licenses for JDK

- Java Research License (JRL)
- Java Internal Use License (JIUL)
- Java Distribution License (JDL)

Java Research License (JRL)

http://java.net/jrl.html

- For academics, researchers, the curious, contributors
- Greatly simplified:
  - Simple 2-pager, click-through
  - Allows very broad use for research
  - You don't get “tainted”
  - No copyright or patent rights
  - No compatibility, TCK requirements
- Not for commercial use, redistribution
- Contribution agreement for giving code back – must be compatible

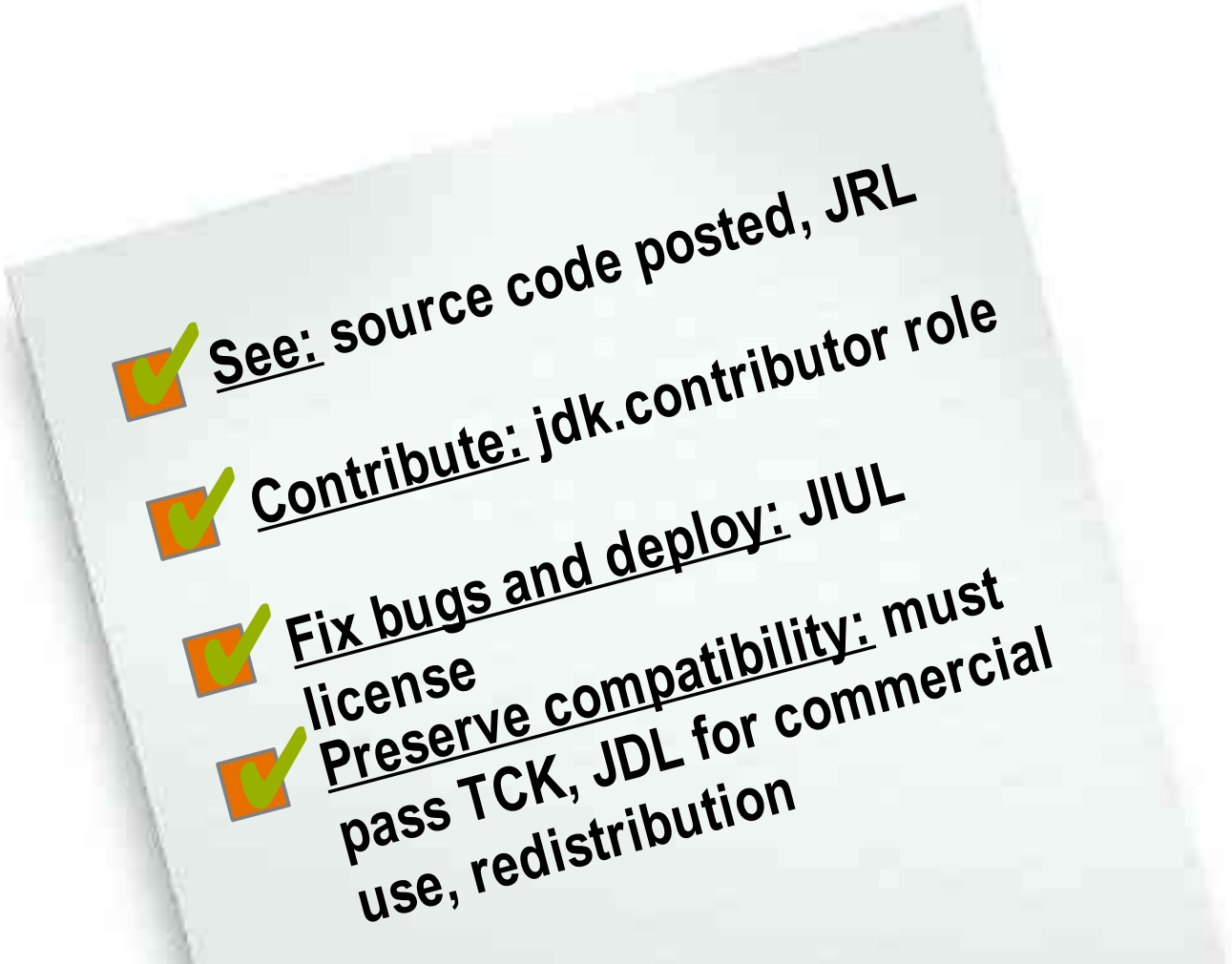
Java Internal Use License (JIUL)

- Enterprise customers can deploy their own bugfixes
  - Sun can enforce compatibility but doesn't require running TCK
  - Honor system - “At the customers own risk”
  - Some guidelines: no API changes
- For internal use
  - Includes software powering public websites
  - Excludes external distribution
- Allows sharing of bugfixes
  - Encourages but doesn't require returning fixes
- All about peace of mind

Java Distribution License (JDL)

- Commercial redistribution license
  - Principal successor to SCSL
  - Simplified (more readable) commercial use license
- Includes main commercial terms from SCSL
  - Requires passing the TCK
  - Includes TCK license
  - Optional support agreement
  - Requires trademark agreement
- No surprises

What Java Developers Get With Project Peabody:



People Are Already Actively Partipating: Contributions Integrated

6207243:	Non-varargs warnings during build, should be cleaned up	Andy Tripp b34
6257449:	Concurrency bug in sound UlawCodec with tempBuffer	Jesse Sterr b34
4238932:	A JTextField in gridBagLayout does not properly set MinimumSize	Swen Meier b42

THANK YOU!


Compiler API
MBeans metadata
JTable upgrades
Longhorn Look & Feel
Splash screens
Split Verifier
Windows system tray
Unicode Normalizer
Services
Attach on demand
chmod
Parallelize Concurrent GC
JConsole upgrades
Core JVM performance
SwingWorker
Annotation processors
Parallel old-space GC
Mustang
Web Services Stack
Password prompting
JVM DTrace
JavaDoc
LCD fonts
Docs in Chinese
JDBC 4.0
Faster JNI
Tags
JAXB 2.0
Free disk space
More gfx acceleration
Improved OOM diagnosability
JVM & CLR Co-Existence
More desktop integration
FireFox support
Pluggable Locales
Scripting Languages
Native L&F
XAWT
HTTP cookie manager
More GC Fidelity
JavaScript engine
XML digital signatures
Improved text rendering
Ergonomics

Mustang Themes

Compatibility, Stability, & Quality!
Diagnosability, Monitoring, & Management
XML & Web Services
Ease of Development
Enterprise Desktop
Transparency

JSR 270: Java SE 6 Release Contents

Expert Group Members

Apache
BEA
David Bock
Capgemini
Google
HP
IBM
Ikayzo
Intel
JBoss
Doug Lea
Metasolv
Oracle
Sam Pullara
SAP
SAS Institute
Michael Santos
ThoughtWorks

Approved Mustang Component JSRs

202: Class File Update

199: Compiler API

269: Annotation Processors

260: Javadoc™ Tag Update — *Ease of Development*

221: JDBC™ 4.0

223: Scripting

105: XML Digital Signature

173: Streaming API for XML — *XML*

222: JAXB 2.0

250: Common Annotations

181: WS Metadata

224: JAX-WS 2.0 — *Web Services*

202: Class File Update

Mustang Component JSRs

Primary change: Split verification

- Adopted from J2ME™ platform
- Verifier checks compiler-generated assertions instead of generating and checking assertions itself

Why?

- Simplicity
- Performance
- Sharing code between ME & SE

199: Compiler API

Mustang Component JSRs: Ease of Development

- Java application code can interact with compiler
- Simplifies IDE development
- Simplifies JSP processing
- Source code can come from anywhere
  - Memory, jar file, file system

269: Annotation Processors

Mustang Component JSRs: Ease of Development

- Annotations introduced in J2SE 5.0
  - apt tool included in JDK
- Extremely useful for simplification of J2EE
- Annotation processors allow simple handling of user defined annotations

260: Javadoc Tag Update

Mustang Component JSRs: Ease of Development

Properties

Label	The string label of the button. (`getLabel`, `setLabel`)
ActionCommand	The command name of the action event fired by this button. (`getActionCommand`, `setActionCommand`)

221: JDBC 4.0 Software

Mustang Component JSRs: Ease of Development

```
private Connection connect(String user, String passwd)
    throws SQLException
{
    String url = "jdbc:mysql://javadb.sfbay/jplan";
    String driver = "com.mysql.jdbc.Driver";
    try {
        Class.forName(driver);
        return DriverManager.getConnection(url,
                                           user,
                                           passwd);
    } catch (ClassNotFoundException x) {
        throw new SQLException("Can't load driver", x);
    }
}
```

221: JDBC 4.0 Software

Mustang Component JSRs: Ease of Development

```
private Connection connect(String user, String passwd)
    throws SQLException
{
    String url = "jdbc:mysql://javadb.sfbay/jplan";
    String driver = "com.mysql.jdbc.Driver";
    try {
        Class.forName(driver);
        return DriverManager.getConnection(url,
                                           user,
                                           passwd);
    } catch (ClassNotFoundException x) {
        throw new SQLException("Can't load driver", x);
    }
}
```


Smaller Mustang Features
Compiler API
Longhorn Look & Feel
MBeans metadata
JTable upgrades
Splash screens
Split Verifier
Windows system tray
Unicode Normalization
Services
Parallelize Concurrent GC
JConsole upgrades
chmod
SwingWorker
Annotation processors
Parallel old-space GC
Password prompting
JVM DTrace
LCD fonts
Docs in Chinese
JDBC 4.0
JavaDoc Tags
Faster JNI
JAXB 2.0
Free disk space
Improved OOM diagnosability
More desktop integration
JVM & CLR Co-Existence
FireFox support
Pluggable Locales
Scripting Languages
Native L&F
HTTP cookie manager
XAWT
JavaScript engine
More GC Fidelity Ergonomics
XML digital signatures
Improved text rendering

Your votes matter!
Bugs Fixed in Mustang (so far)

239	4521075: Drag gesture in Java technology different from Windows
121	4799499: Dead key followed by space should produce non-dead character
91	6205422: Applet load time slow in JRE 5.0 due to Java logo animation
84	4202291: Long lines in wrapped JTextArea cause severe performance degradation
82	4949631: String.getBytes() does not work on some strings larger than 16MB
50	5092063: Extremely slow socket creation using new Socket("ip-address", port)
47	6195591: JDK 1.5.0 software and Fedora Core 3: Java Web Start doesn't work
37	4255200: Modal dialog should come to the front
29	4950176: drawImage bad interpolation for non-opaque INT_ARGB pixels
29	4828461: Support Zip files with more than 64k entries
27	4158988: 2-click on upper left corner can't close JInternalFrame
25	5097939: Unnecessary rw opening of /dev/random while compiling
25	4818143: NetBeans™ IDE hangs on ClipboardTransferable.getClipboardData
19	4302764: Focus is not set in JInternalFrame
18	5106833: NullPointerException in XMenuPeer.repaintMenuItem

http://bugs.sun.com

Your votes matter!
RFEs implemented in Mustang (so far)

697	4057701:	Need way to find free disk space
475	4080029:	Modal Dialog block input to all frame windows not just its parent
378	4726365:	Support LCD anti-aliased text (sub-pixel resolution)
224	4212439:	No way to reload a ResourceBundle for a long-running process
189	4609228:	Provide additional local calendars
152	4879835:	Provide the dynamic bytecode instrumentation capability (JFluid)
133	4528599:	Support all file types in Java Plug-In cache
93	4661156:	Full screen exclusive mode and display change not supported on Linux
43	4986256:	Add support for JSR 175's java.lang.SuppressWarnings
43	4499556:	Use arbitrary (J)Components as JTabbedPane tab labels
32	4339577:	Adding kerning to the text layout process
28	4468566:	DnD should not use selection to show drop location
27	4681682:	Include SwingWorker with the JDK
27	4339415:	Provide a writer plug-in for the GIF file format
26	4502804:	FontSmoothing/AntiAlias not utilized by default despite OS setting

http://bugs.sun.com

Free Disk Space

☑ 697 votes

```
void safeCopy(File src, File dstDir)
    throws IOException
{
    if (src.length() > dstDir.getUsableSpace())
        throw new IOException("Insufficient space");
    doCopy(src, new File(dstDir.getParent(),
           src.getName()));
}
```

Free Disk Space

☑ 697 votes

```
void df(File dir) {
    out.format("Total MB    Used    Free  Use%%%n");
    long t = dir.getTotalSpace();
    long f = dir.getFreeSpace();
    out.format("  %6d  %6d  %6d   %2d%%%n",
               t >> 20,
               (t - f) >> 20,
               f >> 20,
               ((t - f) * 100) / t);
}
% java DF /a
Total MB    Used    Free  Use%
   32766   28632    4134   87%
%
```

Class-Path Wildcards

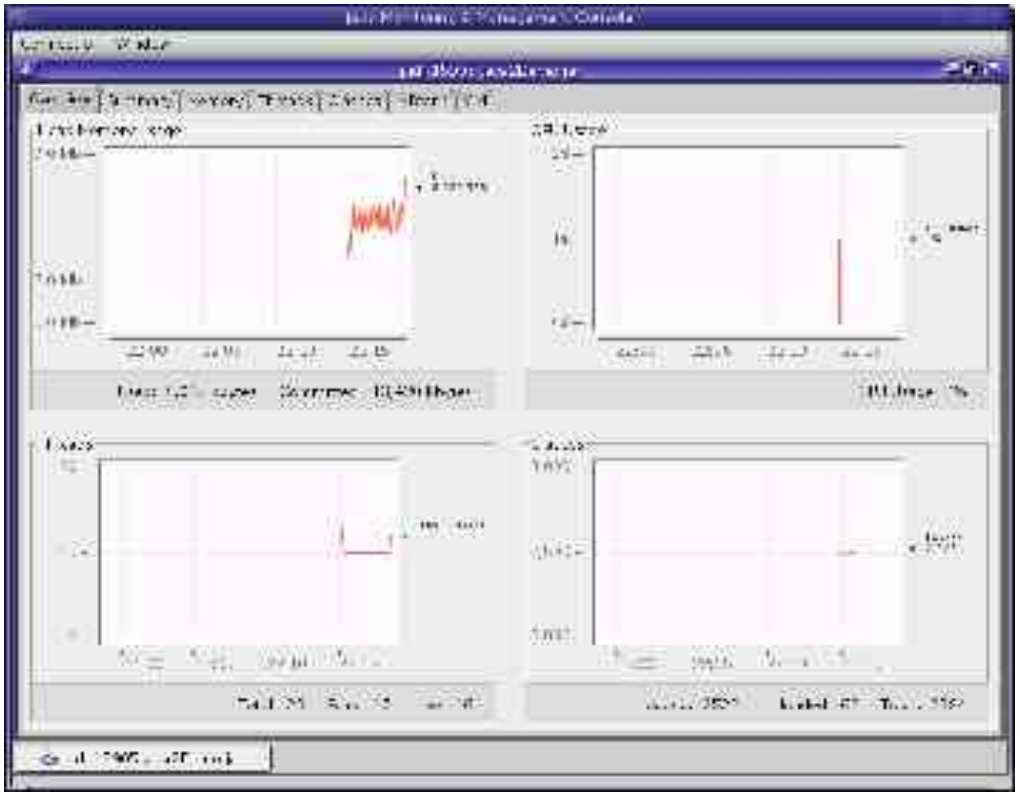
```
% javac -cp .:./jaxb/lib/jaxb-api.jar\
:./jaxb/lib/jaxb-impl.jar\
:./jaxb/lib/jsr173_1.0_api.jar\
:./jaxb/lib/activation.jar \
Sum.java
```

Class-Path Wildcards

```
% javac -cp .:./jaxb/lib/jaxb-api.jar\
:./jaxb/lib/jaxb-impl.jar\
:./jaxb/lib/jsr173_1.0_api.jar\
:./jaxb/lib/activation.jar \
Sum.java
```

```
% javac -cp '.:./jaxb/lib/*' Sum.java
```

JConsole improvements



- Reworked UI
- Attach-on-demand
- Deadlock detector

DTrace

```
hotspot$target:::gc-begin {
    self->ts = vtimestamp;
}

hotspot$target:::gc-finish/self->ts/ {
    @ = quantize(vtimestamp - self->ts);
    self->ts = 0;
}
```

```
   value  ------------- Distribution ------------- count
16777216 |                                         0
33554432 |@@                                       1
67108864 |@@@@@@                                   3
134217728 |@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ 16
268435456 |                                        0
```

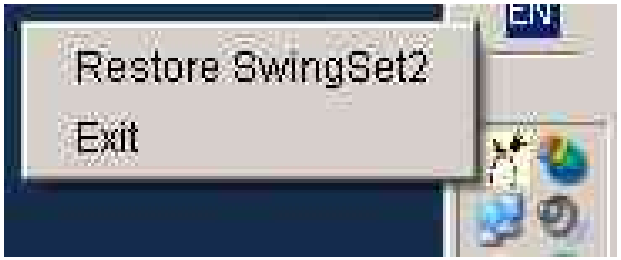
java.awt.Desktop

☑ 138
votes

```
import java.awt.Desktop;

void launchBrowser(URI u) {
    Desktop dt = Desktop.getDesktop();
    dt.browse(u);
}
```

java.awt.{SystemTray,TrayIcon}



Anti-Aliased LCD Text

☑ 378
votes

http://mustang.dev.java.net

How to Get Started

1) Sign up on java.net, click thru JRL, sign, return contribution agreement
2) Download, explore, get it to build on your system
3) Check bug parade or try fixing that bug you really hate
4) Start simple, pick something self-contained, learn the process
5) Submit your fix: https://jdk-collaboration.dev.java.net/
6) Work with the Sun engineer on that module to test, validate, and integrate your fix
7) Feel good about what you've done!

“How To” Tutorials People Wrote

- Building JDK 6.0 on Windows XP
  - http://blogs.sun.com/roller/page/kto?entry=building_the_jdk_6_0
- Java and "Open Source": Good News and Bad News - By Andy Tripp
  - http://www.javalobby.org/articles/fixing-the-jdk/

Contributions In Process

4094886:	java.io: Need a way to convert Readers into InputStreams
4306897:	Add java.util.Arrays.binarySearch(a, key, fromIndex, toIndex)
4346256:	Provide an AbstractTreeModel for the TreeModel hierarchy
5015163:	String merge/join facility that would be the inverse of java.lang.String.split()
5025230:	Creating thread local variables from within ThreadLocal.initialValue()
6176992:	Add support to java.lang.Class for wrapper type conversions
6182942:	JButton.isEnabled() return false however the button is enabled
6197726:	IdentityHashMap.EntrySet.toArray(T[] a) is incorrectly implemented
6205522:	Javadoc warnings for GregorianCalendar
6232484:	ArrayList made from IdentityHashMap.entrySet() fails to create properly
6245410:	javax.swing.text.html.CSS.Attribute: BACKGROUND_POSITION, TEXT_DECORATION is not w3c spec compliant
6246565:	File descriptor leak when using DatagramChannel.socket()
6248507:	AbstractStringBuilder.replace does not handle count < start < end
6254531:	ThreadLocal leak when value references ThreadLocal

The Future of Java EE
Glassfish

What is GlassFish?

- Sun's Open Source Application Server Platform Edition 9
  - CDDL license
  - Open process
- Open access to code and binaries
  - CVS access to source code
  - Nightly builds, weekly promoted builds
- Must support Java EE compatibility
- Renewed partnership between Sun and the larger enterprise Java community

Common Development and Distribution License (CDDL)

- Takes the well known Mozilla Public License (MPL) and makes it reusable without modification
- Clear, consistent, not burdensome for contributors
- As reusable and general as possible
- Meets the definition of Open Source Definition of Open Source Initiative
- "...CDDL... Everything is in place for it to work well." Linus Torvalds 2/1/05

Why Join the GlassFish Project

- Contribute to world class application server
  - Involved in the implementation of cutting edge Enterprise Java, Web Services, SOA standards, Grizzly
  - Easy build environment
- Access to nightly and promoted builds
  - New Java EE 5 features
  - Bug fixes
- Interact with a great group of people inside and outside of Sun
- Fully Participative Community
  - Already non-Sun committers

How to Get Involved

- Join discussions on mailing lists and forums
- Download and try the binary builds
- Submit bugs or request for enhancements
  - IssueTracker
- Submit a patch to fix a bug or provide an enhancment to issue from
  - IssueTracker
  - Web Bugs
- Checkout the GlassFish people page
  - http://wiki.java.net/bin/view/Projects/GlassFishPeoplePage

GlassFish Supports

- Supports all platforms
  - Solaris, Solarix_X86, Windows, Linux and MacOS
- Java Blueprints Solution Catalog EA release runs on GlassFish
  - includes design guidelines, code and applications for JavaServer Faces and AJAX
- NetBeans 5.0 runs on GlassFish
  - You can use NetBeans to develop applications on GlassFish

Building GlassFish

- Build steps to follow
  - Need Maven, JDK 5 and cvs client
  - Member of GlassFish and accept license
  - Checkout bootstrap module
  - Edit property files
  - Checkout rest of server
  - Build and configure server
- Full set of instructions on GlassFish site -really quite easy to build
  - https://glassfish.dev.java.net/public/BuildGlassFish.html

Code organization

- Modules - way of dividing up the app server to help navigate the code
- Web tier and most of online administration are available
- EJB 3.0 pieces are being added
  - Session bean code is in
  - Persistence code is expected to be added very soon
- Others coming soon
  - JMS, JavaMail

The J2EE Challenge

- J2EE is enormously powerful
  - The industry standard for robust enterprise apps
- But that power sometimes gets in the way
  - Too difficult to get started
  - Even simple apps need boring boilerplate
- Can we keep the power, but make typical development tasks simpler?
- **YES**: and that is the focus of Java EE 5!

Ease of Development Improvements in Java EE 5

- POJO-based programming (Plain Old Java Object)
  - More freedom, fewer requirements
- Extensive use of annotations
  - Reduced need for deployment descriptors
- Resource Injection
  - Inversion of control
- New APIs and frameworks

Java EE 5 Major Features

- Simplified web services support
- More web service standards support
- Greatly simplified EJB development
- New persistence API
- Easy web applications with JavaServer Faces

Java EE 5

- EJB 3.0 (JSR-220)
- JavaServer Faces (JSR-252)
- StAX (JSR-173)
- Web Services Metadata (JSR-181)
- New persistence API (JSR-220)
- JAXB (JSR-222)
- JAX-WS (JSR-224)
- Common Annotations (JSR-250)

EJB 2.1

- Very powerful, but complex to use
  - Too many classes, interfaces
  - Java Naming and Directory Interface (JNDI) API lookups
  - javax.ejb interfaces
  - Awkward programming model
  - Deployment descriptors
  - Entity bean anti-patterns
  - ...

Ease of Development Features in EJB 3.0

- Fewer classes and interfaces
- Dependency injection
- Simple lookups
- No required container interfaces
- No required deployment descriptor
- Simplified persistence
- Object/relational mapping

EJB 3.0 Example

```
// Same example, EJB 3.0

@Stateless public class PayrollBean
        implements Payroll {

    @Resource DataSource empDB;

    public void setBenefitsDeduction (int empId, double
    deduction) {
       ...
       Connection conn = empDB.getConnection();
       ...
    }

    ...
}
```

Dependency Injection

- Resources a bean depends upon are injected when bean instance is constructed
- References to:
  - EJBContext
  - DataSources
  - UserTransaction
  - Environment entries
  - EntityManager
  - TimerService
  - Other EJB beans
  - ...

Dependency Injection

- Annotations
  - @EJB
    - References to EJB business interfaces
    - References to Home interfaces (when accessing EJB 2.1 components)
  - @Resource
    - Almost everything else
  - Number of annotations is simplified from EJB 3 specification early draft
- Injection can also be specified using deployment descriptor elements

Simplified Client View

- Session beans have plain Java language business interface
  - No more EJB(Local)Home interface
  - No more EJB(Local)Object interface
- Bean class implements interface
  - Looks like normal Java class to bean developer
- Looks like normal Java language interface to client (POJI)

EJB 3.0 Client Example

```
// EJB 3.0 client view

@EJB ShoppingCart myCart;

    ...

    Collection widgets = myCart.startToShop("widgets");


    ...
```

Persistence Goals of EJB 3.0

- Simplify entity bean programming model
- Support for light-weight domain modeling, including:
  - Inheritance and polymorphism
- Complete query capabilities
- Support for object/relational mapping specification
- Make entity instances usable outside the EJB container

Persistence Model in EJB 3.0

- Entities are simple Java classes
  - Concrete classes—support use of new
  - Getter/setter "property" methods or persistent instance variables
  - No required bean interfaces
  - No required callback interfaces
- Usable as "detached" objects in other application tiers
  - No more need for DTOs (Data Transfer Objects)

EntityManager

- EntityManager serves as untyped “home” for entity operations
- Methods for lifecycle operations
  - Persist, remove, merge, flush, refresh, etc.
- Similar in functionality to Hibernate Session, JDO PersistenceManager, etc.

Persistence Focus: O/R Mapping

- Ease-of-use facility for Java developers mapping domain object model to a relational database
- Developer is aware of mapping between DB schema and domain object model
  - Developer is in control of mapping
  - Developer can rely on mapping and its semantics
- Mappings may be expressed using metadata annotations or XML
  - Default mappings provided

EJB 3.0/2.x Technology Interoperability and Migration

- Applications written to EJB 2.1 specification and earlier work unchanged in EJB 3.0 containers
- Migration path to EJB 3.0 APIs
  - New applications can be clients of older beans
  - Older clients can be clients of new EJB 3.0 components

Java Support for Web Services

- JAX-RPC 2.0 renamed to JAX-WS 2.0 (Java API for XML Web Services)
- Implements new WS stack
  - JAX-WS 2.0 and JAXB 2.0
  - Designed for growth (JAX-WSA, others)
- Heavy use of annotations
  - Ease-of-development
  - Portability
- Supports Fast Infoset
  - Better Performance

JAX-WS and JAXB Integration

- JAX-WS delegates all data binding to JAXB
- Development time
  - JAXB generates Java types from a WSDL's schemas
  - JAXB generates the WSDL's schema from Java types
- Runtime
  - JAX-WS un/marshalls the message (soap:envelope)
  - JAXB un/marshalls the payload (soap:body child, soap:header and soapfault elements)

JAXB 2.0 Is Now Bi-Directional

- 1.0: Schema → Java only
  - JAXB is for compiling schema
  - Don't touch the generated code
- 2.0: Java → XML + schema compiler
  - JAXB is about persisting POJOs to XML
  - Annotations for controlling XML representation
  - Modify the generated code to suit your taste

JAX-RPC 1.1 RI Issues

- Supports only SOAP 1.1over HTTP
- Limited XML Schema support
- Little Control of Mapping Java and WSDL/ XML Schema
- Large non-portable applications
- Large runtime
- Web Service development is too complex
- Servlet container is required

JAX-WS 2.0 New Architecture

- Multiple protocols
  - SOAP 1.1, SOAP 1.2, XML
- Multiple encodings
  - XML, MTOM/XOP, FAST Infoset (Binary XML)
- Multiple transports
  - HTTP
  - Others to be added in future releases
- Data binding
  - Uses JAXB 2.0 (100% XML Schema)

Web Services Annotation

- Java class is annotated with @WebService
  - Becomes wsdl:port-type element in the WSDL world
- Java Methods are annotated with @WebMethod
  - Becomes a wsdl:operation element in the WSDL world
- Java Methods parameters mapping to XML is now handled by JAXB
  - wsdl:message part element types are now defined in a separate schema file imported by the wsdl

Web Service Example

```
@WebService(name="MySimpleWS");
public class RandomClass {

   @WebMethod
   public String sayHello(String s) {...}

   public void unpublished() {...}
}
```

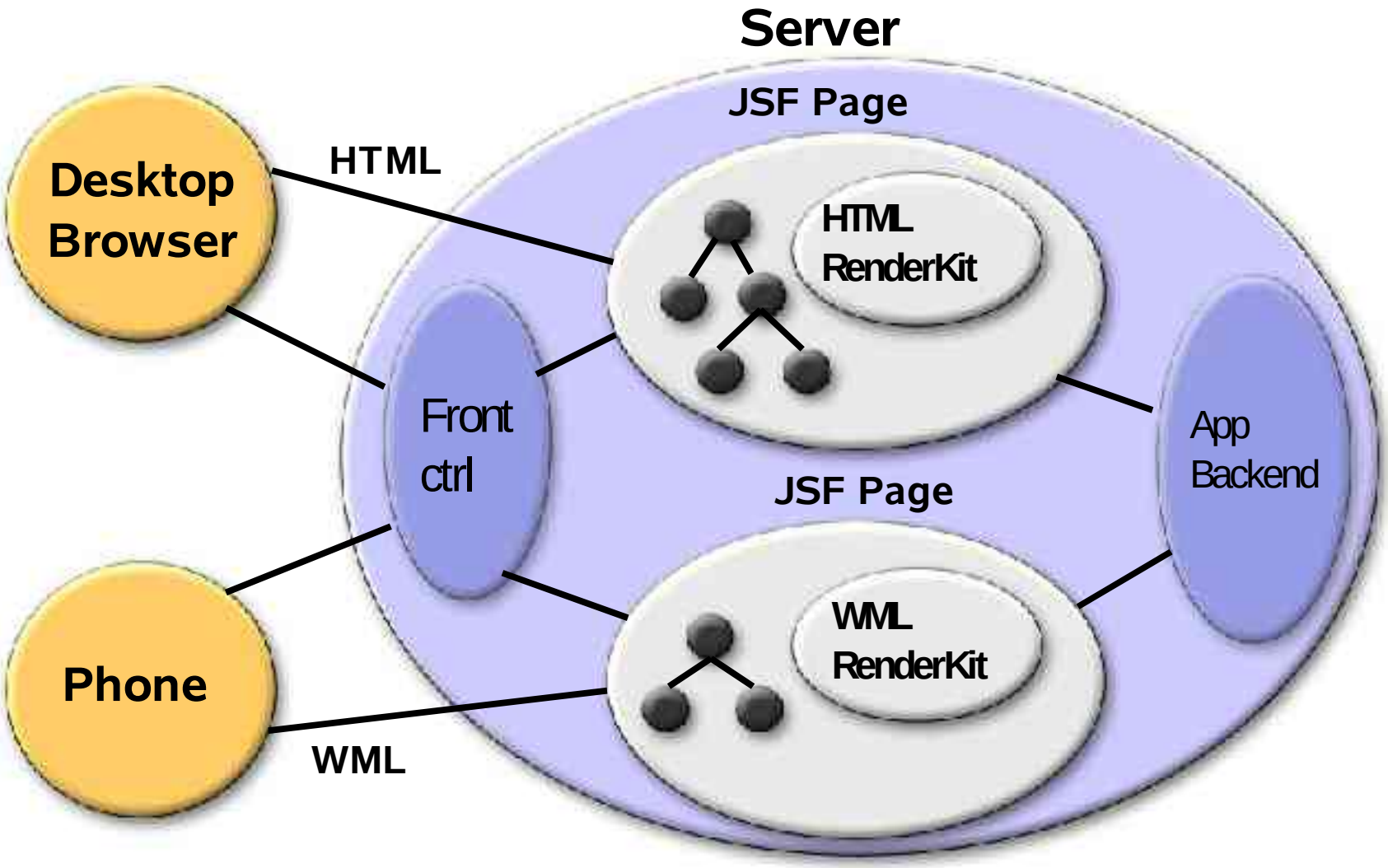
Web Service Client Example

```
@WebServiceRef(
    wsdlLocation=
        "http://localhost:8080/SayHelloService?WSDL");)
static javaone.SayHelloService wsService;

public static void main(String[] args) {

    javaone.SayHello wsPort = wsService.getHello();
    wsPort.sayHello("JavaOne Attendees");
}
```

JSF Architecture



Value Add's of JavaServer Faces

- Extensible UI component model
- Flexible rendering model
- Event handling model
- Validation framework
- Basic page navigation support
- Internationalization
- Accessibility
- Tool friendly

Java EE 5 Status

- Most specs available for review now
- Specs at Proposed Final Draft stage – Q3 2005
- Java EE 5 SDK Beta release – Q4 2005
- Java EE 5 Final release – Q1 2006

Resources and Summary

Summary

- Get involved!
- Join the Sun Developer Network
- Download and try Mustang
- Download and try Glassfish
- Java needs you!

Resources

- Java.net
- http://mustang.dev.java.net
- New features of Mustang
  - http://java.sun.com/developer/technicalArticles/J2SE/Desktop/Mustang_build39.html
- http://glassfish.dev.java.net
- http://java.sun.com/javaee

Java Technology Community Development: Peabody and GlassFish

Simon Ritter
Java Technology Evangelist
Sun Microsystems, Inc.
simon.ritter@sun.com