The Java Development Environment on Gentoo

FOSDEM 2005
February 27
Brussels

Karl Trygve Kalleberg

<karltk@ gentoo.org>
The Java Team

● Who we are
  – Thomas Mathijs (axxo)
  – Jochen Maes (SeJo)
  – Jan Brinkmann (luckyduck)
  – Saleem Abdulrasool (compnerd)

● Who we will (probably) also be
  – Vibhav Garg (vgarg?)
  – Petteri Räty (Betelgeuse)
Our responsibilities

- What we're (not) paid do
  - maintain 300+ packages
  - maintain Java browser plugins
  - maintain all JDKs
  - maintain Eclipse, NetBeans, Tomcat, …
  - on x86, ppc, amd64, alpha and (sparc)
Other responsibilities

- **What we really do**
  - maintain the Java Ebuild Policy
  - maintain the Gentoo Java (sub-)Wiki
  - maintain Java Experimental
  - discuss important developments and philosophical issues on #gentoo-java-scifi
  - maintain `java-config`
Java Development

• Why would you use Java?
  – Beats me, but some of us are simply forced to.

• So how can we put up?
  – One-line installation of any Java library or tool
  – Have full Javadoc, source code and dependency info instantly available
  – Up-to-date, customizable, enterprise-ready, built to fit, re-finance your mortgage now.
Java Development Environment

- Healthy set of JDKs
  - Commercial: Sun, Blackdown, IBM, Jrockit, Compaq
  - Free: Kaffe, SableVM, Cacao, Kissme

- Good set of basic libraries
  - Java GNOME, JBoss, Tomcat, SWT, commons-*, jakarta-*

- IDEs
  - NetBeans (in progress), Eclipse (in regress)

- Other: Ant, Jython, Scala, (even Maven)
Incompleteness

• Not enough tools
  – Substantial parts of the “enterprise stack” missing.
  – Annoying gaps in package set and features for desktop app development.

• Primarily focused on Java developers
  – Growing set of end-user apps written in Java
    • Azureus, Eclipse, Browser plugins, …

• Not compatible with Java 1.5, err, Tiger, err, 5.0
  – Need merge-time, per-package hotswitching of JDKs
Future Development Environment

• Practically everything should be compiled from source
  – Maven from source
    • Sucks, should be scourged from the surface of the planet
      – Luckily vgarg is tackling this sorry excuse for a project
  – Enterprise development
    • Spring, AppFuse, Tomcat, JBoss, ...

• Numerous advantages
  – Packages take more space
  – More difficult to package
Native Compilation

- Probably using gcjx (only in gcc 4.0.x)
- Native Eclipse
  - Cooperation with RedHat/Fedora and Debian
- Native GNOME in Java
  - Cooperation with Java-GNOME
    - In particular, Andrew F. Cowie (AfC)
- Native libraries
  - Must retain .jar files
Conclusion

- Decent base, easy part is done
- Difficult design issues ahead
- Heaploads of work left
  - (for luckyduck, compnerd and axxo)
  - Especially for Enterprise Java (next)