Ji Gong

Freedom vs. FUD

Julien Gouesse
Sven Gothel
Xerxes Ranby
Who are u?

Programmers
Computer exorcists
Users

\( \text{JogAmp Community} \)

http://jogamp.org
What do u do?

- Java Graphics, Audio, Media & Processing
  High Performance Bindings
- One Stop Community Platform
  - SCM, Bugtracking, Build Server, Mailinglist/Forum,..
- Open & Vendor Independent
- BSD License

Our babies

- Gluegen
- JOAL
- JOGL
- JOCL
Slides and BOF Video will be made available on jogamp.org.
What the heck is Ji Gong?

The Buddhist monk Daoji (Ji Gong): openly ate meat & drank wine & wear tattered and dirty robes from traveling.

However, he was kind hearted & roamed the streets and helped people whenever he could.
What the heck is Ji Gong?

It does **not**:

- Implement a VM technology
- Create a new Ecosystem
- Involve Lawyers :)

It **does**:

- Build upon existing work (JamVM, Hotspot, ..)
- Ensure availability on *all* platforms and devices, for real.
- Enhance parts to enable them across devices (*IcedTea-Web* ..)
- Use alternative VM Tech (Dalvik, RoboVM, LLVM-IR ..)
Don't u know there's something called OpenJDK and IcedTea?

IcedTea – our freedom fighter:

- Breathed life into the code dump by Sun Microsystems.
- Added missing parts
- Enabled building the beast on Unix platforms
- Supports mobile CPU architectures
- Merges alternative VM implementations [usable for mobile]
- Creates and supports a JNLP and Browser Plugin (deployment).

GPLv2 Ecosystem
Don't u know there's something called OpenJDK and IcedTea?

However,

*intent* and *priorities* in the corporate world

what users would like to experience

*Freedom*-wise and *technology*-wise

But hey!

Isn't it a *free software*?

*Can't we act*?
Why do u still create Ji Gong?

IcedTea

GPL

Ji Gong

4 Freedoms

- for any purpose ..
- change it .. as you wish ..
- redistribute copies
- distribute .. modified versions ..

OpenJDK

Oracle ?

Fear, Uncertainty, Doubt ?

Patent grant for desktop only
Mobile use *may* be prohibited?
No explicit nor official statement?

Freedom vs. FUD

FOSDEM'14
Why do u still create Ji Gong?

IcedTea

OpenJDK

Oracle?

GPL

Ji Gong

4 Freedoms

- of speech
- of worship
- from want
- from fear

--- Bill of Rights!

Free Software, as in freedom!

Fear, Uncertainty, Doubt?

Patent grant for desktop only
Mobile use may be prohibited?
No explicit nor official statement?

FOSDEM'14
Why do u still create Ji Gong?

Ji Gong

4 Freedoms
VM technology across platforms and devices

OpenJDK

Oracle?

Fear, Uncertainty, Doubt?

No OpenJDK builds
No IcedTea-Web builds

for
Windows?
OSX?
Android?

...
Technical enhancement?

- **Availability** of GPLv2 based OpenJDK / IcedTea Runtime Environment (RT)
- Web Plugin based on IcedTea-Web (JWeb)
- Source Certification Contract (SCC)
Technical enhancement?

- **Availability** of GPLv2 based OpenJDK / IcedTea Runtime Environment (RT)

  Linux
  Windows
  OSX
  ...

  Desktop
  Mobile

  VM CPU support

  Intel/AMD 32bit & 64bit
  ARM based CPUs .. etc

  Android, Jolla, ..
  Tablet OS
  Embedded devices
Technical enhancement?

- Web Plugin based on IcedTea-Web (JWeb)
  - NativeWindow based plugin **without** AWT - **Applet3**
  - Allow utilization of any native Windowing Toolkit
  - Enable Applets on mobile devices

**Browser**

**Plugin**
- RT(vm, libs)
- Plugin-Code

**NativeWindow**

**User Applet**
- NEWT / … ?
- JOGL / .. ?

```
git clone git://jogamp.org/srv/scm/jigong/icedtea-web.git
```
Applet3

Implemented by user.

Applet3 Lifecycle

- createNativeWindow(Applet3Context, NativeWindowUpstream)
- init(Applet3Context)
- start()
- stop()
- destroy()

public interface Applet3 {

    /**
     * Implementation creates a native child window, allowing to be controlled by the plugin.
     * <p>
     * The applet's child window is destroyed by the plugin after it has called {@link #destroy()}.<p>
     * @param context the {@link Applet3Context}
     * @param parent the parent {@link NativeWindowUpstream}, reflecting the plugin's native applet window.
     * @return {link NativeWindowDownstream} users native child window.
     */
    NativeWindowDownstream createNativeWindow(Applet3Context context, NativeWindowUpstream parent);

    void init(Applet3Context context);
    void start();
    void stop();
    void destroy();
}
/**
 * Representing the plugin window, i.e. the user applet's parent window.
 */

public interface NativeWindowUpstream {
    String getDisplayConnection();

    int getScreenIndex();

    long getWindowHandle();

    int getWidth();

    int getHeight();

    void notifySurfaceUpdated(NativeWindowDownstream swappedWin);
}
Applet3

Implemented by user.

Representing the user applet child window, which is controlled by the plugin.

```java
public interface NativeWindowDownstream {
    void destroy();

    NativeWindowUpstream getParent();

    long getWindowHandle();

    void setSize(int width, int height);

    void requestFocus();

    void setVisible(boolean v);

    void display();

    void notifyPositionChanged(NativeWindowUpstream nw);
}
```
Applet3

Provided by Plugin implementation.

```java
public interface Applet3Context {
    /**
     * Returns the {@link Applet3} bound to this context
     */
    Applet3 getApplet();

    String getAppletName();

    String getParameter(String name);
    URL getDocumentBase();
    URL getCodeBase();

    /**
     * Requests that this applet be resized.
     * @param width the new requested width for the applet.
     * @param height the new requested height for the applet.
     */
    void resize(int width, int height);

    void showDocument(URL url);

    void showDocument(URL url, String target);

    void showStatus(String status);

    ...
```
Technical enhancement?

- Source Certification Contract (SCC)
  - Trust the **Source**, Users!
  - Not yet based solely on source code (**toolchain**)
  - Replace Certification Authority (CA) with Chain of Trust
    - Use gpg, pgp, ..
    - Why **trust** a company and force users to pay?
  - Store binary signatures inside public accessible source code repositories
  - Gracefully revoke binary builds with security flaws:
    - Revoke compromised binary signatures instead of author's Identity
    - Fine Grained Release Control
  - Ease access to the binary's source code
Who benefits from Ji Gong?

• Users
  - Unified Experience across Devices
  - Enhanced trust in the source code and its developers!

• Platform Designers (devices)
  - Clarify licensing issues
  - Confidence in choosing a VM technology
  - Enhanced collaboration with community

• Software Developer
  - Ease collaboration with the community
  - Revoke the binary instead of the author's identity
Thank You & Love You

OpenJDK

IcedTea

IcedTea-Web

Sven Gothel

Julien Gouesse

Xerxes Ranby

Rami Santina

Harvey Harrison

Mark Raynsford

Wade Walker

... all the many contributors & users