Apache Geronimo 3.0 Deep Dive

Kevan Miller, IBM

kevan@apache.org

Nov. 9, 2011

Who Am I

- Kevan Miller

- Employer: IBM

- Active at the ASF since 2005:
  - Geronimo (PMC Chair)
  - OpenEJB, Aries, OpenWebBeans
  - Mentoring Incubator Podlings:
    - Bean Validation, Kitty, SIS, VCL, Wink

Agenda

- Discuss Geronimo 3.0

  - Introduction

  - Advanced configuration

  - Advanced commands

  - Custom assemblies
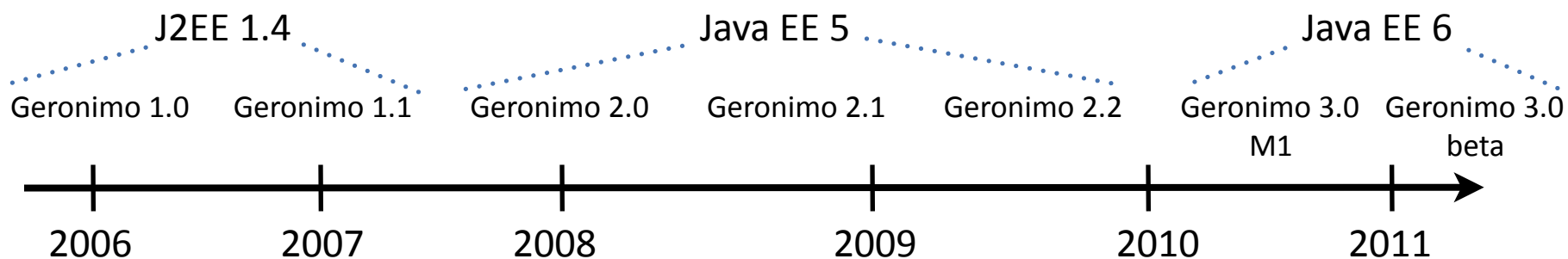
  - ClassLoading

Geronimo Overview

- ASF project started in 2003

- Mission:

  Create an Open Source application server
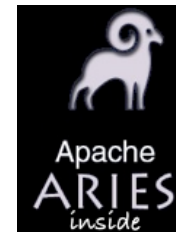 *from Best-of-Breed Open Source components*

- Release history:

J2EE 1.4		Java EE 5			Java EE 6	
Geronimo 1.0	Geronimo 1.1	Geronimo 2.0	Geronimo 2.1	Geronimo 2.2	Geronimo 3.0 M1	Geronimo 3.0 beta
2006	2007	2008	2009	2010	2011	

Apache Geronimo

APACHE
GERONIMO

APACHE
GERONIMO

CXF

Apache
ARIES
inside

Karaf

OpenEJB

wink

Tomcat

jetty://

ActiveMQ

Derby

Bean Validation

axis

OpenWebBeans

ApacheCon
North America 2011
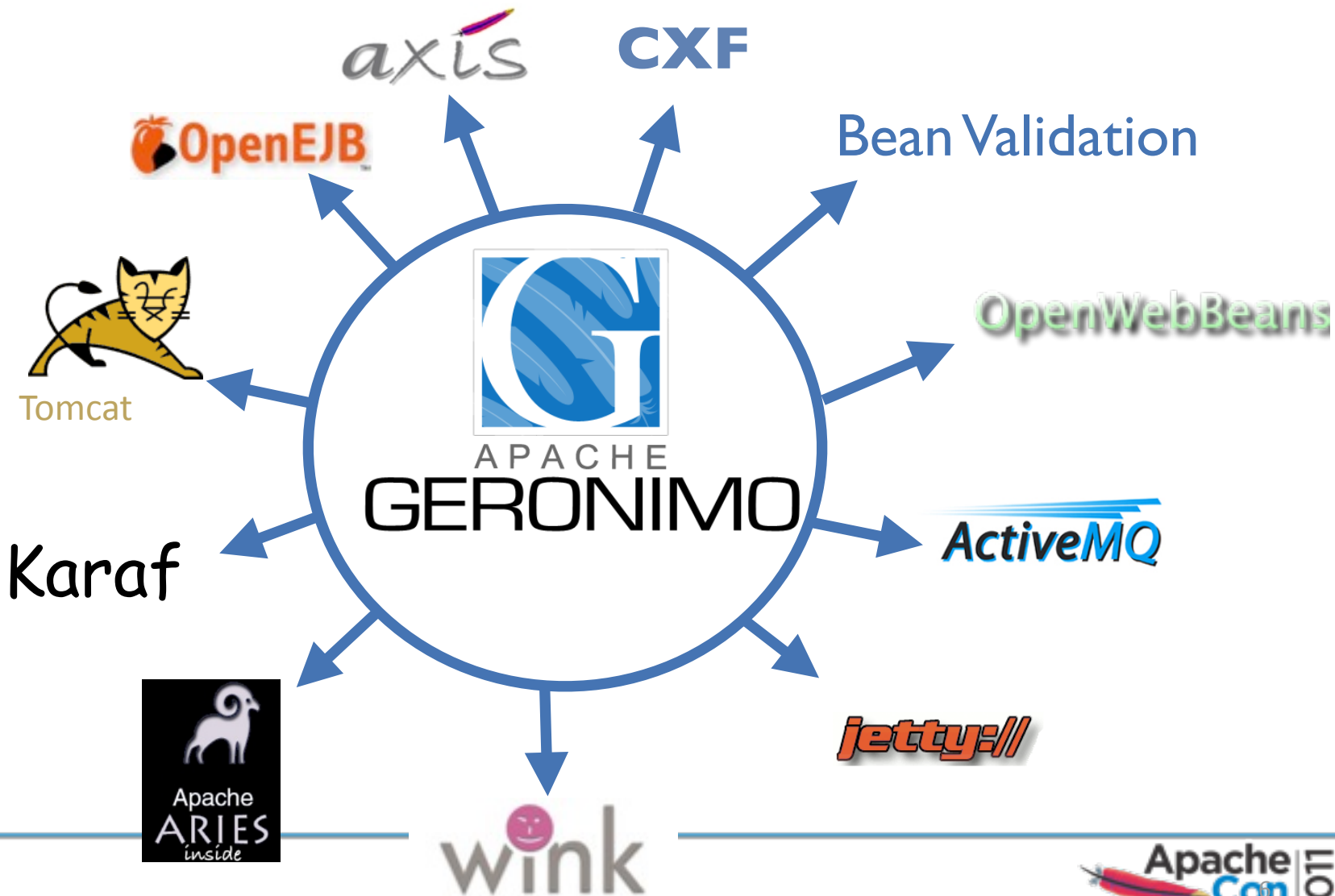
Apache Geronimo

Geronimo 3.0

- 3.0-beta release vote underway
  - Java EE 6 (full and web profile)
  - OSGi Enterprise Applications (Aries)
  - OSGi Kernel Restructure

Geronimo Directory Layout

Directory	Description
`bin/`	Geronimo commands (e.g. deploy, geronimo run, …)
`deploy/`	Hot-Deploy for Java EE apps
`etc/`	OSGi / Karaf configuration
`hot-bundles/`	Hot-Deploy for OSGi bundles
`lib/`	Small set of geronimo and karaf jars
`repository/`	Maven-format repository
`schema/`	Schemas for geronimo deployment plans
`var/`	geronimo, tomcat, logging configuration files

OSGi Configuration

- **`etc/`**
  - **config.properties**
    - choose framework implementation (felix or equinox). Current default is equinox (OSGi 4.3)
    - OSGi bootdelegation packages
    - supported platforms (i.e. Java 6 and Java 7)
  - **startup.properties**
    - defines core bundles and their start levels

Tomcat Configuration

- **var/catalina**

  - server.xml

    - configuration of the Tomcat servlet container

  - conf/web.xml

    - defines default values for apps in the container

Security Configuration

- **var/security**
  - groups.properties
    - defines the security groups
      - e.g. admin=system
  - users.properties
    - defines user/passwords
      - e.g. **monitor={Simple}rO0ABXNyABlqYXZheC5...**
      - **{Simple}** represents the default encryption/obfuscation that will be performed
      - encryption is user customizable

Server configuration

- **var/config**
  - config.xml
    - defines the Geronimo modules/services that will be started and also GBean configuration overrides

```
...
<module name="org.apache.geronimo.configs/uddi-tomcat/3.0-SNAPSHOT/car">
    <gbean name="DerbyProperties">
      <attribute name="systemProperties">
        uddi.openjpa.ConnectionURL=jdbc:derby://localhost:${DerbyPort + PortOffset}/UddiDatabase;create=false;
      </attribute>
    </gbean>
  </module>
...
```

Configuration Overrides

- **`var/config/config-substitutions.properties`**
  - Simple property file to configure commonly changed configuration options:

```
NamingPort = 1099
ServerHostname = 0.0.0.0
PortOffset = 0
MaxThreadPoolSize = 500
MinThreadPoolSize = 200
JMXPort = 9999
HTTPPort = 8080
HTTPSPort = 8443
```

Logging

- **var/log/**
  - contains log files and log configuration

    server-log4j.properties

    client-log4j.properties

    deployer-log4j.properties

Interactive Shell

- If Geronimo is started in foreground, an interactive shell environment is started

- Provides:
  - Geronimo commands: start/stop, deploy/undeploy, etc.
  - OSGi bundle commands
  - equinox:diagnose
  - Shell commands

Geronimo Plugins

- Plugins are bundles with additional meta-data

- Define dependencies on bundles and other plugins

- Plugin installation installs all dependencies

- Dependencies form a DAG, determining:

  ‣ startup order, service visibility

  ‣ ClassLoading hierarchy
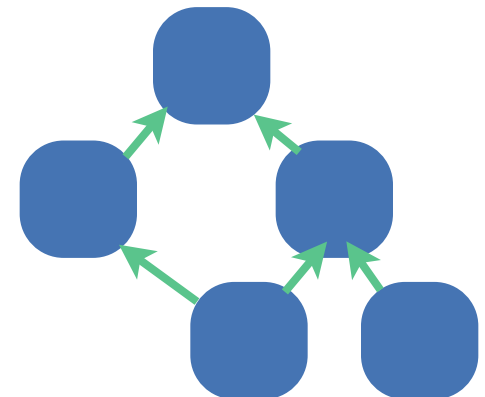
Geronimo GBeans

- GBeans are a Geronimo-specific service model that support JSR-77 (J2EE Management)

- Attributes, single/multivalued service references

- References supported within and between Plugins

- OSGi bundles used for classloading, but GBeans are managed by Geronimo kernel

- Easy to expose GBeans as OSGi services and OSGi services can be exposed as GBeans
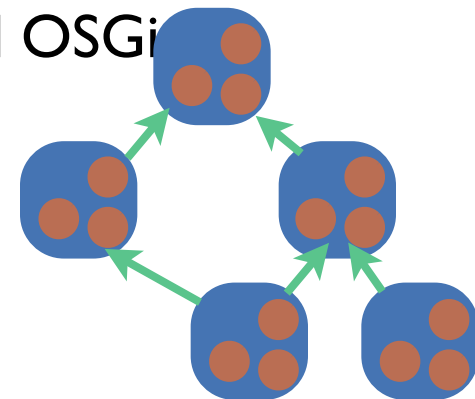
Custom Assemblies

- Custom assemblies allow server capabilities to be customized to a specific feature set

- "Profile" is a pre-configured assembly

Framework Assembly
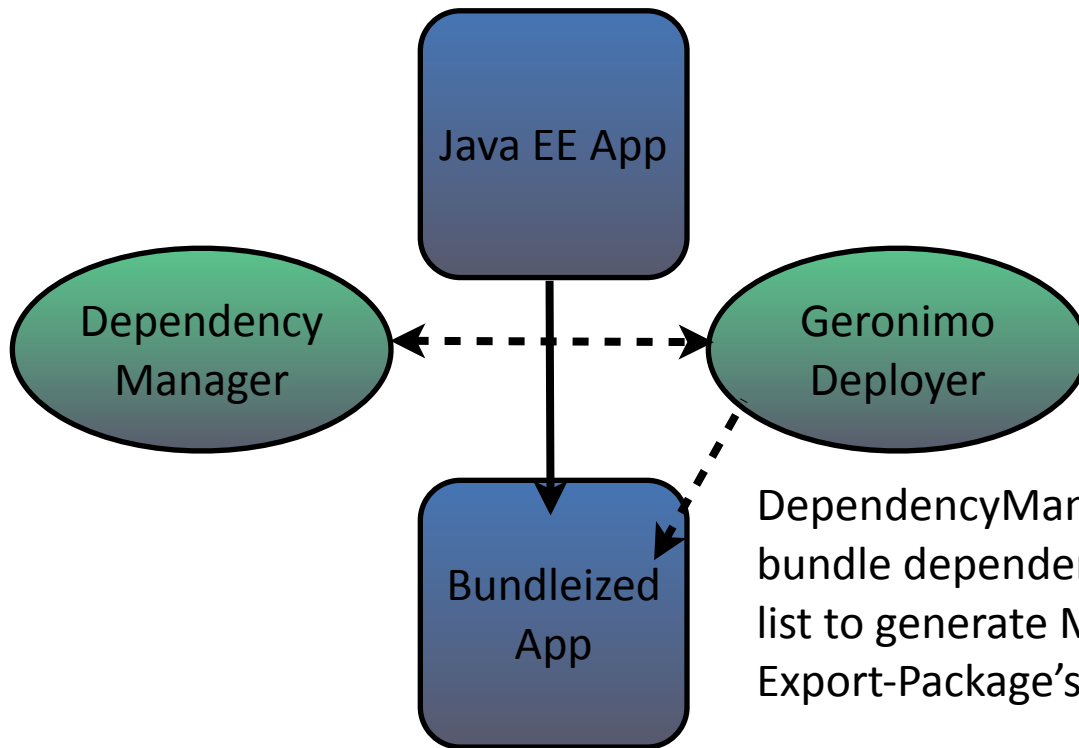
Web Profile

Java EE Profile
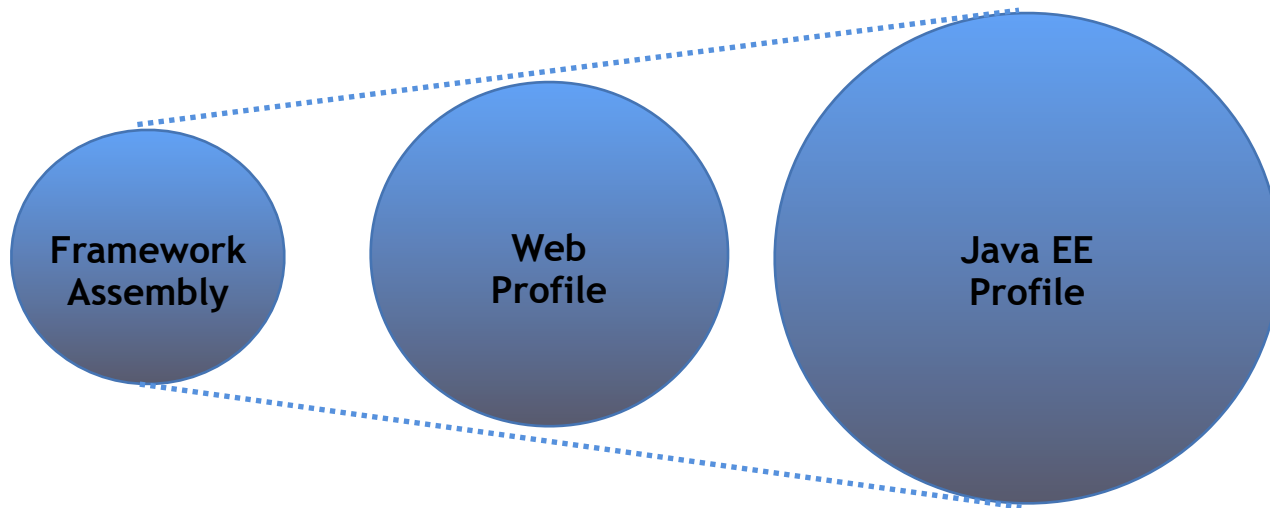
Application Deployment

- All applications are converted to bundle during deployment
  - Geronimo will calculate the dependency information (geronimo-plugin.xml) as well as the manifest.mf

Application Deployment...

Java EE App

Dependency Manager

Geronimo Deployer

Bundleized App

DependencyManager keeps list of all bundle dependencies. Deployer uses this list to generate MANIFEST.MF from the Export-Package's of the dependencies

Manifest.mf
```
Import-Package: org.apache.wink.server.internal.resources,org.apache.w
ink.common.internal.runtime,javax.ws.rs;version="1.1",javax.ws.rs.cor
e;version="1.1",javax.ws.rs.ext;version="1.1",javax.validation,org.ap
ache.bval.jsr303,...
```

<environment> element

```
<web-app xmlns="http://geronimo.apache.org/xml/ns/j2ee/web-2.0.1"
         xmlns:sys="http://geronimo.apache.org/xml/ns/deployment-1.2">

    <sys:environment>
        <sys:moduleId>
            <sys:groupId>org.foo</sys:groupId>
            <sys:artifactId>bar</sys:artifactId>          maven-based naming for
            <sys:version>1.0</sys:version>                module control
            <sys:type>car</sys:type>
        </sys:moduleId>

        <sys:dependencies>
            <sys:dependency>
                <sys:groupId>org.foo</sys:groupId>
                <sys:artifactId></sys:artifactId>         legacy dependency
                <sys:version>1.2</sys:version>            management
                <sys:type>car</sys:type>
            </sys:dependency>
        </sys:dependencies>

        <sys:bundle-activator/>
        <sys:bundle-classPath/>                           OSGi-based mechanism
        <sys:import-package/>                             to control ClassLoading
        <sys:export-package/>
        <sys:require-bundle/>
        <sys:dynamic-import/>

        <sys:hidden-classes/>                             legacy ClassLoading
        <sys:inverse-classloading/>                       controls

    </sys:environment>

</web-app>
```

What's Next?

- Splitting up some deployer dependencies
  - OpenWebBeansDeployer will transitively bring in nearly all deployers/modules

- Go on a diet
  - Will be analyzing the bundles currently included
  - reduce download size and startup time

- Long range/exploratory:
  - Move from GBean architecture to OSGi extender model

Thanks!

Resources

- Geronimo Information:
  - http://geronimo.apache.org
  - https://cwiki.apache.org/GMOxDOC30/documentation.html
- Contact the Geronimo project:
  - user@geronimo.apache.org
  - dev@geronimo.apache.org
  - irc://irc.freenode.net/geronimo

Contact

- Kevan Miller
  - kevan@apache.org
  - kevan.miller@gmail.com
  - Twitter: @kevanmiller