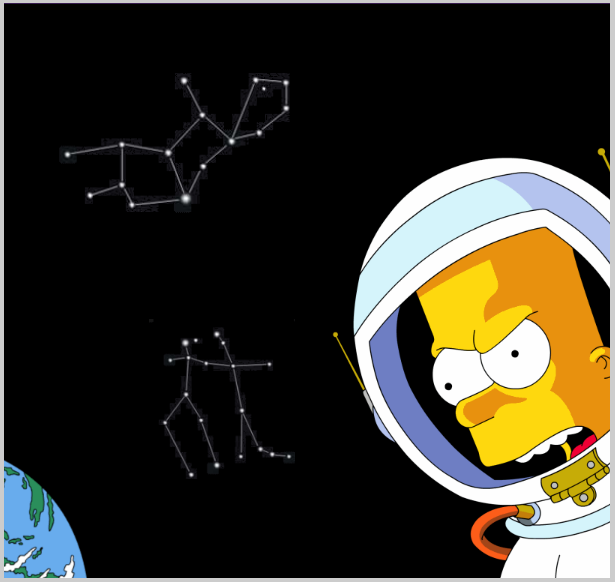
Virgo F2F

Nov-Dec 2010

Southamton, UK



Hristo Iliev
Borislav Kapukaranov
Violeta Georgieva
Krasimir Semerdzhiev
Georgi Stanev

THE BEST-RUN BUSINESSES RUN SAP™

Agenda

1. **File system**
2. Dependencies cleanup
3. Nested frameworks
4. Provisioning
5. Shell
6. Web
7. Security
8. Roadmap

File system

now

Now

- Proprietary structure

Problems

- Eclipse integration
  - ~~target platform~~
  - ~~P2 provisioning~~

ty

earning curve ↑

Relies on Eclipse file structure and indirectly on target platforms (update repositories)

[Using p2 for fun and profit](#)
[Discovering the p2 API](#)

http://help.eclipse.org/helios/index.jsp?topic=/org.eclipse.pde.doc.user/concepts/target.htm

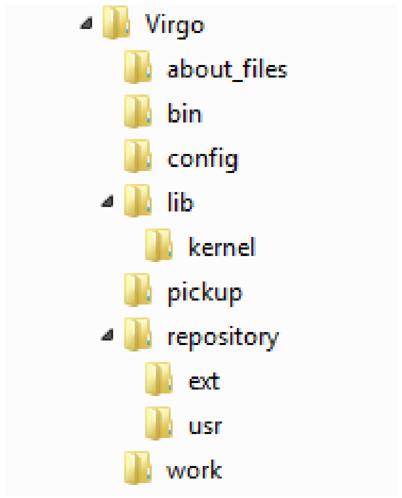
Used to:
- compile
- launch
- calculate dependencies
- state
- other tools

File system

proposal

Now

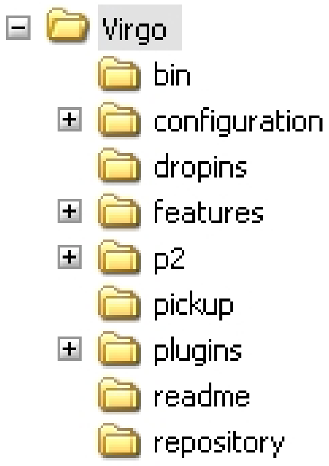
- Proprietary structure



```
Virgo
  about_files
  bin
  config
  lib
    kernel
  pickup
  repository
    ext
    usr
  work
```

Proposal

- Eclipse structure



```
Virgo
  bin
  configuration
  dropins
  features
  p2
  pickup
  plugins
  readme
  repository
```

Open questions

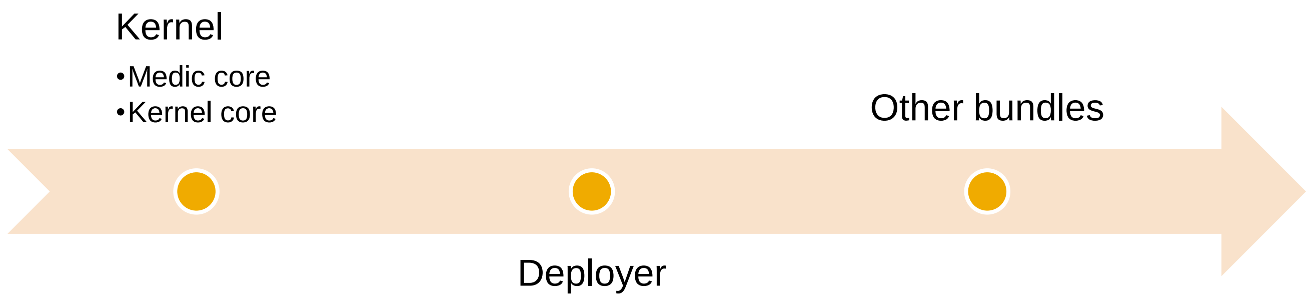
- Plugins ←→ Dropins
- Config ←→ Configuration - merge?

Agenda

Dependencies cleanup

now

- Startup order
  - fixed



- Virgo Kernel
  - Spring DM

Dependencies cleanup

problems

- Startup order
  - **fixed**
  - ~~OSGi practicies~~
  - ~~pluggable~~

- Spring DM in kernel
  - footprint ↑
  - complexity ↑

Dependencies cleanup

proposal

- Startup order
  - dynamic

- Virgo Kernel
  - Replace Spring DM / Blueprint with Declarative services
    - part of equinox
    - footprint ↓
    - complexity ↓

	Size [KB]	Bundles
Blueprint	570 178 32 780	gemini-blueprint-core gemini-blueprint-extender gemini-blueprint-io
Declarative Services	183 77 67 327	org.eclipse.equinox.ds org.eclipse.equinox.util org.eclipse.osgi.services

Agenda

1. File system
2. Dependencies cleanup
3. **Nested frameworks**
4. Provisioning
5. Shell
6. Web
7. Security
8. Roadmap

Nested Frameworks

now

- Virgo regions
  - Kernel
  - User

- Isolation
  - Class loading
  - Versioning

- Equinox
  - Prototype implementation

Nested Frameworks

problems

- Implementation
  - ~~follows specification~~

- Isolation
  - ~~Clear separation app / infrastructure~~
    - ~~API level~~
    - ~~Resources~~
      - ~~threads~~
      - ~~Memory~~
      - ~~…~~

Nested Frameworks

proposal

- Use new RFC 138 – framework hooks
  - ~~user region~~
  - Equinox 3.7 or 3.8?

- Applet-like environment
  - Sandbox isolation
    - API restrictions
      - File / socket
    - resources
      - threads
      - memory
      - …
    - security
  - Applications → kernel
  - ~~Kernel → application~~

Agenda

1. File system
2. Dependencies cleanup
3. Nested frameworks
4. **Provisioning**
5. Shell
6. Web
7. Security
8. Roadmap

Provisioning

p2 as a provisioning concept – P2 Repository

- Unified view of the artifacts – presented as Installable Unit (IU)
- Separate the artifacts form its metadata (IU) – the dependency resolution is based on metadata only.
  - Metadata goes to the metadata repository
  - Artifacts goes to the artifact repository
- IU - a real UNIT
  - What is provided
  - What is required
  - What operations can be applied to it (install, uninstall, copy, configure)

```
-<unit id="MyDemoPlugin" version="1.0.0.201011292309">
  <provided namespace="org.eclipse.equinox.p2.iu" name="MyDemoPlugin" version="1.0.0.201011292309" />
  <provided namespace="osgi.bundle" name="MyDemoPlugin" version="1.0.0.201011292309" />
  <provided namespace="java.package" name="mydemoplugin" version="0.0.0" />
  <provided namespace="org.eclipse.equinox.p2.eclipse.type" name="bundle" version="1.0.0" />
  </provides>
-<requires size="3">
-  <required namespace="osgi.bundle" name="org.eclipse.core.runtime" range="0.0.0" />
  <required namespace="java.package" name="org.eclipse.equinox.p2.core" range="2.0.0" />
  </requires>
- <artifacts size="1">
  <artifact classifier="osgi.bundle" id="MyDemoPlugin" version="1.0.0.201011292309" />
  </artifacts>
  <touchpoint id="org.eclipse.equinox.p2.osgi" version="1.0.0" />
```

Provisioning

p2 as a provisioning concept - Publisher

- Register artifacts to the repository - generating metadata (IU) for them
  - Plug-in and feature publisher
    - The feature is just an IU that requires other IU
  - Product
    - Include features, pluigins
    - Plug-in configuration
  - Product publisher
  - Eclipse integration
    - The publishing functionality is tightly integrated in Eclipse export wizards.

Provisioning

p2 as a provisioning concept – P2 Profile

- P2 Profile

  “Profiles are the target of install/management operations. They are a list of IUs that go together to make up a system. They are roughly equivalent to the traditional Eclipse *configurations*. When an IU is *installed* it is added to a profile. That profile can then be run and the artifacts associated with the installed IUs executed (or whatever). Later the IU can be uninstalled or updated in that profile. The exact same IU can be installed simultaneously in many profiles..”

- Provisioning - change of the profile

(Most problematic part of the concept – can have different plans that finish in the same runtime)

Provisioning

p2 as a provisioning concept – P2 Director

- P2 Director

  “The director is a high level API that combines the work of the planner and the engine. That is, the director invokes the planner to compute the provisioning operations to perform, and then invokes the engine with the planner's output to achieve the desired profile changes. “

- P2 Planer

  “The planner is responsible for determining what should be done to a given profile to reshape it as requested. That is, given the current state of a profile, a description of the desired end state of that profile and metadata describing the available IUs, a planner produces a list of provisioning operations (e.g., install, update or uninstall) to perform on the related IUs.”

- P2 Engine

  “The engine is responsible for carrying out the desired provisioning operations as determined by a director. Whereas the subject of the director's work is metadata, the subject of the engine's work is the artifacts and configuration information contained in the IUs selected by the director. Engines cooperate with repositories and transport mechanisms to ensure that the required artifacts are available in the desired locations. The engine runs by invoking a set of engine Phases and working with the various Touchpoints to effect the desired result.”

Provisioning

p2 as a provisioning concept – Simple Configurator

- Manage runtime state of the installed artifacts
- Bundles.info – description of desired state of installed bundles

```
org.eclipse.equinox.p2.artifact.repository,1.1.0.v20100513,plugins/org.eclipse.equinox.p2.artifact.repository_1.1.0.v20100513.jar,4,true

org.eclipse.equinox.p2.console,1.0.200.v20100601,plugins/org.eclipse.equinox.p2.console_1.0.200.v20100601.jar,4,false

org.eclipse.equinox.p2.core,2.0.0.v20100510,plugins/org.eclipse.equinox.p2.core_2.0.0.v20100510.jar,4,false
```

- The SimpleConfigurator bundle starts first and bring the system to the state described in the bundles.info file

Provisioning

p2 as a provisioning concept – summary

- P2 end to end provisioning process starting form publishing of artifacts and finishing with installing and updating of products

- Extensible
  - Open repository interface
  - Touchpoints and actions

- Integrated in the Eclipse

Provisioning

goals

- Initial provisioning – unziping is a problem when you need to manage different versions of different components
- Extending the product – just a server doesn’t do anything we need of applications on top of it.
- Offline server update.
- Environment specific install (native parts for different OS etc.)

Provisioning

proposal

- Make Virgo installable via P2
  - Virgo P2 Repository
    - Publishing plans, configurations, PAR
  - P2 complained file structure

- Make Virgo updatable via P2
  - Keep the P2 Profile in consistent state
    - P2 as a part of the Virgo kernel
    - Integrate Virgo kernel (deployment) with the p2 Profile Registry

  - Keep Simple Configurator data (bundles.info) in consistent state
    - Integrate Virgo kernel (deployment) with P2 Simple Configurator

- P2 repository as a backend implementing Virgo Repository interface or Virgo specific Touchpoint – for plans, configurations and PARs

Agenda

1. File system
2. Dependencies cleanup
3. Nested frameworks
4. Provisioning
5. **Shell**
6. Web
7. Security
8. Roadmap

Shell supportability

now

- Shell
  - Based on Equinox 3.6.1

- Connectivity
  - Console
  - Telnet

- Supportability
  - VSH
  - Class loading commands

Shell supportability

problems

- Connectivity
  - Telnet
    - Not easy to restrict (localhost)
  - SSH / other protocols
- Usability
  - ~~line editing~~
  - ~~tab completion~~
  - ~~parameter completion~~
- Supportability
  - Class loading commands
    - not all cases covered
    - ~~documentation~~
  - Nested frameworks
    - no support
  - Declarative services
    - incomplete support

Shell supportability

proposal

- RFC147
  - Command Interfaces, Discovery…
  - Apache Gogo - reference implementation
    - Adoption of Equinox 3.7 for Q3 2011
- Connectivity
  - Restrict to IP/hostname (localhost)
- Usability
  - line editing
  - tab completion
  - parameter completion?
- Supportability
  - Improved Class loading commands
    - Search for class (no package)
    - System loader scan
  - Nested frameworks
  - Declarative services

Agenda

1. File system
2. Dependencies cleanup
3. Nested frameworks
4. Provisioning
5. Shell
6. **Web**
7. Security
8. Roadmap

Web

now

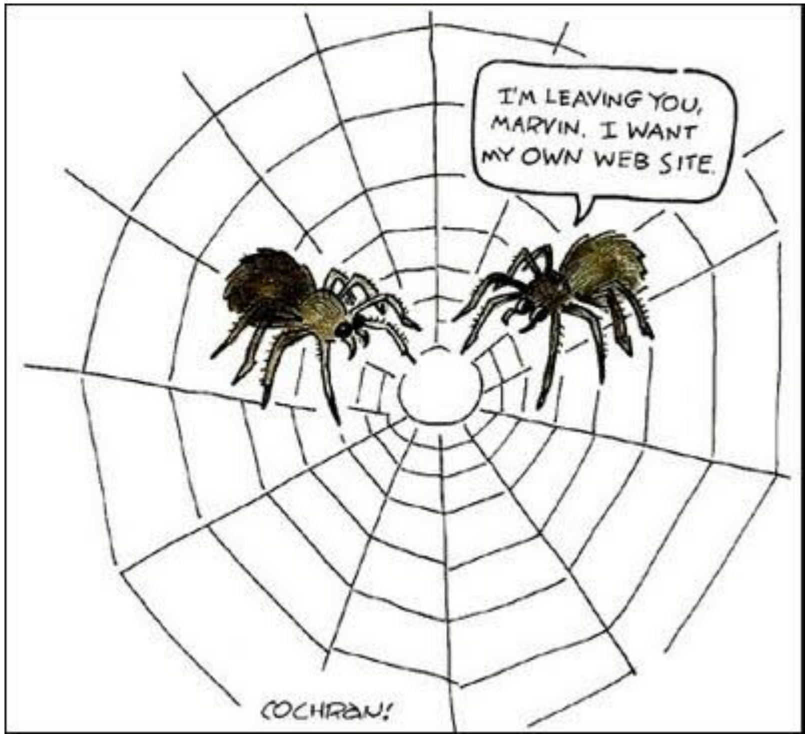
- Tomcat
  - Tomcat 6.0.x + SpringSource modifications (Tomcat 7.0.x features)
  - Servlet 2.5
  - partial support for Servlet 3.0
  - programmatic OSGi service lookup

- Snaps

Web problems

- Tomcat
  - Modified / Forked
    - New version of Tomcat → additional changes have to be applied
    - Issues only in the modified Tomcat
  - ~~Servlet 3.0~~
  - ~~Read web app context configuration from archive~~

- Gemini WEB
  - ~~Consumable~~
    - ~~Eclipse~~
    - ~~tools~~
  - OSGi services injection in servlets
  - Snaps
  - Custom global web.xml outside org.eclipse.gemini.web.tomcat bundle



Web

proposal

- Move to Tomcat 7.0.x
  - ~~Fork~~
  - Servlet 3.0
  - Read web app context configuration from archive

- P2 update site for Gemini WEB
  - Standard way to consume & develop
  - Target platform support

- **Open** questions
  - OSGi services injection in servlets
    - not in specification
    - provides key functionality
    - prototype can be implemented
  - Custom global web.xml outside org.eclipse.gemini.web.tomcat bundle

Agenda

1. File system
2. Dependencies cleanup
3. Nested frameworks
4. Provisioning
5. Shell
6. Web
7. **Security**
8. Roadmap

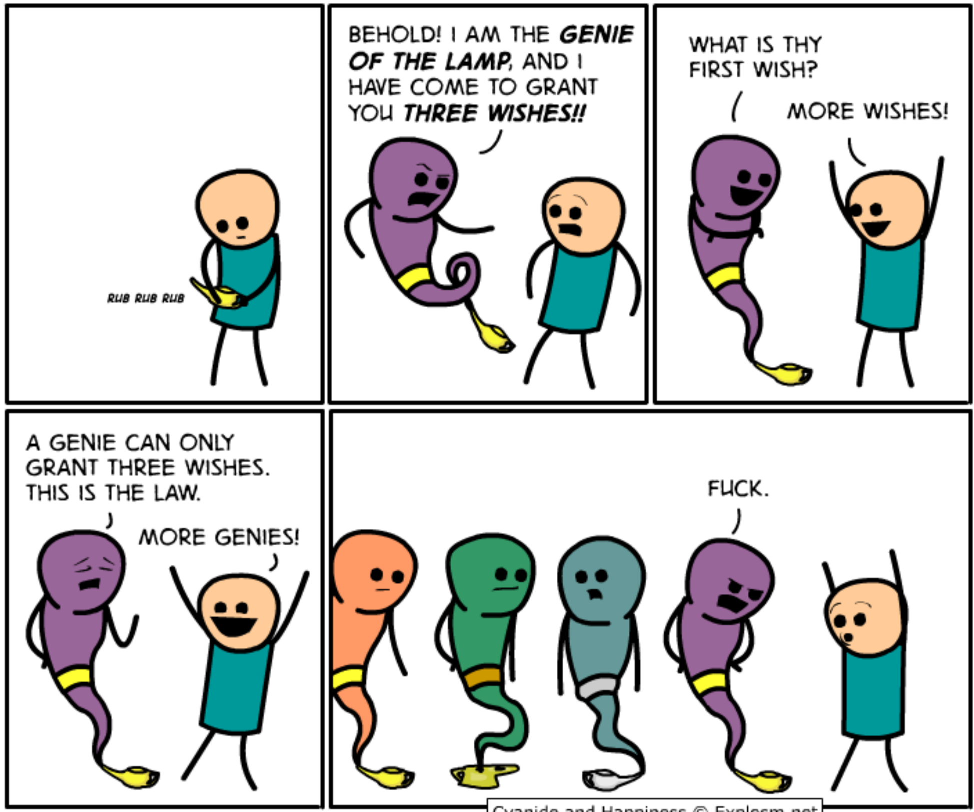
Security

goals

- **Target:** Isolation between apps and infrastructure:
  - applet-like environment
  - Restrict certain API usages (OSGI framework, tomcat management, mbeans, etc.)
  - Restrict OS access (java.io, reflection, java.net)
  - Protect the system from misbehaving apps (OOM, threads)

- Base Sandbox concept - a combination of:
  - **Security manager** (or lightweight variant of it)
  - **Static code** introspection on deployment
  - **Codebase security**

- Unclear whether RFC 138 covers that: needs further prototyping?
- Proposal:
  - go for a PoC?
  - Subproject in Virgo?


RUB RUB RUB
BEHOLD! I AM THE GENIE OF THE LAMP, AND I HAVE COME TO GRANT YOU THREE WISHES!!
WHAT IS THY FIRST WISH?
MORE WISHES!
A GENIE CAN ONLY GRANT THREE WISHES. THIS IS THE LAW.
MORE GENIES!
FUCK.

Cyanide and Happiness © Explosm.net

Agenda

1. File system
2. Dependencies cleanup
3. Nested frameworks
4. Provisioning
5. Shell
6. Web
7. Security
8. **Roadmap**

Roadmap

- Current Virgo plan

Jan 2010

Initial contribution proposal

DM Server

Gemini Web

Oct 2010

Initial graduation release

Slices prototype

Equinox 3.6.1

Tomcat 6.0.29

Gemini Web 1.1.0

Diagnostic improvements

Various bugfixes

Jun 2011

Jetty support

Use Equinox config admin instead of that of Felix

Use Equinox log service instead of that of Felix

Use Equinox event admin instead of that of Felix

Upgrade Spring to 3.0.4 or later

Critical bug fixes

Future releases

Integration server

Batch server

p2 repository in the repository chain

Rebase region support on OSGi RFC138

Upgrade from Spring DM to Gemini Blueprint

Admin console administration of remote Virgo instances

Multiple user regions

Tomcat 7/servlet 3.0

- Proposed plan changes:
  - With regards to increased capacity - switch to monthly milestone releases
  - Aim for the following items in Q1 2011:
    - Tomcat 7 adoption
    - Support for p2 repositories (initial provisioning and update)
    - Deployer refactoring PoCs (façade)
    - Transition Virgo kernel to Declarative Services
    - OSGi Enterprise tools alignment
    - Basic sandbox concept **(?)**
    - Dependency Injection support
  - Aim for the following items in Q2 2011:
    - Final Indigo adoption?
    - Sandboxed user regions **(?)**
    - Further integration of P2 and Virgo deployer
    - Remote Virgo instances monitoring/management
    - Gemini management alignment
    - Session-tolerant graceful shutdown

Roadmap

- Proposed plan changes:
  - Aim for the following items in Q3 2011:
    - Serviceability dump extensions
    - Snaps alignment with Java EE 7 discussions **(?)**
    - Java 7 adoption (or at least compatibility check)
  - Aim for the following items in Q4 2011:
    - Finalization of OSGi Enterprise spec.
  - Aim for the following items in Q4 2011:
    - Java EE 6 web profile **(?)**
- Open questions:
  - Aim to have a Java EE 6 Web release of Virgo (2012)?
  - Do we aim at tight integration with particular Eclipse release (3.7/3.8)?
  - Do we plan to position the Virgo kernel as general-level app server infrastructure?