

OSGi Application Provisioning

Deep Dive

Subsystems, Repository, Contracts and more...

David Bosschaert

About me

David Bosschaert (davidb@apache.org)

- ⦿ Works at Adobe R&D Basel
- ⦿ Co-chair OSGi EEG
- ⦿ Apache committer
- ⦿ Open-source and cloud enthusiast

Topics

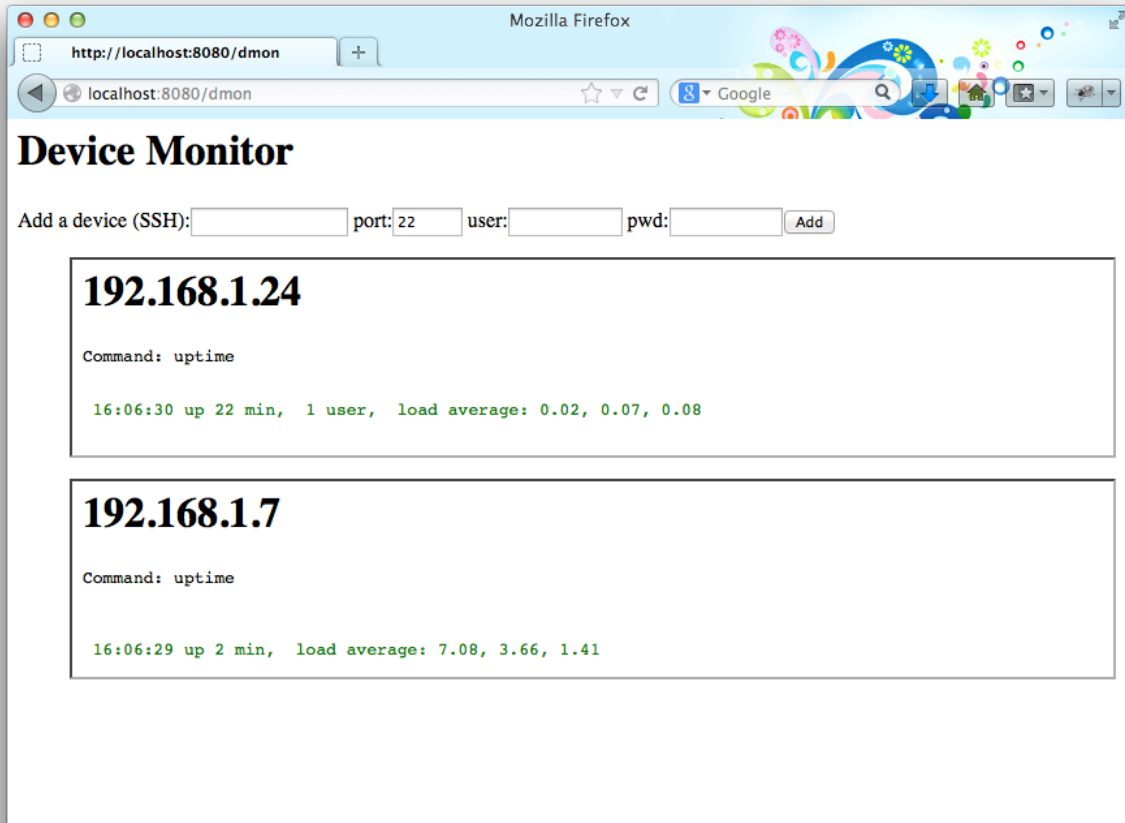
- ⦿ A look at the latest Declarative Services
- ⦿ Using Subsystems to package and deploy
- ⦿ OSGi Repository to resolve dependencies
- ⦿ Portable Java Contracts

... demo throughout ...

Running Example

Device Monitor

... a little webapp to monitor all your gadgets ...



Declarative Services

Being updated for Enterprise R6

- Improved Configuration Admin integration
- Introspective API
- Prototype Service Factory
- ... other smaller improvements ...

DS Configured Component

```
import javax.servlet.Servlet;  
import org.osgi.service.component.annotations.*;  
import org.osgi.service.http.HttpService;  
  
@Component  
public class DeviceMonitor {  
    private HttpService httpService;  
  
    @Reference  
    public void setHttpService(HttpService svc) {  
        httpService = svc;  
    }  
  
    @Activate  
    public void activate(MonitorConfig cfg) {  
        String rootCtx = cfg.ctxPrefix();  
        if (!rootCtx.endsWith("/"))  
            rootCtx = rootCtx + "/";  
  
        registerServlet(rootCtx + "dmon", new MonitorServlet());  
        registerServlet(rootCtx + "device", new DeviceServlet());  
    }  
  
    private void registerServlet(String ctx, Servlet servlet) {  
        httpService.registerServlet(ctx, servlet, null, null);  
    }  
}
```

Configuration using Annotations

```
public @interface MonitorConfig {  
    String ctxPrefix() default "/";  
  
    boolean autoRefresh() default false;  
    int interval() default 30;  
}
```

annotation used as an ordinary interface,
with added defaults

... weird but works great ...

Example Bundle Manifest

```
Bundle-ManifestVersion: 2
Bundle-SymbolicName: devicemon-ds
Bundle-Version: 0.0.1
Service-Component: OSGI-INF/component.xml
Import-Package: javax.servlet;version="[2.5,3)",
    org.coderthoughts.devicemon.ssh;version="[1.0,2)",
    org.osgi.service.http;version="[1.2,2)"
Require-Capability:
    osgi.extender;filter:="(&(osgi.extender=osgi.component)
        (version>=1.2.0)(!(version>=2.0.0)))",
    osgi.whiteboard;filter:="(osgi.whiteboard=osgi.http)"
```

Note that the `javax.servlet` import should really use contracts!

Finished app...

2 Bundles

- `devicemon.jar`
depends on DS and HTTP Service
- `devicemon-ssh.jar`
depends on Apache Mina SSHD

Deploy it...

Now I want to easily deploy my app



create a Subsystem of it!

Subsystems

OSGi Enterprise spec 134

A subsystem...

- collection of bundles put together
 - feature - everything shared
 - application - isolated, nothing shared out
 - composite - configurable in-between
- a zip file with .esa extension
- can be nested
- can depend on bundles or other subsystems

Subsystems can use OSGi Repositories to resolve dependencies

devicemon-ds.esa

Just a zip file...

```
$ tar tvf devicemon-ds.esa
-rwxrwxrwx    99  8 Jan 15:40 OSGI-INF/SUBSYSTEM.MF
-rwxrwxrwx 18184  8 Jan 16:01 devicemon-ds.jar
-rwxrwxrwx  5968  8 Jan 16:02 devicemon-ssh.jar
```

Subsystem Manifest

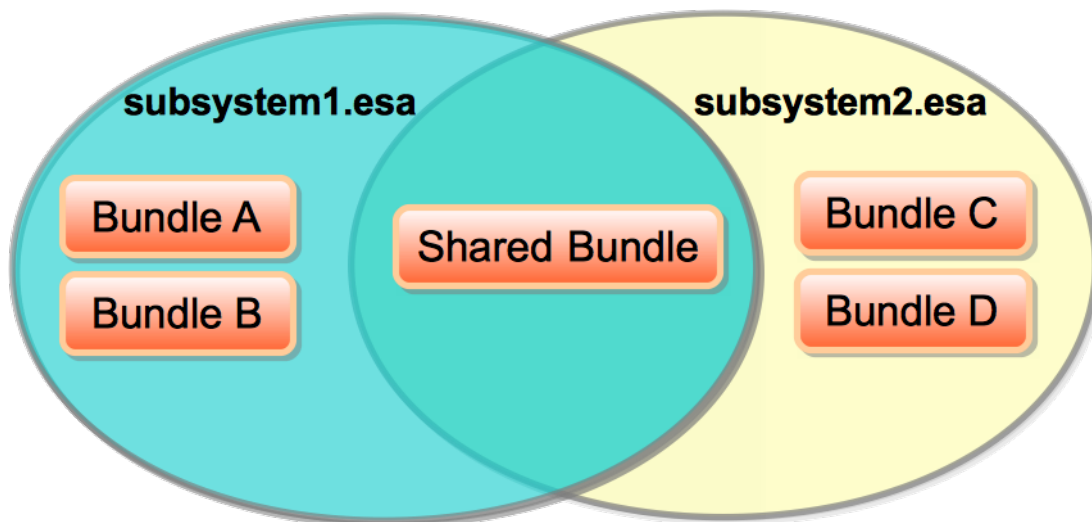
```
$ cat OSGI-INF/SUBSYSTEM.MF
Subsystem-SymbolicName: devicemon-ds
Subsystem-Version: 0.0.1
Subsystem-Type: osgi.subsystem.feature
```

note, I didn't put my dependencies in the .esa file

Feature subsystems

devicemon-ds.esa: a *Feature Subsystem*

- All the bundles inside work just as shared bundles in OSGi
- Subsystem installed/started/stopped as 1 unit
- Dependencies pulled in either from .esa or from repository
 - as needed
 - are reference-counted



Apache Felix + Apache Aries Subsystems

```
7. java
Welcome to Apache Felix Gogo

g! lb
START LEVEL 1
ID|State      |Level|Name
0|Active      | 0|System Bundle (4.3.0.SNAPSHOT)
1|Active      | 1|Apache Aries Subsystem API (1.0.0)
2|Active      | 1|Apache Aries Subsystem Core (1.0.1.SNAPSHOT)
3|Active      | 1|Apache Aries Util (1.1.0)
4|Active      | 1|Apache Felix Configuration Admin Service (1.8.0)
5|Active      | 1|Apache Felix Coordinator Service (0.0.1.SNAPSHOT)
6|Active      | 1|Apache Felix Gogo Command (0.12.0)
7|Active      | 1|Apache Felix Gogo Runtime (0.10.0)
8|Active      | 1|Apache Felix Gogo Shell (0.10.0)
9|Active      | 1|Apache Felix Resolver (1.0.0)
10|Active     | 1|Region Digraph (1.1.0.v20120522-1841)
11|Active     | 1|slf4j-api (1.7.5)
12|Resolved   | 1|slf4j-simple (1.7.5)
13|Active     | 1|Subsystem Gogo Command (1.0.1)
14|Active     | 1|org.osgi.service.subsystem.region.context.0 (1.0.0)

g! █
```

Add HTTP subsystem

```
7. java
ID|State      |Level|Name
0|Active      |    |0|System Bundle (4.3.0.SNAPSHOT)
1|Active      |    |1|Apache Aries Subsystem API (1.0.0)
2|Active      |    |1|Apache Aries Subsystem Core (1.0.1.SNAPSHOT)
3|Active      |    |1|Apache Aries Util (1.1.0)
4|Active      |    |1|Apache Felix Configuration Admin Service (1.8.0)
5|Active      |    |1|Apache Felix Coordinator Service (0.0.1.SNAPSHOT)
6|Active      |    |1|Apache Felix Gogo Command (0.12.0)
7|Active      |    |1|Apache Felix Gogo Runtime (0.10.0)
8|Active      |    |1|Apache Felix Gogo Shell (0.10.0)
9|Active      |    |1|Apache Felix Resolver (1.0.0)
10|Active     |    |1|Region Digraph (1.1.0.v20120522-1841)
11|Active     |    |1|slf4j-api (1.7.5)
12|Resolved   |    |1|slf4j-simple (1.7.5)
13|Active     |    |1|Subsystem Gogo Command (1.0.1)
14|Active     |    |1|org.osgi.service.subsystem.region.context.0 (1.0.0)
g! subsystem:install file:/Users/David/clones/devicemon/devicemon-repo/local-repo/http.esa
Installing subsystem: file:/Users/David/clones/devicemon/devicemon-repo/local-repo/http.esa
Subsystem successfully installed: http-service-subsystem; id: 1
g! subsystem:start 1
```

3 bundles in 1 operation

```
7. java
1b
START LEVEL 1
ID|State      |Level|Name
0|Active      |0|System Bundle (4.3.0.SNAPSHOT)
1|Active      |1|Apache Aries Subsystem API (1.0.0)
2|Active      |1|Apache Aries Subsystem Core (1.0.1.SNAPSHOT)
3|Active      |1|Apache Aries Util (1.1.0)
4|Active      |1|Apache Felix Configuration Admin Service (1.8.0)
5|Active      |1|Apache Felix Coordinator Service (0.0.1.SNAPSHOT)
6|Active      |1|Apache Felix Gogo Command (0.12.0)
7|Active      |1|Apache Felix Gogo Runtime (0.10.0)
8|Active      |1|Apache Felix Gogo Shell (0.10.0)
9|Active      |1|Apache Felix Resolver (1.0.0)
10|Active     |1|Region Digraph (1.1.0.v20120522-1841)
11|Active     |1|slf4j-api (1.7.5)
12|Resolved   |1|slf4j-simple (1.7.5)
13|Active     |1|Subsystem Gogo Command (1.0.1)
14|Active     |1|org.osgi.service.subsystem.region.context.0 (1.0.0)
15|Active     |1|Apache Felix Metatype Service (1.0.8)
16|Active     |1|Apache Felix EventAdmin (1.3.2)
17|Active     |1|Apache Felix Http Jetty (2.2.2)
g! █
```


Our subsystem doesn't install!

```
7. java
17|Active    |    1|Apache Felix Http Jetty (2.2.2)
18|Active    |    1|Apache Felix Declarative Services (1.8.2)
g! subsystem:install file:/Users/David/clones/devicemon/devicemon-ds-subsystem/g
enerated/devicemon-ds.esa
Installing subsystem: file:/Users/David/clones/devicemon/devicemon-ds-subsystem/
generated/devicemon-ds.esa
gogo: SubsystemException: org.osgi.service.resolver.ResolutionException: Unable
to resolve /var/folders/fk/0n8fx3k91ls8lh981zzmm2lh0000gn/T/inputStreamExtract81
44331333039951611.zip/devicemon-ds.jar: missing requirement org.apache.aries.sub
system.core.archive.ImportPackageRequirement: namespace=osgi.wiring.package, att
ributes={}, directives={filter=(&(osgi.wiring.package=org.coderthoughts.devicemo
n.ssh)(&(version>=1.0.0)(!(version>=2.0.0))))}, resource=/var/folders/fk/0n8fx3k
91ls8lh981zzmm2lh0000gn/T/inputStreamExtract8144331333039951611.zip/devicemon-ds
.jar [caused by: Unable to resolve /var/folders/fk/0n8fx3k91ls8lh981zzmm2lh0000g
n/T/inputStreamExtract8144331333039951611.zip/devicemon-ssh.jar: missing require
ment org.apache.aries.subsystem.core.archive.ImportPackageRequirement: namespace=
osgi.wiring.package, attributes={}, directives={filter=(&(osgi.wiring.package=org
.apache.sshd)(&(version>=0.9.0)(!(version>=1.0.0))))}, resource=/var/folders/f
k/0n8fx3k91ls8lh981zzmm2lh0000gn/T/inputStreamExtract8144331333039951611.zip/dev
icemon-ssh.jar]
g! subsystem:list
0      ACTIVE  org.osgi.service.subsystem.root 1.0.0
1      ACTIVE  http-service-subsystem 1.2.0
g! lb
```

NB: a failed subsystem install doesn't leave any bundles behind...

OSGi Repository

OSGi Enterprise spec 132

- Inspired by Felix OBR
- Simple but powerful
- Actual repo can be remote or local
- Find resources
 - based on their capabilities
 - any resource
 - any capability
- OSGi has defined:
 - standard Bundle capabilities
 - Bundle, Subsystem resource types
 - some more general capabilities

Add resources using standard XML

Example

```
<repository increment='1389281419631' name='MyRepo' xmlns='http://www.osgi.org/xmlns/repository/v1.0.0'>
  <resource>
    <capability namespace='osgi.identity'>
      <attribute name='osgi.identity' value='org.apache.sshd.core' />
      <attribute name='type' value='osgi.bundle' />
      <attribute name='version' type='Version' value='0.9.0' />
    </capability>
    <capability namespace='osgi.content'>
      <attribute name='osgi.content' value='a1c64578808c38a63cd6563e9936f025638aeaf9de70f36765367db81c0af' />
      <attribute name='url' value='local-repo/sshd-core.jar' />
      <attribute name='size' type='Long' value='464733' />
      <attribute name='mime' value='application/vnd.osgi.bundle' />
    </capability>
    <capability namespace='osgi.wiring.package'>
      <attribute name='osgi.wiring.package' value='org.apache.sshd' />
      <attribute name='version' type='Version' value='0.5.0' />
      <attribute name='bundle-symbolic-name' value='org.apache.sshd.core' />
      <attribute name='bundle-version' type='Version' value='0.9.0' />
      <directive name='uses' value='org.apache.sshd.client' />
    </capability>
    <!-- More capabilities -->
    <requirement namespace='osgi.wiring.package'>
      <directive name='filter'
        value='(&(osgi.wiring.package=org.slf4j)(version>=1.6.0)(!(version>=2.0.0)))' />
    </requirement>
    <!-- More requirements -->
  </resource>
  <!-- More resources -->
</repository>
```

Repository XML

- Format defined by OSGi Repository Spec
- Standard way to feed a repository with information
- Standard way for repositories to exchange data
- Generate it with the bindex/RepoIndex tool:

<https://github.com/osgi/bindex>
(<https://github.com/osgi/bindex>)

Repository Service

Obtain resources from the repository

Find a bundle...

```
Repository repo = ... // from Service Registry ...  
  
RequirementBuilder rb = new RequirementBuilder("osgi.wiring.package");  
rb.addDirective("filter",  
    "&(osgi.wiring.package=org.apache.ssh)(version=0.5.0)");  
Requirement req = rb.build();  
Collection<Resource> bundleResources = repo.findProviders(req);
```

... or find some photo's from the North Pole ...

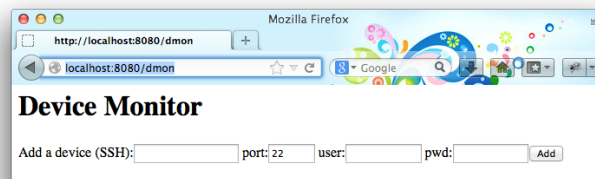
```
RequirementBuilder rb = new RequirementBuilder("com.acme.photo");  
rb.addDirective("filter", "(latitude>=66.5622)");  
Requirement req = rb.build();  
Collection<Resource> photoResources = repo.findProviders(req);
```

Add and prime a Repository

```
7. java
5|Active | 1|Apache Felix Coordinator Service (0.0.1.SNAPSHOT)
6|Active | 1|Apache Felix Gogo Command (0.12.0)
7|Active | 1|Apache Felix Gogo Runtime (0.10.0)
8|Active | 1|Apache Felix Gogo Shell (0.10.0)
9|Active | 1|Apache Felix Resolver (1.0.0)
10|Active | 1|Region Digraph (1.1.0.v20120522-1841)
11|Active | 1|slf4j-api (1.7.5)
12|Resolved | 1|slf4j-simple (1.7.5)
13|Active | 1|Subsystem Gogo Command (1.0.1)
14|Active | 1|org.osgi.service.subsystem.region.context.0 (1.0.0)
15|Active | 1|Apache Felix Metatype Service (1.0.8)
16|Active | 1|Apache Felix EventAdmin (1.3.2)
17|Active | 1|Apache Felix Http Jetty (2.2.2)
18|Active | 1|Apache Felix Declarative Services (1.8.2)
g! install file:/Users/David/checkouts/felix_obr/bundlerepository/target/org.apache.felix.bundlerepository-1.7.0-SNAPSHOT.jar
Bundle ID: 19
g! start 19
g! obr:repos add file:/Users/David/clones/devicemon/devicemon-repo/index.xml
g! obr:list
cdi-subsystem (0.5.0)
http-subsystem (1.2.0)
org.apache.sshd.core (0.9.1, ...)
org.apache.felix.scr (1.8.2)
```

Our subsystem works!

```
7. java
4|Active | 1|Apache Felix Configuration Admin Service (1.8.0)
5|Active | 1|Apache Felix Coordinator Service (0.0.1.SNAPSHOT)
6|Active | 1|Apache Felix Gogo Command (0.12.0)
7|Active | 1|Apache Felix Gogo Runtime (0.10.0)
8|Active | 1|Apache Felix Gogo Shell (0.10.0)
9|Active | 1|Apache Felix Resolver (1.0.0)
10|Active | 1|Region Digraph (1.1.0.v20120522-1841)
11|Active | 1|slf4j-api (1.7.5)
12|Resolved | 1|slf4j-simple (1.7.5)
13|Active | 1|Subsystem Gogo Command (1.0.1)
14|Active | 1|org.osgi.service.subsystem.region.context.0 (1.0.0)
15|Active | 1|Apache Felix Metatype Service (1.0.8)
16|Active | 1|Apache Felix EventAdmin (1.3.2)
17|Active | 1|Apache Felix Http Jetty (2.2.2)
18|Active | 1|Apache Felix Declarative Services (1.8.2)
19|Active | 1|Apache Felix Bundle Repository (1.7.0.SNAPSHOT)
20|Installed | 1|Apache Mina SSHD :: Core (0.9.0)
21|Installed | 1|devicemon-ssh (0.0.1)
22|Installed | 1|devicemon-ds (0.0.1)
g! subsystem:list
0 ACTIVE org.osgi.service.subsystem.root 1.0.0
1 ACTIVE http-service-subsystem 1.2.0
3 INSTALLED devicemon-ds 0.0.1
g! subsystem:start 3
```



Descriptor Subsystems

A subsystem can be just a descriptor

... with all resources obtained from a Repository

```
$ jar tvf generated/devicemon-ds-decl.esa
  0 Thu Jan 09 12:37:32 GMT 2014 OSGI-INF/
175 Wed Jan 08 15:50:28 GMT 2014 OSGI-INF/SUBSYSTEM.MF
```

Subsystem Manifest

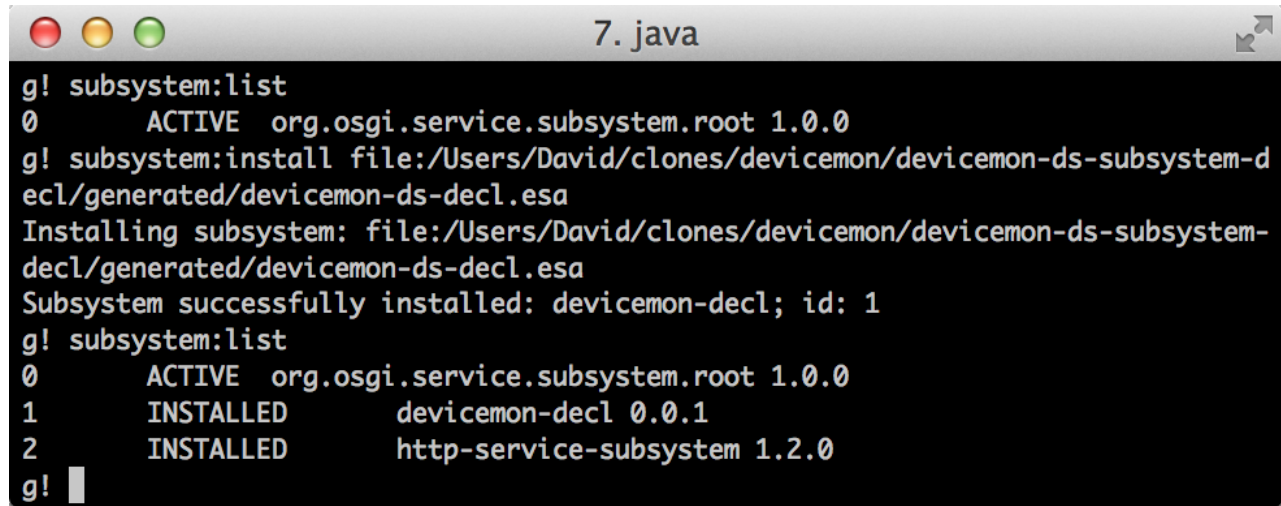
```
$ cat OSGI-INF/SUBSYSTEM.MF
Subsystem-SymbolicName: devicemon-ds-decl
Subsystem-Version: 0.0.1
Subsystem-Type: osgi.subsystem.feature
Subsystem-Content: devicemon-ds;version=0.0.1,
    devicemon-ssh;version=0.0.1
```


Start with only Subsystems + Repository

```
7. java
0|Active      | 0|System Bundle (4.3.0.SNAPSHOT)
1|Active      | 1|Apache Aries Subsystem API (1.0.0)
2|Active      | 1|Apache Aries Subsystem Core (1.0.1.SNAPSHOT)
3|Active      | 1|Apache Aries Util (1.1.0)
4|Active      | 1|Apache Felix Configuration Admin Service (1.8.0)
5|Active      | 1|Apache Felix Coordinator Service (0.0.1.SNAPSHOT)
6|Active      | 1|Apache Felix Gogo Command (0.12.0)
7|Active      | 1|Apache Felix Gogo Runtime (0.10.0)
8|Active      | 1|Apache Felix Gogo Shell (0.10.0)
9|Active      | 1|Apache Felix Resolver (1.0.0)
10|Active     | 1|Region Digraph (1.1.0.v20120522-1841)
11|Active     | 1|slf4j-api (1.7.5)
12|Resolved  | 1|slf4j-simple (1.7.5)
13|Active     | 1|Subsystem Gogo Command (1.0.1)
14|Active     | 1|org.osgi.service.subsystem.region.context.0 (1.0.0)
15|Active     | 1|Apache Felix Bundle Repository (1.7.0.SNAPSHOT)
g! obr:list
devicemon-ds (0.0.1)
devicemon-ssh (0.0.1)
cdi-subsystem (0.5.0)
http-subsystem (1.2.0)
org.apache.sshd.core (0.9.1, ...)
org.apache.felix.scr (1.8.2)
g! █
```

Dependencies at work

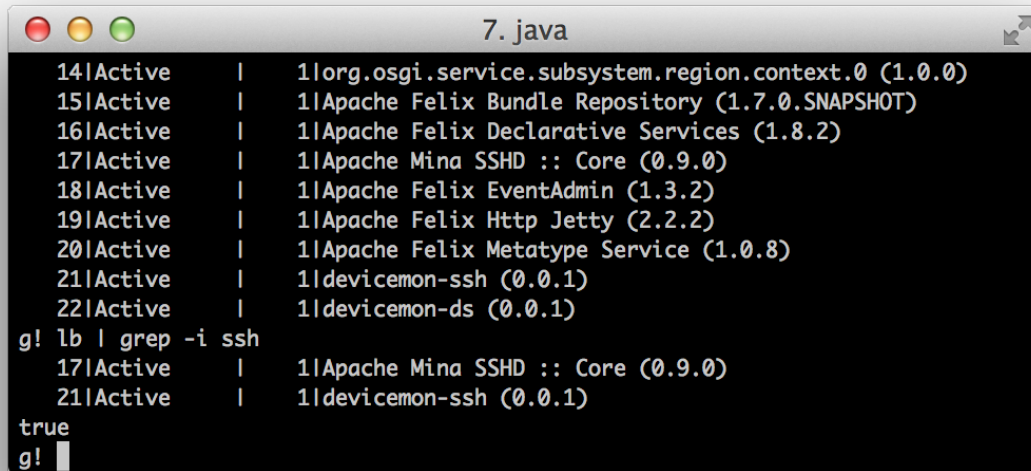
http.esa pulled in automatically



```
7. java
g! subsystem:list
0      ACTIVE  org.osgi.service.subsystem.root 1.0.0
g! subsystem:install file:/Users/David/clones/devicemon/devicemon-ds-subsystem-decl/generated/devicemon-ds-decl.esa
Installing subsystem: file:/Users/David/clones/devicemon/devicemon-ds-subsystem-decl/generated/devicemon-ds-decl.esa
Subsystem successfully installed: devicemon-decl; id: 1
g! subsystem:list
0      ACTIVE  org.osgi.service.subsystem.root 1.0.0
1      INSTALLED  devicemon-decl 0.0.1
2      INSTALLED  http-service-subsystem 1.2.0
g! |
```

Unpredictable dependency

both 0.9.0 and 0.9.1 satisfy requirement



```
7. java
14|Active | 1|org.osgi.service.subsystem.region.context.0 (1.0.0)
15|Active | 1|Apache Felix Bundle Repository (1.7.0.SNAPSHOT)
16|Active | 1|Apache Felix Declarative Services (1.8.2)
17|Active | 1|Apache Mina SSHD :: Core (0.9.0)
18|Active | 1|Apache Felix EventAdmin (1.3.2)
19|Active | 1|Apache Felix Http Jetty (2.2.2)
20|Active | 1|Apache Felix Metatype Service (1.0.8)
21|Active | 1|devicemon-ssh (0.0.1)
22|Active | 1|devicemon-ds (0.0.1)
g! lb | grep -i ssh
17|Active | 1|Apache Mina SSHD :: Core (0.9.0)
21|Active | 1|devicemon-ssh (0.0.1)
true
g! |
```

Subsystem Deployment Descriptor

Once QA-ed, create a DEPLOYMENT.MF to fix dependencies

```
$ jar tvf generated/devicemon-ds-decl-dd.esa
 283 Wed Jan 15 16:39:34 GMT 2014 OSGI-INF/DEPLOYMENT.MF
 175 Wed Jan 08 15:50:28 GMT 2014 OSGI-INF/SUBSYSTEM.MF
```

it can freeze deployments

```
$ cat OSGI-INF/DEPLOYMENT.MF
Subsystem-Version: 0.0.1
Subsystem-Type: osgi.subsystem.feature
Deployed-Content: devicemon-ds;deployed-version=0.0.1,
  devicemon-ssh;deployed-version=0.0.1
Provision-Resource: org.apache.sshd.core;deployed-version=0.9.1,
  org.apache.felix.scr;deployed-version=1.8.2,
  http-subsystem;type=osgi.subsystem.feature;deployed-version=1.2.0
```

note version 0.9.1 for org.apache.sshd.core

Ensures that the runtime deployment is the same as the QA one.

SSHD dependency now as predicted

```
7. java
g! subsystem:install file:/Users/David/clones/devicemon/devicemon-ds-subsystem-decl-dd/generated/devicemon-ds-decl-dd.esa
Installing subsystem: file:/Users/David/clones/devicemon/devicemon-ds-subsystem-decl-dd/generated/devicemon-ds-decl-dd.esa
Subsystem successfully installed: devicemon-ds-decl-dd; id: 1
g! subsystem:list
0      ACTIVE  org.osgi.service.subsystem.root 1.0.0
1      INSTALLED devicemon-ds-decl-dd 0.0.1
2      INSTALLED http-service-subsystem 1.2.0
g! lb | grep -i ssh
17|Installed | 1|Apache Mina SSHD :: Core (0.9.1)
21|Installed | 1|devicemon-ssh (0.0.1)
true
g! █
```

Portable Contracts

Our bundle contains:

```
Import-Package: javax.servlet;version="[2.5,3)",
```

This API is defined by the JCP.

but... spot the problem!

Semantic Versioning

... versioning policy for *exported packages*.

OSGi versions: <major>.<minor>.<micro>.<qualifier>

Updating package versions:

- fix/patch (no change to API):
 - **update micro**
- extend API (affects implementers, not clients):
 - **update minor**
- API breakage:
 - **update major**

Using semantic versioning allows creating components that can work with future patches and other compatible releases of deps

just Import-Package a range like: [5.3, 6)

javax.servlet Problem

```
Import-Package: javax.servlet;version="[2.5,3)",
```

Servlet 3.0 is actually compatible with 2.5

- JCP specs don't follow semantic versioning
- 3.0 is more of a marketing version
- But what version to use for `javax.servlet` in OSGi?
- Different providers made different decisions
 - Some use 2.6 to export `javax.servlet` from Servlet 3 spec
 - Others use 3.0
 - Some even use 0.0.0

This prohibits creating portable bundles using these APIs

Portable Java Contracts

`osgi.contract` capability to the rescue

- client bundle requires capability
 - with single 'marketing' or 'spec' version
- and also imports the package
without version
 - `osgi.contract` provider binds the contract version to package versions

```
Import-Package: javax.servlet, javax.servlet.http  
Require-Capability: osgi.contract;  
    filter:=("&(osgi.contract=JavaServlet)(version=2.5)")
```

Enables creating *portable* bundles
importing non-semantically versioned
packages.

Our bundle manifest should really be

```
Bundle-ManifestVersion: 2
Bundle-SymbolicName: devicemon-ds
Bundle-Version: 0.0.1
Import-Package: javax.servlet,
                   org.coderthoughts.devicemon.ssh;version="[1.0,2)",
                   org.osgi.service.http;version="[1.2,2)"
Require-Capability:
  osgi.extender;filter:="(&(osgi.extender=osgi.component)
    (version>=1.2)(!(version>=2.0)))",
  osgi.whiteboard;filter:="(osgi.whiteboard=osgi.http)
    (version>=1.0)(!(version>=2.0))",
  osgi.contract;filter:="(&(osgi.contract=Servlet)
    (version=2.5))"
```

Providing Portable Contracts

The bundle exporting the package should do this:

```
Export-Package: javax.servlet;version=2.6,  
    javax.servlet.http;version=2.6  
Provide-Capability: osgi.contract;  
    osgi.contract=JavaServlet;version:Version=2.5;  
    uses:="javax.servlet, javax.servlet.http",  
    osgi.contract;osgi.contract=JavaServlet;version:Version=3;  
    uses:="javax.servlet, javax.servlet.http"
```

as it only knows how the mapping is done.

Each marketing version is mapped to a package version.

*For more details, see [OSGi RFC 180](https://github.com/osgi/design/raw/master/rfcs/rfc0180/rfc-0180-portable-java-contracts.pdf)
(<https://github.com/osgi/design/raw/master/rfcs/rfc0180/rfc-0180-portable-java-contracts.pdf>).*

For all packages other than
JCP-ones

Use Semantic Versioning!

bndtools can **really** help with version
maintenance!

<http://www.bndtools.org>

(<http://www.bndtools.org>)

Where can I get it?

Core R6 spec released this week:

<http://www.osgi.org/Specifications/HomePage>

(<http://www.osgi.org/Specifications/HomePage>)

Enterprise R6 draft released this week:

<http://www.osgi.org/Specifications/Drafts>

(<http://www.osgi.org/Specifications/Drafts>)

RFCs 189, 190, 208 included in zip

Links

- OSGi specs:

<http://www.osgi.org/Specifications>

[\(http://www.osgi.org/Specifications\)](http://www.osgi.org/Specifications)

- Apache Felix Project:

<http://felix.apache.org> (<http://felix.apache.org>)

- OSGi RFC 180

<https://github.com/osgi/design/raw/master>

[/rfcs/rfc0180/rfc-0180-portable-java-contracts.pdf](https://github.com/osgi/design/raw/master/rfcs/rfc0180/rfc-0180-portable-java-contracts.pdf)

[\(https://github.com/osgi/design/raw/master](https://github.com/osgi/design/raw/master)

[/rfcs/rfc0180/rfc-0180-portable-java-contracts.pdf\)](https://github.com/osgi/design/raw/master/rfcs/rfc0180/rfc-0180-portable-java-contracts.pdf)

- devicemon project:

<https://github.com/bosschaert/devicemon>

[\(https://github.com/bosschaert/devicemon\)](https://github.com/bosschaert/devicemon)

- Subsystem Gogo command

<https://github.com/bosschaert/coderthoughts>

/tree/master/subsystem-gogo-command

(https://github.com/bosschaert/coderthoughts

/tree/master/subsystem-gogo-command)

Questions?