

Enterprise OSGi and JBoss AS7

The logo for the JBoss Community, featuring the text "JBoss" in a large, bold, blue font above the word "Community" in a smaller, bold, blue font. The text is set against a white rectangular background with a subtle glow effect.

**JBoss
Community**

David Bosschaert

david@redhat.com

Thomas Diesler

thomas.diesler@jboss.com

Agenda

- OSGi Core
- Enterprise OSGi
- OSGi in AS7
- Demos



OSGi Core standards

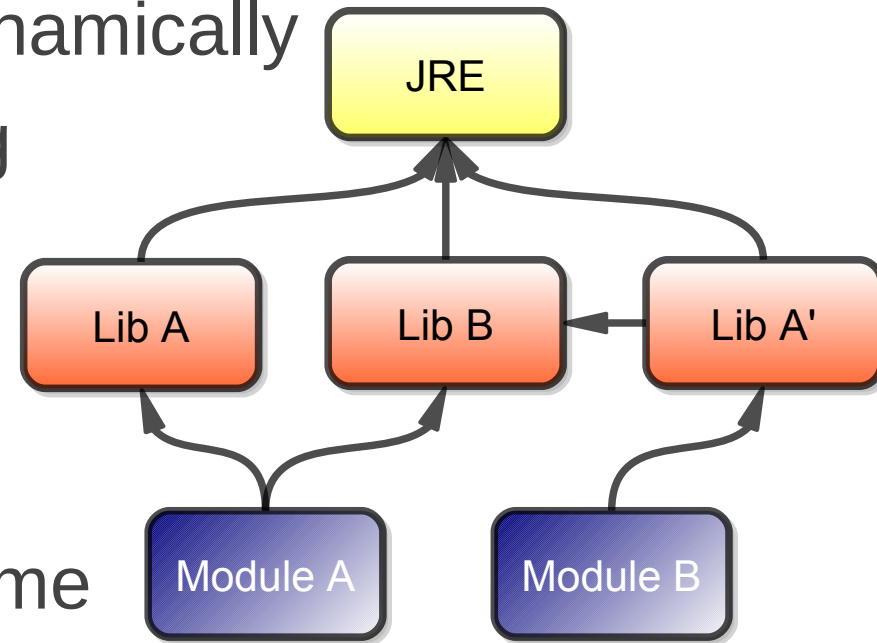
Topics

- OSGi Modularity
- OSGi Services
- What about jboss-modules and jboss-msc?
- JavaSE 8 Modules



OSGi Core: Modularity

- OSGi modules are called 'Bundles'
- Computed classloader delegation graph
- Bundles can be added, removed & updated dynamically
- Side-by-side versioning
- Private truly private
- No more Jar Hell
- OSGi Bundle can be a plain Jar at the same time



OSGi Core: Services



- “SOA inside the JVM”
- Services looked up by type and/or custom filter
 - “I want a service that implements org.acme.Payment where location=US”
 - One or many
- Dynamic! Services can be updated without taking down the consumers
 - OSGi Service Consumers react to dynamism



OSGi bundles vs. JBoss modules

- OSGi is a standard
- Uses a complex resolver
- Wiring based on rich metadata. Package version just one criteria
- Attachable Fragments
- Sensitive to exec env
- Many compendium specifications and other 3rd party bundles
- Modules not portable
- Uses explicit wiring
- Wiring based on module name. Multiple “slots” are supported
- Exist in the context of AS7
- Very fast resolution



OSGi services vs. MSC services

- Keyed by Interface they implement
- Properties-based filtering
- Lifecycle tied to owner bundle or explicit API
- Service dependency and injection via DS, Blueprint
- ServiceListener supported
- Instance per registered service supported
- ServiceFactory supported
- Keyed by unique service name
- No filtering, name only
- Lifecycle tied to owner module or explicit API
- Service dependency and injection by MSC API
- ServiceListener supported
- Instance per service name



Making jboss-modules & jboss-msc OSGi Compliant

...is exactly what the JBoss OSGi subproject in AS7 does!

More details later...



OSGi[™]
Alliance



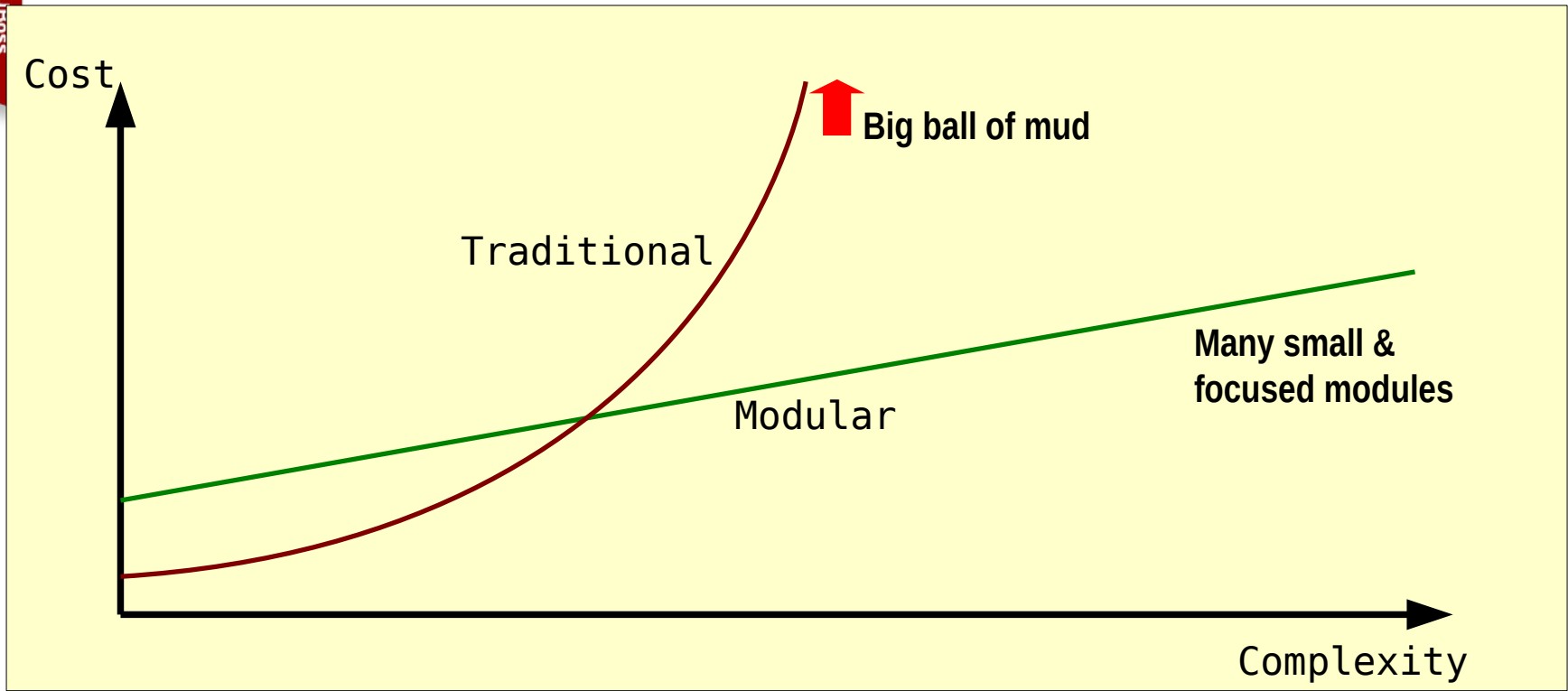
What about JavaSE 8 Modules?



- We'll have to wait and see...
- But unless JavaSE 8 adopts OSGi in full, OSGi will have additional features
 - JavaSE 8 will most likely be less dynamic
- The good news:
 - Migration to OSGi from JavaSE 8 easier
 - Biggest hurdle is project modularization itself
 - This hurdle now moves to JavaSE 8
- JavaSE 8 scheduled for 2013



Cost vs Complexity



Modularity highly improves maintainability*



OSGi Demo



- PDE tooling
- Services
- Show provider code
- Show client code
- ASCII pic with bug
- ASCII pic working
- No need to restart client when updating service

Enterprise OSGi standards

Topics

- Today
- In progress



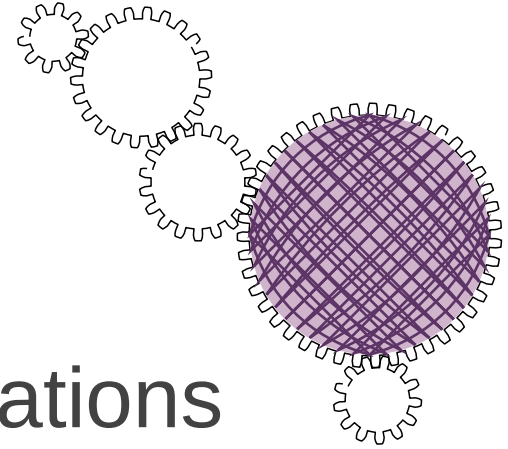


Enterprise OSGi – today

- Mapping of a number of EE specs
 - Web (specifically .WAR files and servlets)
 - JTA
 - JNDI
 - JDBC
 - JMX
 - JPA
- Remote Services
- Blueprint standard (inspired by Spring)
- Configuration Admin Service
- Event Admin Service
- Declarative Services



Enterprise OSGi – in progress



- RFC 152 Subsystems / Applications
- RFC 112 OBR: OSGi Bundle Repository
- RFC 146 JCA (driver: Jesper Pedersen)
- RFC 167 `java.util.ServiceLoader` and `META-INF/services` support (driver: DB)
- RFC 169 JMX enh. (driver: Oracle + DB)
- RFP 133 Cloud Computing



OSGi in JBoss AS7



Topics

- OSGi Core Support
- AS7 XServices
- Enterprise OSGi in AS7



OSGi Core in AS7

- AS7 is a fully compliant OSGi 4.2 Core Framework
 - TCK tests pass 100%
- Just deploy your OSGi bundles
 - Through the standard AS7 deployment channels



OSGi Core in AS7: under the hood



- Implemented from scratch on top of jboss-modules and jboss-msc
- For the OSGi resolver we use the one from Apache Felix
 - Resolver computes the bundle classloader wirings
 - Based on OSGi Metadata
 - e.g. Import-Package / Export-Package



OSGi in AS7 XServices



- Use components from EE as OSGi services
- Use OSGi services in EE components
- JBoss modules/msc as integration layer
- Every bundle is a module. Not every module is also bundle
 - Modules can be given OSGi/Resolver metadata
- Every OSGi Service is an MSC service.
 - MSC services can be aliased to become OSGi services



Enterprise OSGi in AS7



- AS7 OOTB:
 - OSGi Core 4.2
 - Enterprise 4.2 APIs
 - Config Admin
 - JNDI
 - JTA
 - JMX
 - With JBoss OSGi 'installer'
 - WebApps
 - HTTP Service
 - Blueprint
 - Web Console



Next steps

- Integrate more JBoss EE products using Enterprise OSGi:
 - Web Apps using JBoss web
 - CDI integration
 - Possibly JPA/Hibernate support
- OSGi Core 4.3 specification
- OSGi Enterprise Application Models

Demo

- Show that the client can switch to EJB



Links



- AS7 OSGi home page
<http://www.jboss.org/jbossas/osgi>
- OSGi Specifications
<http://www.osgi.org/Download/Release4V42>
- JBoss OSGi Diary
<http://jbossosgi.blogspot.com>



Questions?



Acknowledgements

- Clip art from openclipart.org
 - HectorGomez/netallow/Chrisdesign/tomas_arad/Martouf/Peileppe/kattekrab/eeysja/skotan/AJ/carlitos/georges-dahdouh/augustoschwartz/Gerald_G/tom/lor doftheloch/Viscious-Speed/Andy
- Cost vs Complexity graph: Hal Hildebrand (2009): <http://bit.ly/i42uBo>



**JOIN THE CONVERSATIONS AND STAY
IN TOUCH WITH JBOSS YEAR ROUND!**

LIKE US ON FACEBOOK

facebook.com/jboss

FOLLOW US ON TWITTER

@JBossDeveloper

@JBossOperations

@JBossNews

READ THE BLOG

redhat.com/about/news/blog/
jboss.org/feeds/

SUMMIT

**JBoss
WORLD**

PRESENTED BY RED HAT

