Themes

• How did we get here
• Java: Fear of Forking
• OSGi Trends, Directions and Opportunities
How Did We Get Here?

- Object Orientation
- Component-based development
- Smalltalk, of course
- The eternal question of reuse – what’s the ideal granularity for a bundle/service/component?
- Java – fear of forking
Java: Fear of The Fork

- Massively successful
- One Big Hairball, or three?

- Yet modularity was an afterthought
Java Centres Of Gravity

- the ASF
- the Eclipse Foundation
- Google
- IBM
- Oracle
- OSGi
- VMware Springsource
Unforkable Software?

- Oracle – OSGi in WebLogic, Glassfish
- Java Community Process?
- LibreOffice, ForgeRock, Illumos
Eclipse

- Governance ++
- Shared implementations
- Shared specifications
- Licensing matters
- Consumability
- Extensibility
Google <3 Java?

- Android
  - Fragmentation – or diverse ecology?
  - Java on linux for mobile, promise realised
  - Real innovation
  - Cluttered API?
  - Dalvik and the case against

- GWT

- Instantiations – Google as an IDE provider

- Summer of Code
IBM: A Disturbance In The Force

- WebSphere refocusing on technology, developers, leading the OSGI charge

- See Expeditor
“For the first time in 10 years, I have no desire to go to JavaOne, or whatever's left of it.”

Rod Johnson, CEO SpringSource

- Spring made JEE less painful
- SpringDM for OSGi, dm Server
- Groovy
OSGi in middleware

- Apache Aries, Sling
- Day Software
- Eclipse BIRT
- Progress Software/ServiceMix
- Oracle Glassfish
- WS02 Carbon
Who are you calling a Lightweight?

- WebSphere, Jboss, Glassfish, Tomcat... Jetty
- Lightweight architectures offer raw performance
Rise of the (Virtual) Machines

- language diversity
- Performance – VM performance improvements
The Rise and Rise of Social Coding

- Agile
- People write code, machines run code
- Github
  - Linus does it again. Open Source is Social Networking.
- Celebrate the fork, but ensure resolution
- Developers get the gist
- Reuse doesn’t always mean a singleton
Ubuntu as Master Packager

- Investing in User Experience
- Easy as apt-get
- Making choices for the user
What About The REST of OSGi?

- SOA became restful
- Web refactoring
- What would OSGi with a web refactoring look like?
- Or – how can OSGi appeal to web-savvy developers?
- Or – OSGi for humans
The Stackless Stack

- Nothing is defined until it is

- Check out the Paremus Nimble shell/framework
  - video demo here.
The Gandhi Model Of Tech Adoption

“First they ignore you, then they ridicule you, then they fight you, then you win”
“OSGi is a great spec for vendors, but a terrible spec for developers. It inflicts a lot of pain.”

Ross Mason, CTO MuleSoft
Brains
“We’re modelling hot deployment on OSGi principles. If someone comes up with a way to make OSGi consumable by the masses we'll jump right on it.”

Ross Mason, CTO MuleSoft
Atlassian – another master packager

• Atlassian plugin model - OSGi without the OSGi

• “OSGi has let us write big features in a portable, modular way across products, and has dramatically sped up the development process via its hot deployment capability. Along the way, we've had some issues, particularly with libraries that don't play well in modular environment, but they certainly haven't outweighed the benefit IMO; OSGi certainly does rock.” – Don Brown, CTO
Why Should Developers Care

- Choose your own container
- Find Code To Start Your Projects
- Constraints can make life easier - Flow
- Dependencies - OK When You Know What They Are
- Runtimes Aren’t Locked into Frameworks
Why Should Managers Care

- Running full stacks for all app dev is expensive
- Fit for purpose runtimes
- Define policies, drive adherence through tools
- Performance
- A standard to rely on
- For embedded- minimum runtime to declare victory
- OSGi as a forcing factor to encourage modularity.
Futures

- OSGi pervasive
  - Automotive
  - Smart grids and home gateways
  - Phones
  - Automation (sensors and actuators)

- OSGi for desktop integration – Lotus + Cisco
  - Come on IBM, where’s the Expeditor developer play?
OSGi and the Cloud

• Private clouds will be all about OSGi
• Platform as a Service – who will deliver?
NoSQL

- Apache SOLR
- Cassandra
- Hadoop
- Riak
- Tokyo Cabinet
The Really Big Challenge

- Retooling Java legacy – a toolchain for modularity
- There is no silverbullet
- Migration services and “retrofitting”
Its all about the earbuds

• The best packager wins
• Users like constraints
• Developers like constraints
• Applications like constraints
• Design matters
Thanks!
• Podium: http://www.flickr.com/photos/spine/2408993662/
• Standing Over The Clouds http://www.flickr.com/photos/donabelandewen/470780785/
• 1, 2, 3 Jump!: http://www.flickr.com/photos/cristeenq/
• Netbook: http://www.flickr.com/photos/ndevil/3349323242/
• Giant hand: http://www.flickr.com/photos/76074333@N00/147857534/
• Pylon: http://www.flickr.com/photos/imuttoo/2423929597/
• Brown field: http://www.flickr.com/photos/brian-m/2812607097/