

Introduction to the Newton Project Distributed OSGi & SCA

Robert Dunne
Paremus

Introduction to the Newton Project Distributed OSGi & SCA

Robert Dunne
Paremus

Newton Overview

- Distributed Application runtime
- OSGi from the ground up
- SCA to describe the distributed apps
- Started early 2005 by Paremus.
 - ♦ <http://www.paremus.com>
- Open sourced mid 2006
 - ♦ <http://newton.codecauldron.org>

Deploy, Maintain, Remove

- Full Application Lifecycle
- Software not people

Deploy, Maintain, Remove

Deploy, Maintain, Remove

- What?
 - ◆ Distributed Applications
 - ◆ Topologies vary
 - ◆ SLAs vary
 - ◆ Building blocks?

Deploy, Maintain, Remove

- What?
 - ♦ Distributed Applications
 - ♦ Topologies vary
 - ♦ SLAs vary
 - ♦ Building blocks?
- Where?
 - ♦ To appropriate compute resource

Deploy, Maintain, Remove

- What?
 - ♦ Distributed Applications
 - ♦ Topologies vary
 - ♦ SLAs vary
 - ♦ Building blocks?
- Where?
 - ♦ To appropriate compute resource
- When?
 - ♦ Interleaved with other deployments

Deploy, **Maintain**, Remove

Deploy, **Maintain**, Remove

- The requirements
 - ◆ Topology
 - ◆ SLA

Deploy, **Maintain**, Remove

- The requirements
 - ◆ Topology
 - ◆ SLA
- As the environment changes
 - ◆ Connectivity
 - ◆ Compute resource

Deploy, **Maintain**, Remove

- The requirements
 - ◆ Topology
 - ◆ SLA
- As the environment changes
 - ◆ Connectivity
 - ◆ Compute resource
- Including its own
 - ◆ Self healing

Deploy, Maintain, Remove

Deploy, Maintain, Remove

- Simple
 - ♦ Just say go away

Deploy, Maintain, Remove

- Simple
 - ♦ Just say go away
- Safe
 - ♦ Dependency management
 - ♦ Garbage Collection

Building blocks - OSGi

Building blocks - OSGi

- Sane classloading

Building blocks - OSGi

- Sane classloading
- Valuable metadata
 - ♦ Code & (some) service dependencies

Building blocks - OSGi

- Sane classloading
- Valuable metadata
 - ◆ Code & (some) service dependencies
- Full lifecycle
 - ◆ Code & Services

Building blocks - OSGi

- Sane classloading
- Valuable metadata
 - ♦ Code & (some) service dependencies
- Full lifecycle
 - ♦ Code & Services
- Popular programming models
 - ♦ Spring, Guice, (and soon) just about everything

Building blocks - OSGi

- Sane classloading
- Valuable metadata
 - ◆ Code & (some) service dependencies
- Full lifecycle
 - ◆ Code & Services
- Popular programming models
 - ◆ Spring, Guice, (and soon) just about everything
- Distributed OSGi
 - ◆ remoting

Building Blocks - SCA

Building Blocks - SCA

- Topological View
 - ◆ Simple high level structure
 - ◆ configuration
 - ◆ matching metadata

Building Blocks - SCA

- Topological View
 - ♦ Simple high level structure
 - ♦ configuration
 - ♦ matching metadata
- Target State
 - ♦ Topology + SLA
 - ♦ SCA system as goal

Target State

Target State

- Newton provisioning system
 - ♦ Tries to maintain target state for all systems
 - ♦ Only addresses the delta

Target State

- Newton provisioning system
 - ◆ Tries to maintain target state for all systems
 - ◆ Only addresses the delta
- Target state - work needed for
 - ◆ New systems
 - ◆ Environment changes
 - ◆ Modified systems
 - ◆ Retired systems

Newton Project

- Website
 - ♦ <http://newton.codecauldron.org>
- Commercial version
 - ♦ Infiniflow
 - ♦ <http://www.paremus.com>