JUDCon

JBoss Users & Developers Conference

2012:India

JBoss Operations Network Management Simplified.

Sumit Bhat
Support Relationship Manager
Red Hat

Why Manageability Matters?

Managing middleware is complex

- Large & diverse infrastructure - Multiple servers, multiple applications, multiple versions, multiple configurations, mixed source, mixed vendor environment
- Different environments / different needs - Dev / Test / Production
- Lack of tools / too many tools

- Customers that leverage RHQ/JBoss ON are more productive
  - Automation of common administrative tasks
  - Operational control of all environments (Dev, Test, Prod)
  - Experience fewer operational issues throughout the application lifecycle

Key Features

RHQ/JBoss Operations Network helps customers Deploy, Manage & Monitor their JBoss servers and applications to reduce costs and ensure performance and reliability

Deploy

Provisioning

Application Deployment

Manage

Discovery & Inventory

Configuration Management

Content Mgt.

Monitor

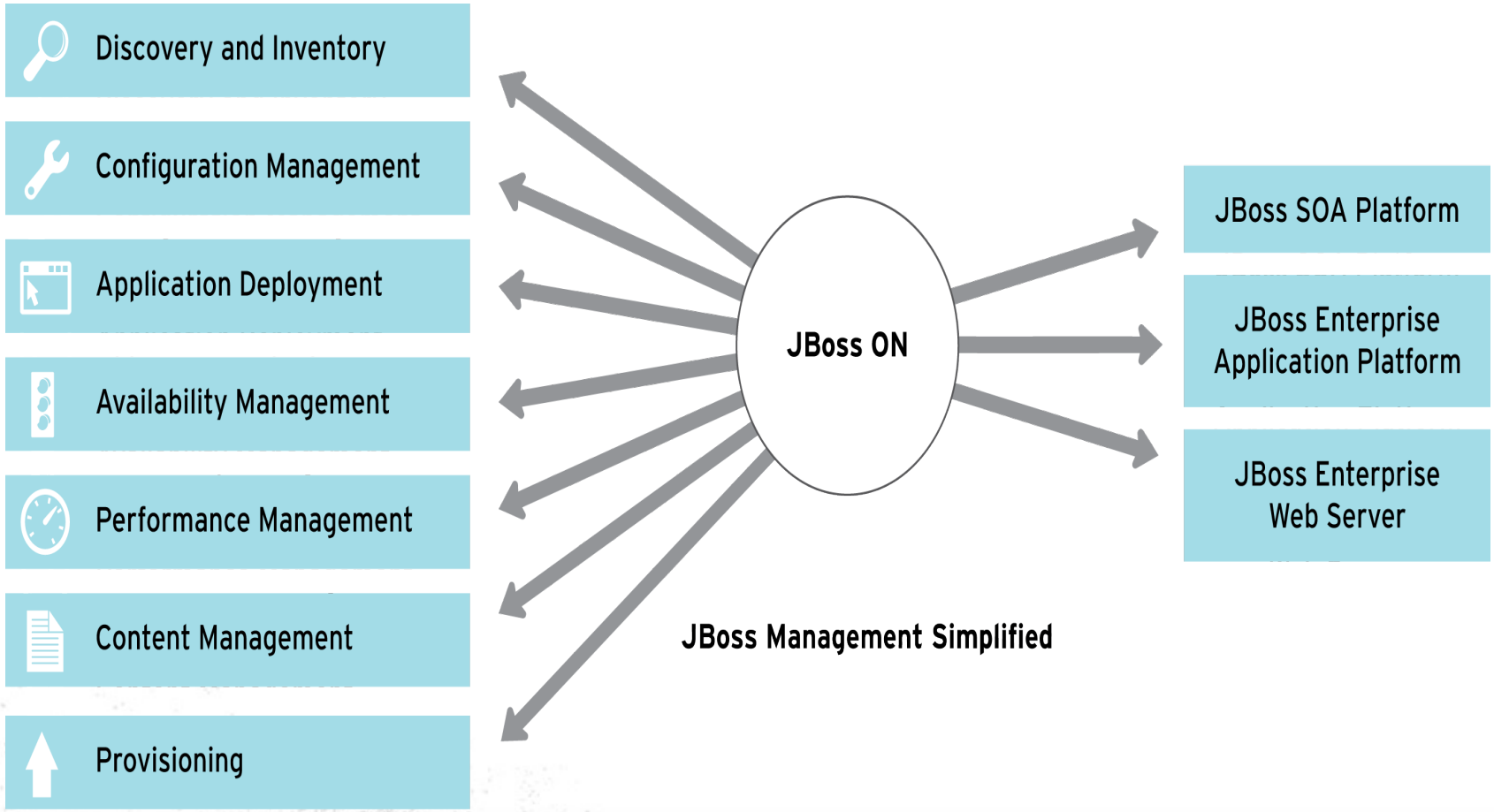
Performance

Availability

Alerts & Notifications

JBoss Operations Network

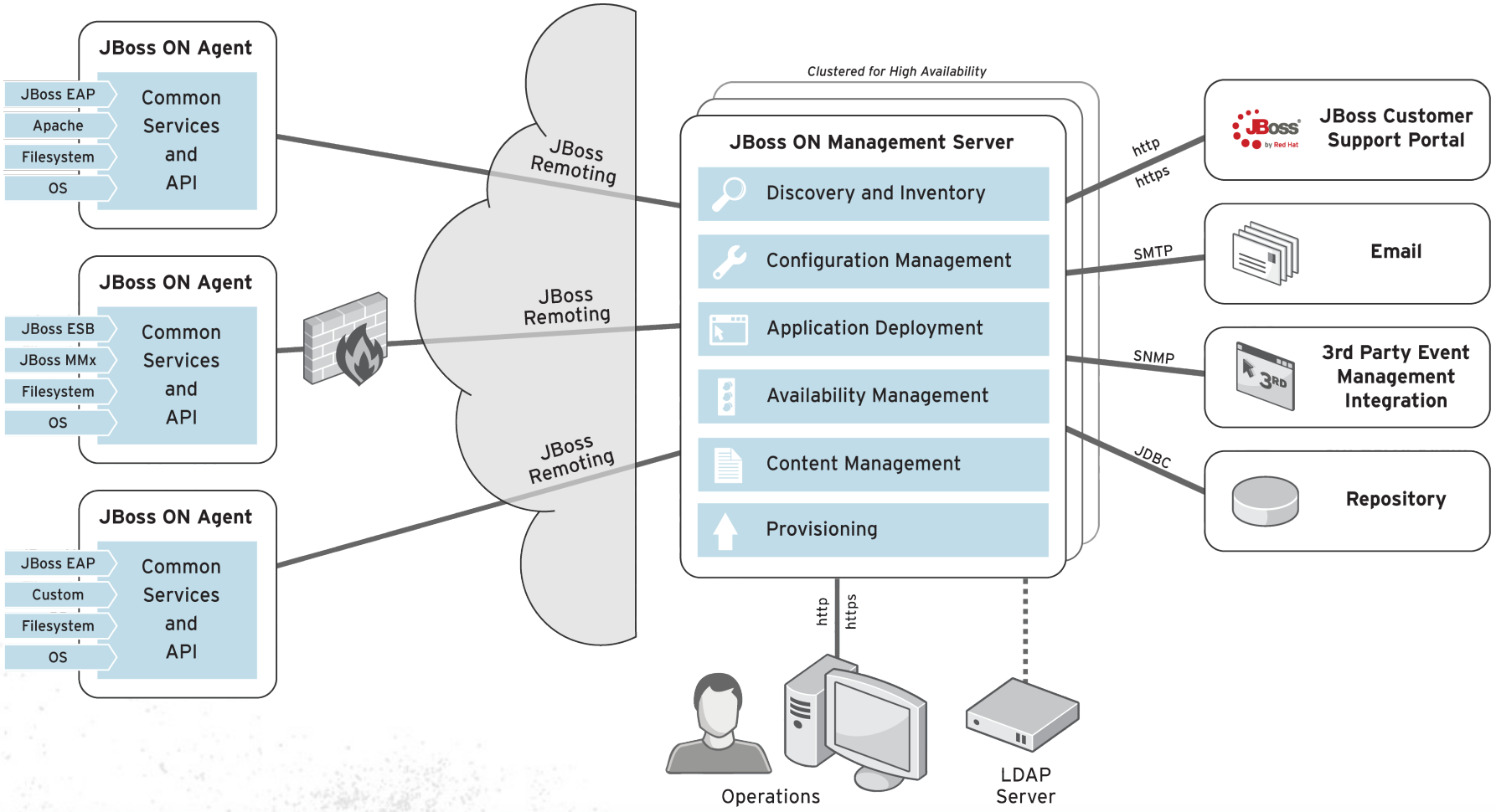
Solution Overview



Benefits

- Reduce software maintenance and management costs
- Manage application service levels (availability & performance)
- Simplify JBoss provisioning and deployment (infrastructure and applications)
- Improve visibility and communication between Development and IT Operations
- Increase IT operational efficiency and support IT governance objectives

JBoss ON - Architecture



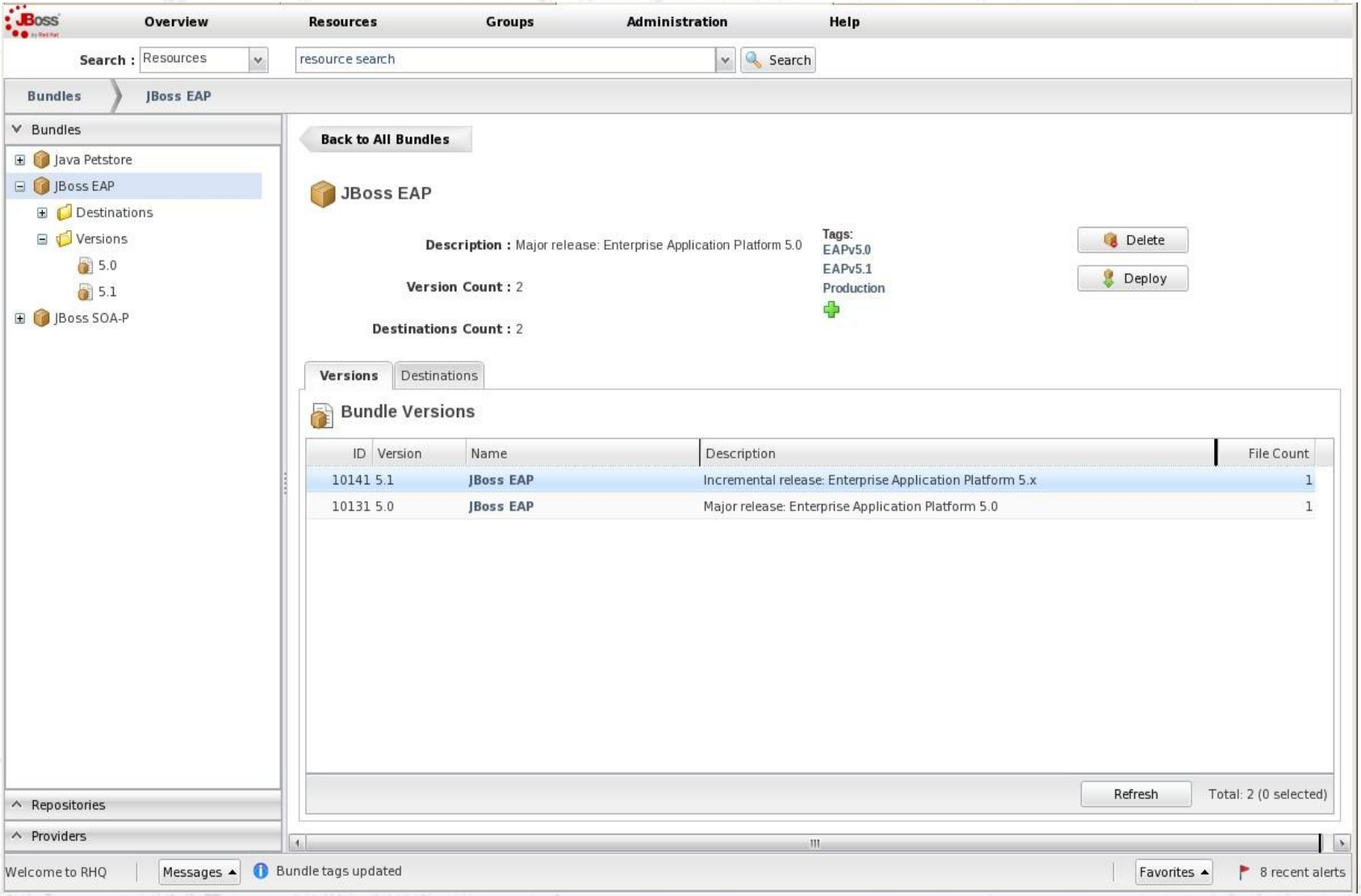
Use Cases for JBoss Operations Network

- Deploying a new application version
- Monitoring application performance in production
- Testing an application prior to deployment

UC1: Deploying a New Application Version

- Use Case: Customer has a new version of their application that they need to deploy to production
- Challenges:
  - Multiple servers in production running different versions of JBoss
  - Multiple applications & versions exist
  - Different configurations for the same application
- How JON Can Help:
  - Maintains an inventory of JBoss servers & different versions
  - Manages multiple versions of the application from one console
  - Allows you to quickly deploy the application and required configuration to a group.
  - Allows you to tie upgrade to to application performance history

Deploying a New Application with JON

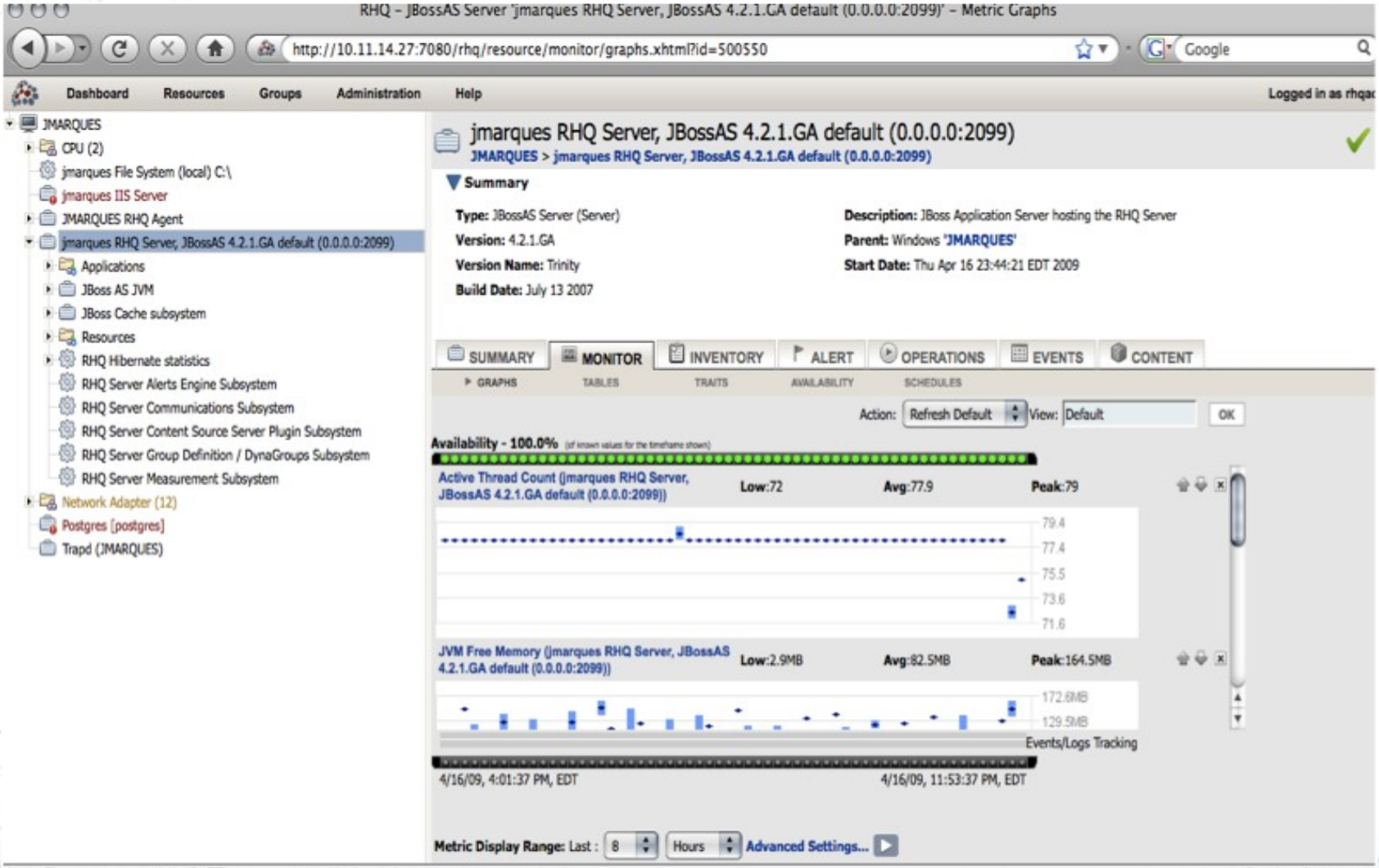


UC2: Monitoring Application Performance

- Use Case: Customer has a production application experiencing intermittent performance issues
- Challenges:
  - Many different applications running in production and multiple JBoss servers supporting each application
  - Understanding if application service levels are being met

- How JON Can Help:
  - Monitors performance of multiple applications and servers
  - Automatically alerts you when application metrics fail specified thresholds / service levels not being met
  - Allows you to automatically run operations like starting additional servers in response to an alert

Monitoring Application Performance with JON



UC3: Testing Application Prior to Deployment

- Use Case: Customer needs to test the performance of a new application in development before rolling it out to production
- Challenges:
  - Deploying the application in a test staging environment in order to run load tests against it
  - Isolating performance bottlenecks during the load test
  - Applying updates and patches to tune application performance and migrating same to production
- How RHQ/JON Can Help:
  - Quickly deploy JBoss applications in Dev/Test environments
  - Monitor application infrastructure performance during load test
  - Apply patches and updates & help manage production deployment
  - Ensures consistency from Dev/Test to Production environments

JON 3.0 Features

- **UI Improvements** - A rewritten GUI offers a new style and more efficient AJAX driven architecture that preserves a familiar layout with a cleaner look and more responsive experience. Other UI improvements include a new dynamic, customizable dashboard for both global views and specific resources.
- **Drift Management and Repair** - JBoss ON already provides a secure, audit trail for configuration changes and the ability to notify operators when changes occur using the console. Drift management lets users 'lock-down' the configuration of managed JBoss servers and related content to address change that occurs outside of the console.
- **Responsive Provisioning and Configuration** - Based on performance and availability users may automate resource configuration or application deployment. This offers a number of options to increase automation, improve availability or meet other SLAs, including the ability to dynamically scale JBoss applications and services up or down based on current state and performance requirements.