



redhat.

# JBossON Management platform and RHQ, its open source upstream

Heiko W. Rupp <[heiko.rupp@redhat.com](mailto:heiko.rupp@redhat.com)>

Java User Group Switzerland 11.11.2008

# Agenda

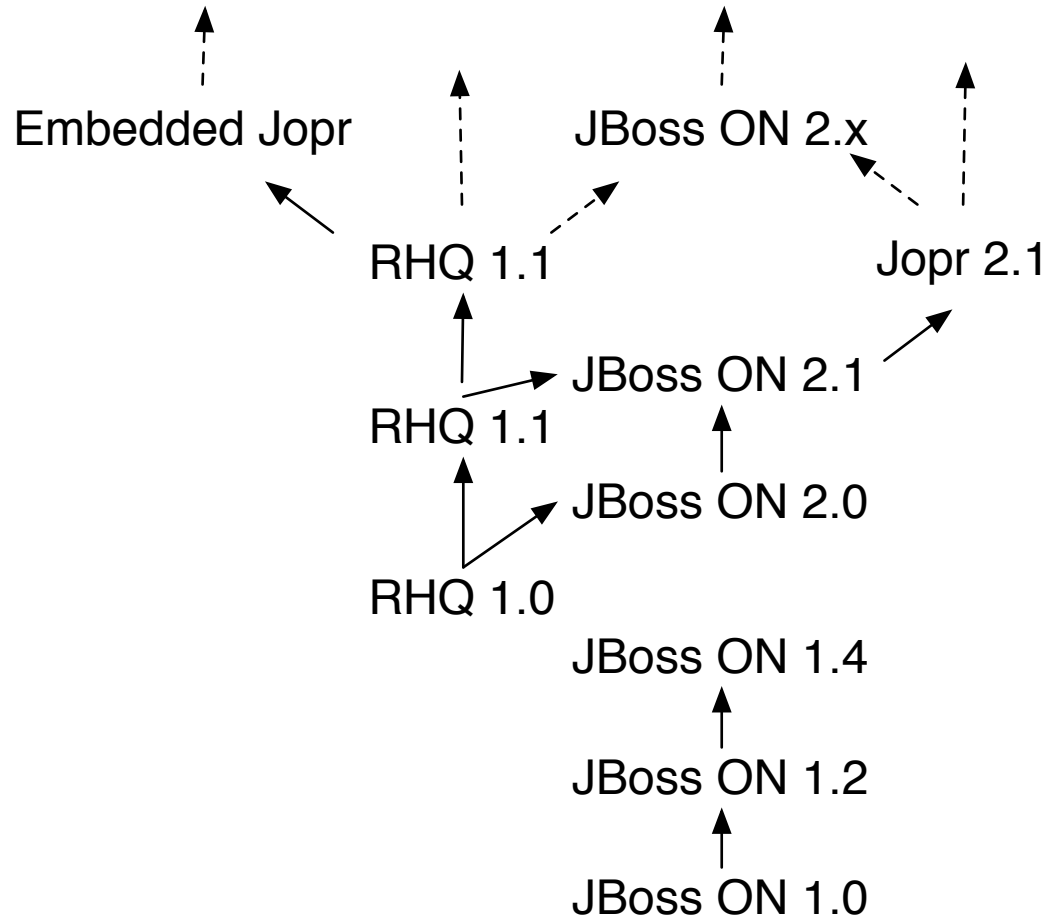
- Introduction
- History
- Architecture
- Subsystems
- Demo
  
- Development of Plug-ins



# Introduction

- RHQ - Common Project by Red Hat and Hyperic
- RHQ upstream for JBoss Operations Network 2 (JBossON)
- RHQ is Open Source since February 2008
  - Start page: <http://www.rhq-project.org/>
  - Forums: <http://forums.rhq-project.org/>
  - JIRA: <http://jira.rhq-project.org/secure/Dashboard.jspa>
  - Wiki: <http://support.rhq-project.org/display/RHQ/Home>
  - SVN: <http://www.rhq-project.org/svn.php>
  - Plugin-Community: <http://support.rhq-project.org/display/RHQ/RHQ+Plugin+Community>
- Jopr contains the remaining parts and is upstream for JBossON
- Embedded Jopr is a replacement for JMX-Console
  - Both build on top of RHQ
  - <http://www.jboss.org/jopr/> and <http://www.jboss.org/embjopr/>

# History

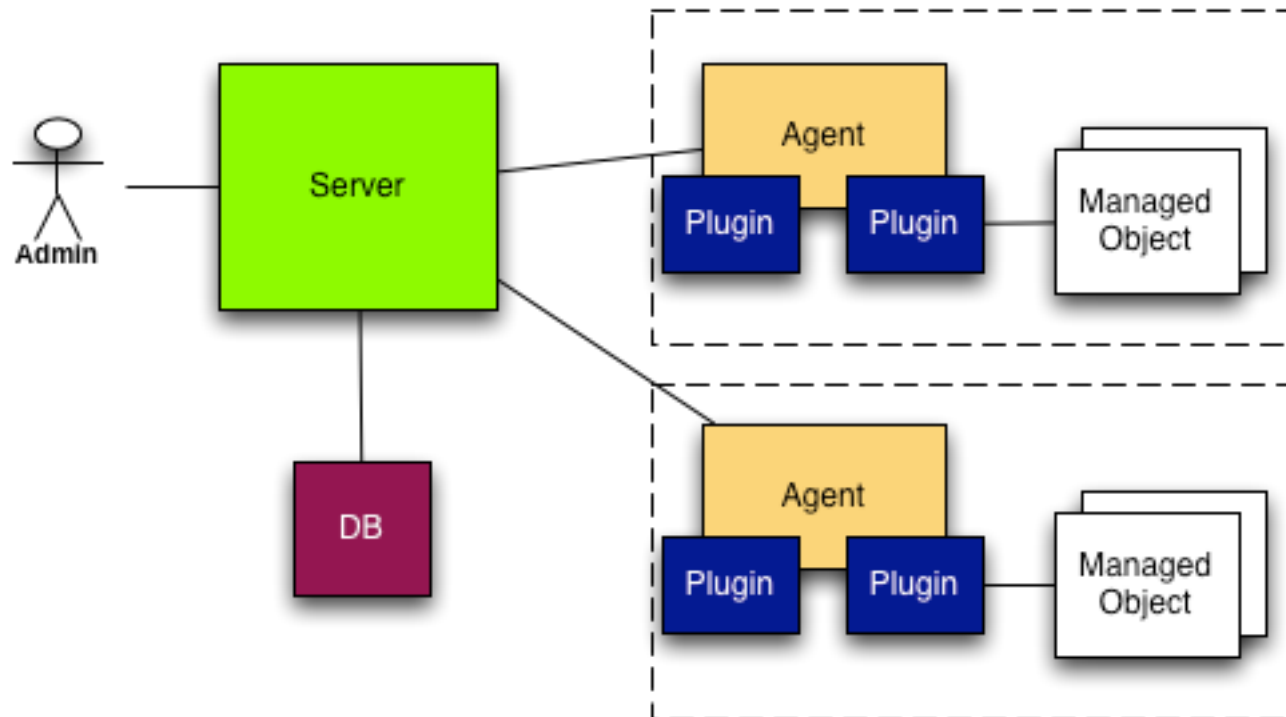


## RHQ 1.x

- To a large extent written from scratch and uses
  - JBoss AS 4.2.1
  - JBoss Remoting
  - EJB 3
  - JPA / Hibernate
  - JSF / Richfaces
- SIGAR Library by Hyperic (also in the Open Source)
- Parts of the GUI are still from JBoss ON 1.x
- Parts are written from scratch in JSF

# Architecture

- Central Server or cluster thereof
- Agent per Machine („platform“)

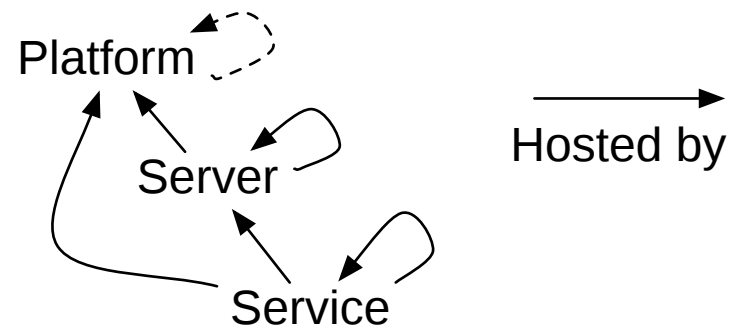
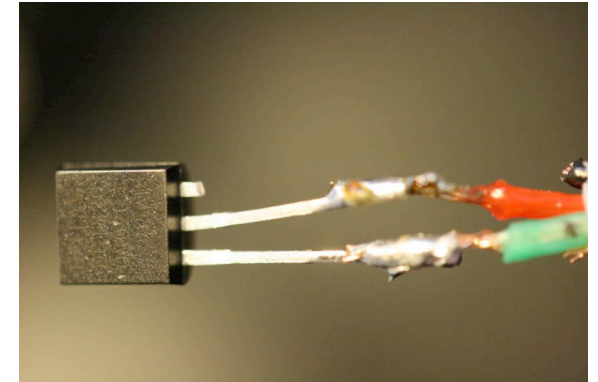


# Subsystemes

- Monitoring
  - Measurement of values and states
  - Graphs of metrics on the server GUI
- Inventory
  - The central component that knows all components
- Configuration
  - Configuration of Resources
- Alerts
  - Alerting of operators
  - Automatic actions
- Operations
  - Execute operations on resources
- Packages
  - Software upload on to Resources

# Resources

- Resources
  - Everything that can be managed / monitored
- Resource Type
  - Z.B. Linux, Max OS X, JBoss AS, Hibernate Stats, DB-Tabelle
- Resource Category
  - Place in the resource hierarchy
  - Parent - Child - Relation
  - Platform = machine

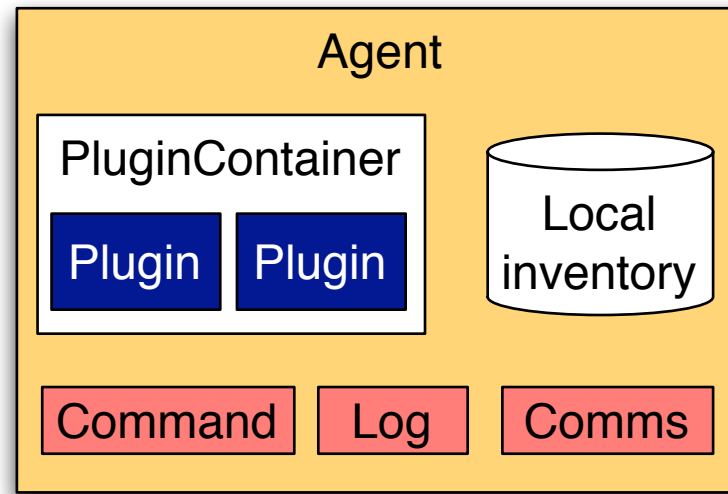




# Demo

## Development of Plug-ins

- Plug-ins will be installed in the plug-in container

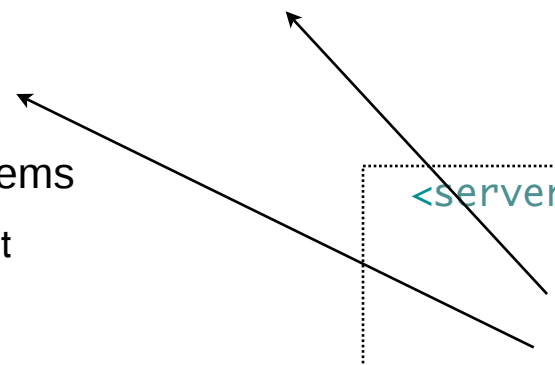


- Hot-Deploy via server
  - Agent pulls the plugins down and restarts the Plug-in-Container

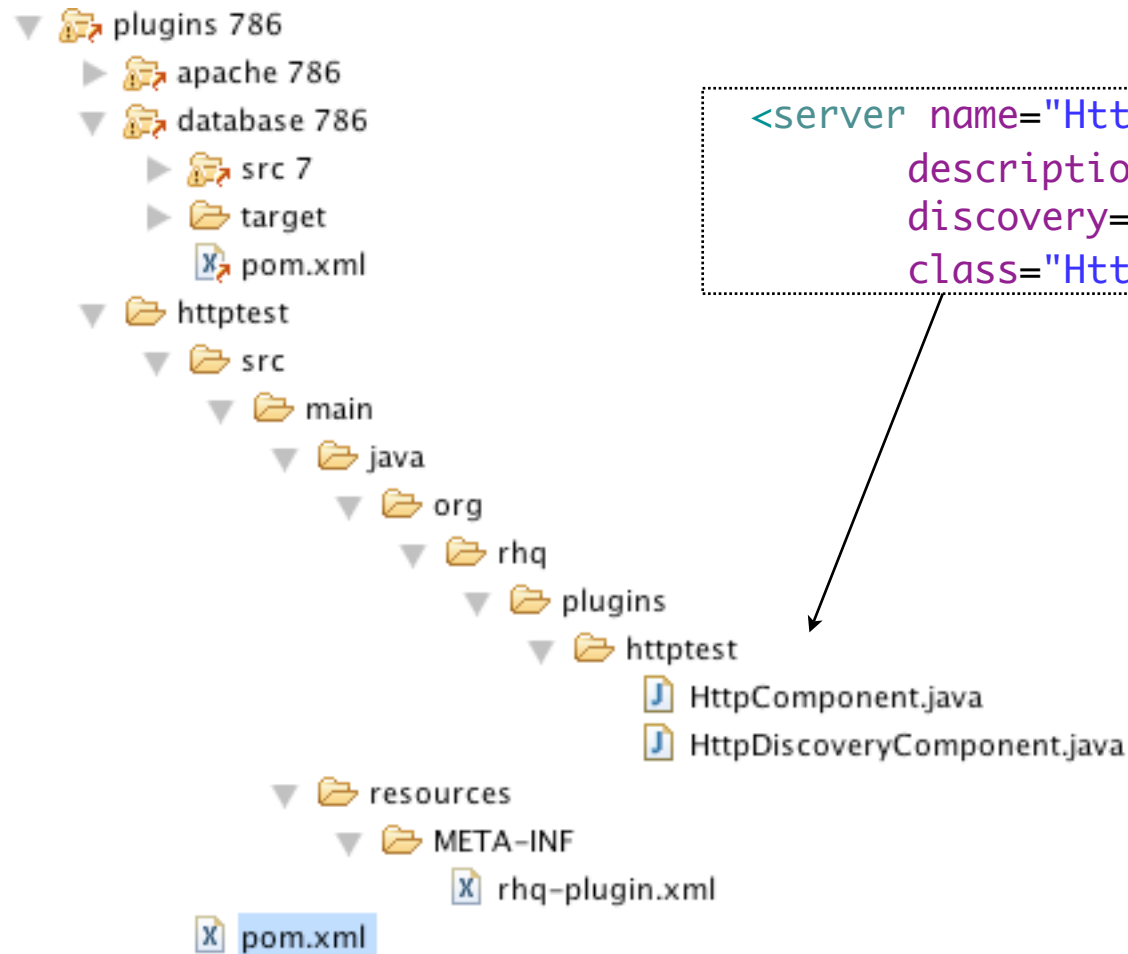
## 3 Components of a Plug-in

- Discovery Component
  - Responsible to discover and add new resources
  - Resource key must be and stay unique !
- Plugin Component
  - Implements *Facets*
  - Analog to the subsystems
    - MeasurementFacet
    - OperationsFacet
    - ...
- Plugin-Descriptor
  - Defines the features of the plug-in as metadata
  - Wires the Java classes

```
<server name="HttpCheck"  
description="Httpserver pinging"  
discovery="HttpDiscoveryComponent"  
class="HttpComponent">
```



# File system structure



```
<server name="HttpCheck"
description="Httpserver pinging"
discovery="HttpDiscoveryComponent"
class="HttpComponent">
```

## More ...

- JBoss ON 2 Docs: <https://docs.jbosson.redhat.com/confluence/display/JON2/Home>
- JBoss ON Homepage: <http://www.jboss.com/products/jbosson>
- JBoss ON Forum: <http://www.jboss.com/index.html?module=bb&op=viewforum&f=285>
- Blog-Postings about Plug-in development: <http://pilhuhn.blogspot.com/2008/05/writing-rhq-plugin-part-1.html> -- 5 postings