Introduction to JON 2.0

Greg Hinkle, Red Hat

February 13th, 2008

Agenda

- Overview
- Architecture
- JON 2 Enhancements
- Extension and Integration
- Project Status

Overview

- IT Operations must administer and maintain multiple applications across multiple environments
  - Applications are deployed across development, QA, staging and production
  - Multiple versions deployed
  - Inconsistent and un-audited configurations
  - Varying access privileges
  - Inconsistent release and promotions cycles
- Developers and administrators must use varying and inconsistent configuration mechanisms
  - Developers don't know what features are deployed and what their impact is
  - Configuration don't get optimized

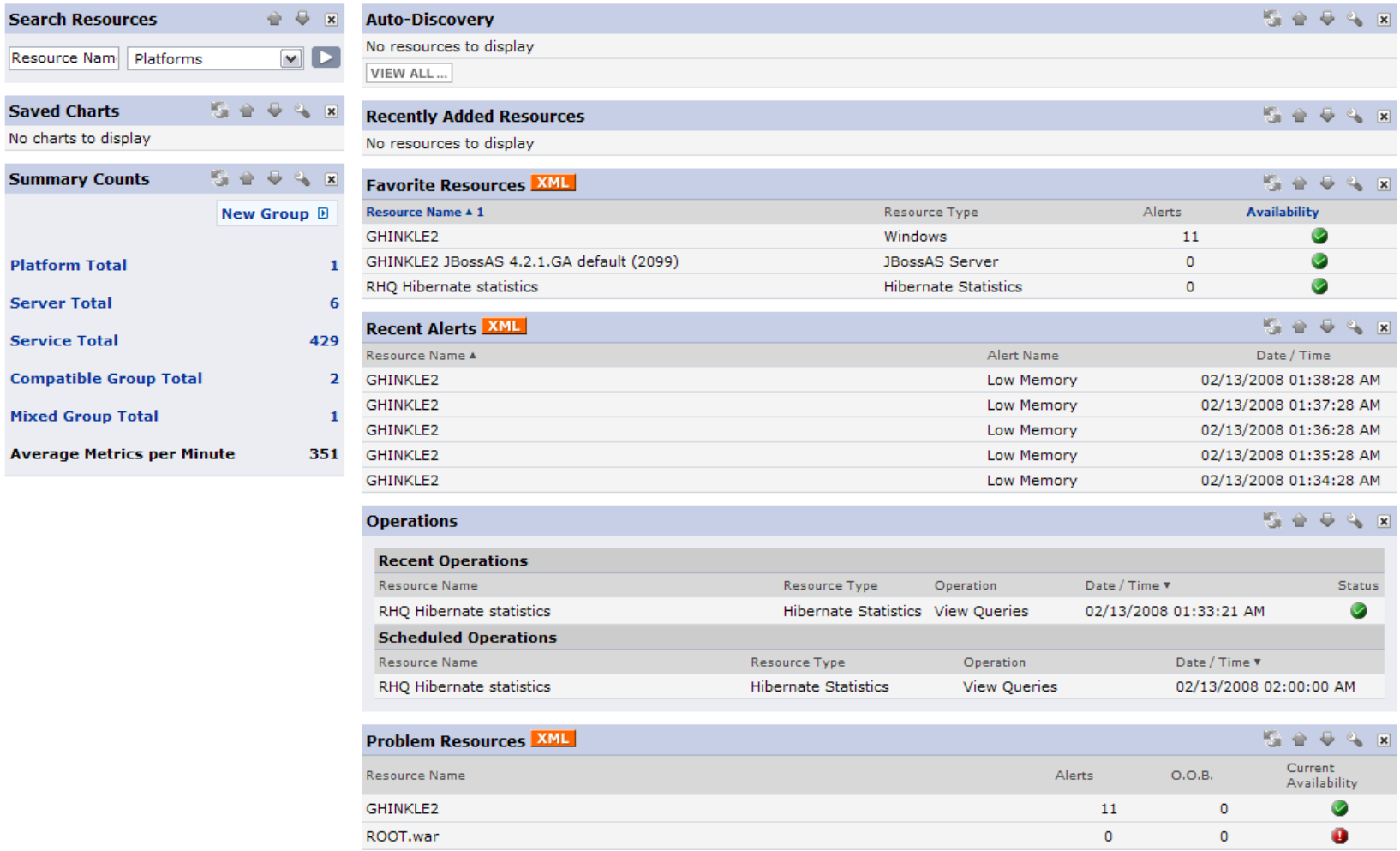
Overview

- JBoss Operations Network delivers enterprise visibility into your JBoss middleware infrastructure
- This centralized management tool delivers discovery and inventory, monitoring, administration and configuration, operational control and software patching
- Built on an agent-server architecture and supports visibility of low-level resources, including cpu, disk and network utilization
- Automated discovery for JBoss AS instances and services and applications deployed within
- Delivery of software notifications and audited installation of certified cumulative patches

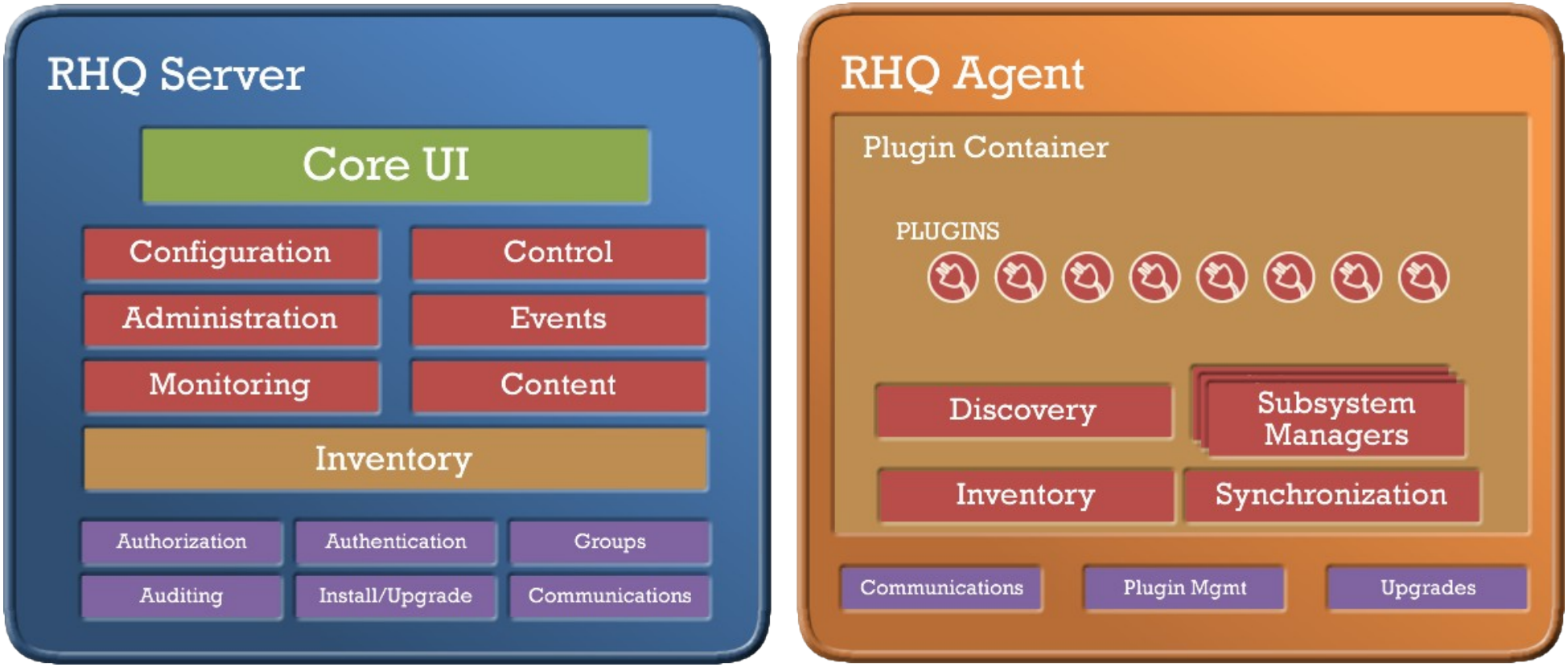
Features

- Monitors low level OS statistics (CPU, memory, swap, IO, disks)
- Monitors processes and services (Apache, JBoss, Postgres)
- Monitors detailed services (EJBs, URL response times, Datasources and Queues)
- Alerts and baselines on all monitored metrics
- Manages configuration of resources (Datasources, Queues, Deployments of Ears and Wars)
- Operational controls with scheduling and auditing
- JBoss AS software inventory and patch installation

Dashboard



Architecture



Support

- Server and agent deployments support on Linux, Windows, Solaris, HP-UX and AIX
- Java 1.5 required for server and agent
- PostgreSQL and Oracle are supported data stores

Enhancements: Features

- Improved configuration interface for more complex configurations and validation
- Configuration history stored and changes detected
- Faster detection of resources being unavailable
- Enhanced resource hierarchy supports more complex service modeling
- Improved group operation execution
- Improved managed deployments with auditing and versioning
- Inventorying of installed application server libraries and versions
- Method performance monitoring for EJB3

Configuration History

Version: 550
Status: Success
User Who Requested Change: (unknown)
Date Submitted: Tue Feb 12 15:37:23 EST 2008
Date Completed: Tue Feb 12 15:37:23 EST 2008

Configuration History

Version	Date Submitted	Date Completed	Status	User
550 *	Tue Feb 12 15:37:23 EST 2008	Tue Feb 12 15:37:23 EST 2008	Success	

DELETE SELECTED ROLLBACK SELECTED Total: 1 Items Per Page: 15

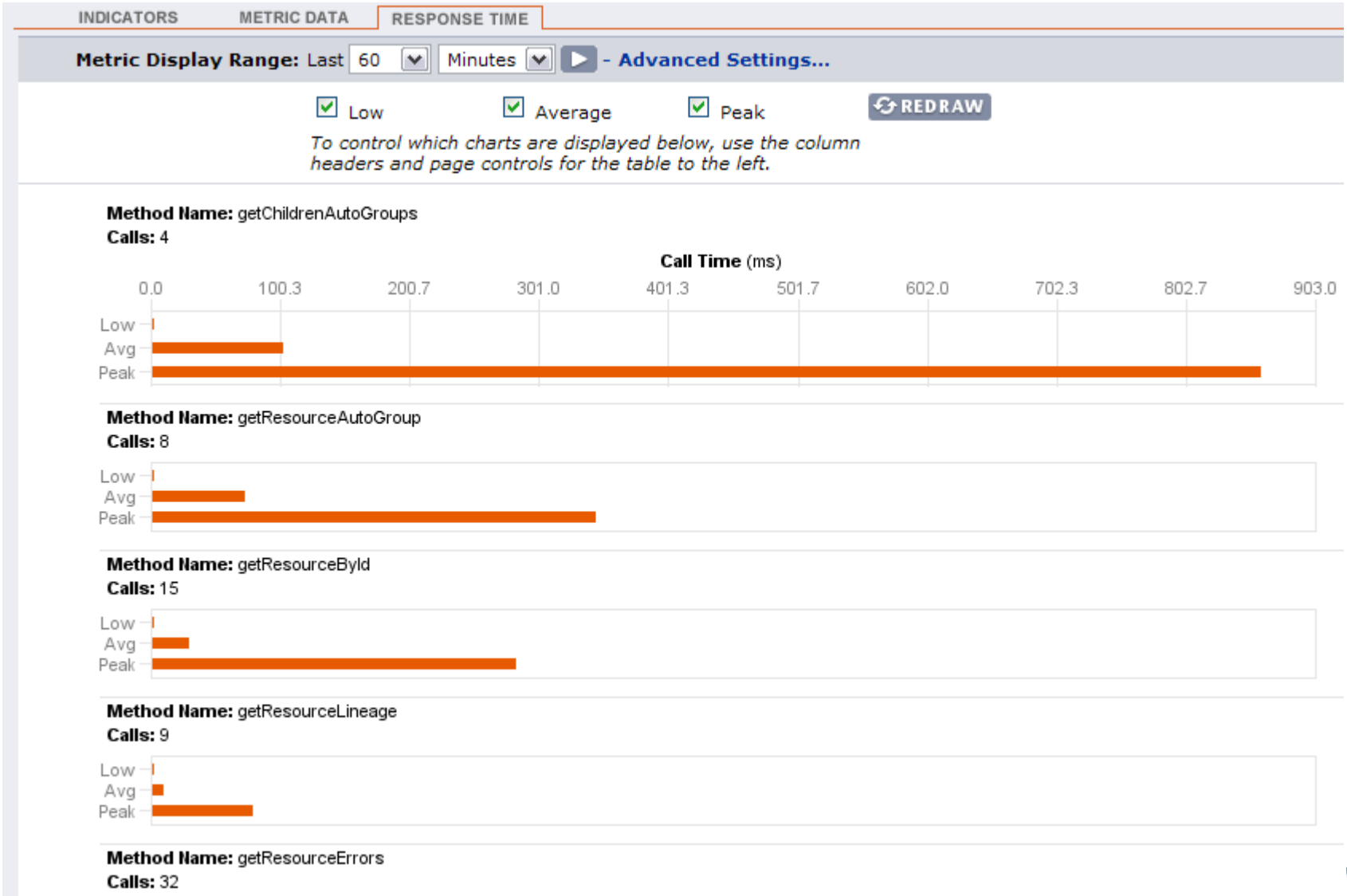
Local TX Datasource
XA Datasource

JNDI Name	RHQDS	*The JNDI name under which the DataSource wrapper will be bound.*
Driver Class	${rhq.server.database.driver-cla	*The fully qualified name of the JDBC driver or datasource class*
Connection Url	${rhq.server.database.connectic	*The JDBC driver connection URL string*
User Name	${rhq.server.database.user-nam	
Password	*********	
Min Pool Size	10	
Max Pool Size	100	

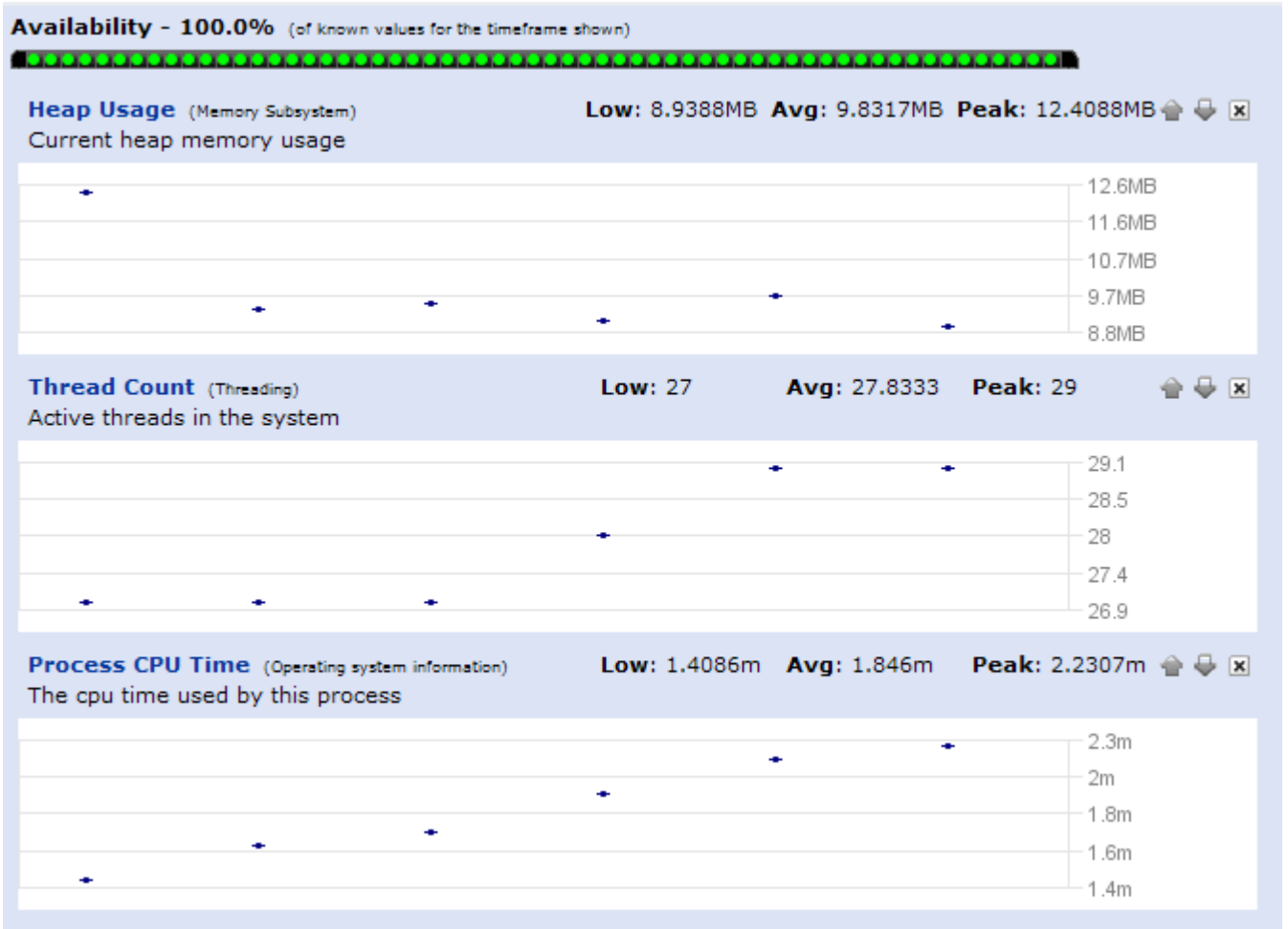
Advanced Collapse

Name	Unset	Value	Description
Transaction Isolation		Read Uncommitted / Read Committed / Repeatable Read / Serializable / None	*The Transaction Isolation level. The default setting is to use whichever isolation level is provided by default by the database.*
Blocking Timeout Millis		30000	*Indicates the maximum time in milliseconds to block while waiting for a connection before throwing an exception. Note that this blocks only while waiting for a permit for a connection, and will never throw an exception if creating a new connection takes an inordinately long time. If nothing is specified the default of 30000 milliseconds is used.*
Idle Timeout Minutes		15	*Indicates the maximum time in minutes a connection may be idle before being closed. The actual maximum time depends also on the IdleRemover scan time, which is 1/2 the smallest idle-timeout-minutes of any pool. If nothing is specified the default of 15 minutes is used.*
Prepared Statement Cache Size		75	*The number of prepared statements per connection in an LRU cache.*
Valid Connection Checker Class Name			*An org.jboss.resource.adapter.jdbc.ValidConnectionChecker that provides a SQLException isValidConnection(Connection e) method to validate is a connection is valid. An exception means the connection is destroyed. This overrides the*

EJB3 Method Monitoring



JVM Monitoring



Enhancements: Extensibility

- New plugin API is simpler to implement and allows broader integration
- Extensible JMX plugin allows easier support for developing custom JMX based management
- Configuration can now be implemented by any plugin
- Advanced operations can take complex parameters and return results
- Non-native mode allows support for any platform with a 1.5 Java Runtime Environment

Example Plugin Descriptor

```
<service name="Threading" discovery="MBeanResourceDiscoveryComponent" class="ThreadDataMeasurementComponent"
         description="Information on threading in the VM"
         createDeletePolicy="neither" singleton="true">
   <plugin-configuration>
      <c:simple-property name="objectName" readOnly="true" default="java.lang:type=Threading"/>
      <c:simple-property name="nameTemplate" default="Threading"/>
   </plugin-configuration>

   <operation
      displayName="Reset Peak Thread Metrics"
      name="resetPeakThreadCount"
      description="Reset the peak number of threads"/>

   <operation name="findMonitorDeadlockedThreads" description="Finds cycles of threads that are in deadlock waiting to acq

   <operation name="threadDump">
      <results>
         <c:simple-property name="totalCount"/>
         <c:list-property name="threadList">
            <c:map-property name="thread">
               <c:simple-property name="name" summary="true"/>
               <c:simple-property name="id" summary="true"/>
               <c:simple-property name="state" summary="true"/>
               <c:simple-property name="stack" type="longString"/>
            </c:map-property>
         </c:list-property>
      </results>
   </operation>

   <metric displayName="Thread Count"
           defaultOn="true"
           displayType="summary"
           category="performance"
           property="ThreadCount"
           description="Active threads in the system"/>
```

Enhancements: Scalability

- Greatly increased the amount of data that can be monitored by one server
- Reduced load on managed servers by allowing some batch data collection
- Optimized data storage to allow for more data to be collected and stored for a given db infrastructure
- Improved data model reduces load and increases the speed of user interface and agent interaction
- Stateful plugin model allows for optimized managed server connection maintenance
- Dynamic Groups for easier maintenance of large inventories

Dynamic Groups

Group Definition's Managed Resource Group

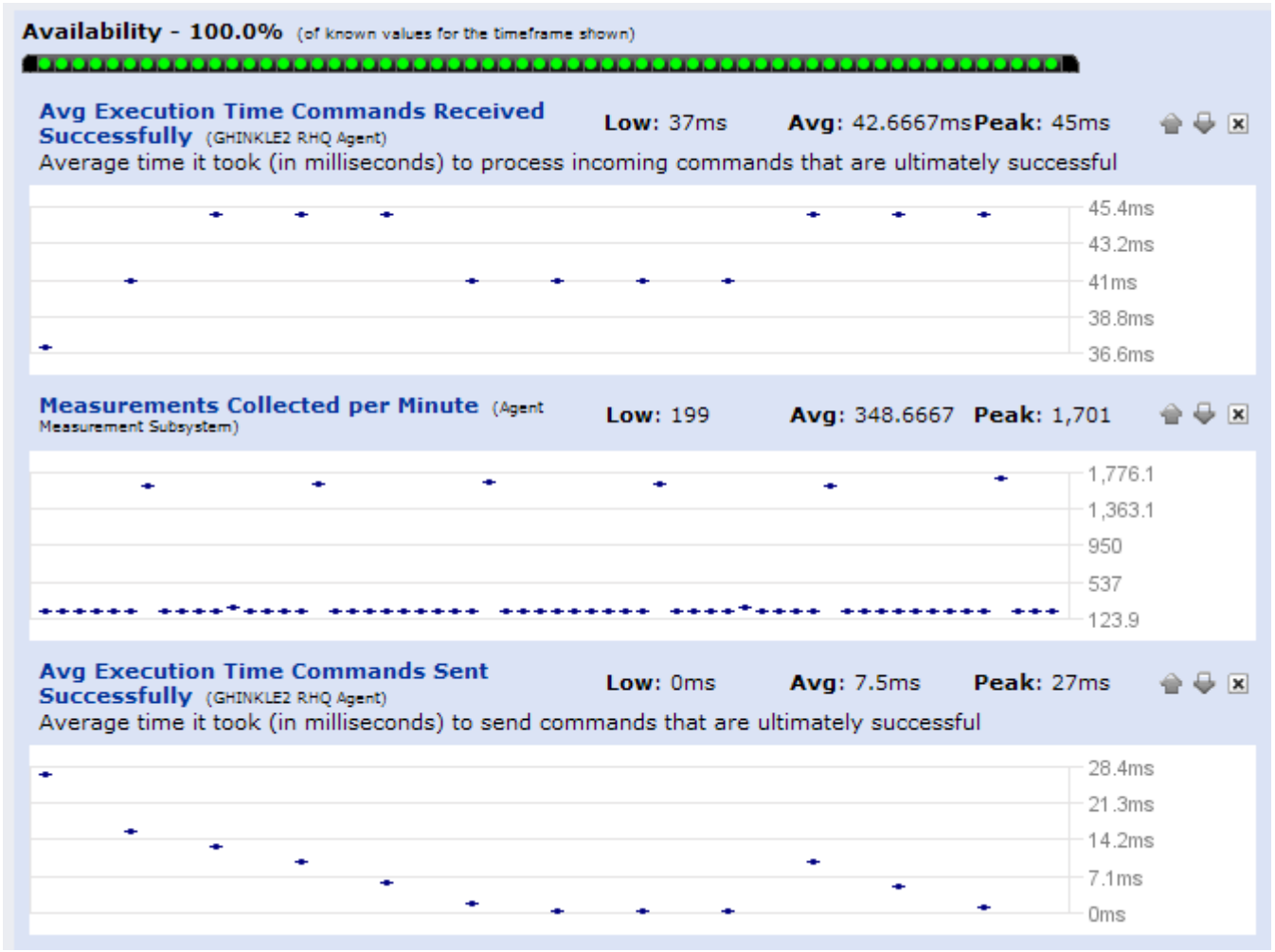
Name	Category	Grouped By	Members	Availability
DynaGroup - GroupByParentAndType (CPU,Platforms,GHINKLE2)	compatible group	CPU,Platforms,GHINKLE2	2	
DynaGroup - GroupByParentAndType (Connector,JBossAS,GHINKLE2 Embedded JBossWeb Server 2.0.0.GA (127.0.0.1))	compatible group	Connector,JBossAS,GHINKLE2 Embedded JBossWeb Server 2.0.0.GA (127.0.0.1)	3	
DynaGroup - GroupByParentAndType (Database,Postgres,Postgres [postgres])	compatible group	Database,Postgres,Postgres [postgres]	3	
DynaGroup - GroupByParentAndType (Datasource,JBossAS,GHINKLE2 JBossAS 4.2.1.GA default (2099))	compatible group	Datasource,JBossAS,GHINKLE2 JBossAS 4.2.1.GA default (2099)	4	
DynaGroup - GroupByParentAndType (EJB3 Entity Tree Cache Interceptor,JBossAS,EJB3 Entity Tree Cache)	compatible group	EJB3 Entity Tree Cache Interceptor,JBossAS,EJB3 Entity Tree Cache	1	
DynaGroup - GroupByParentAndType (EJB3 Entity Tree Cache,JBossAS,GHINKLE2 JBossAS 4.2.1.GA default (2099))	compatible group	EJB3 Entity Tree Cache,JBossAS,GHINKLE2 JBossAS 4.2.1.GA default (2099)	1	
DynaGroup - GroupByParentAndType (EJB3 Session Bean,JBossAS,GHINKLE2 JBossAS 4.2.1.GA default (2099))	compatible group	EJB3 Session Bean,JBossAS,GHINKLE2 JBossAS 4.2.1.GA default (2099)	52	
DynaGroup - GroupByParentAndType (Embedded Tomcat Server,JBossAS,GHINKLE2 JBossAS 4.2.1.GA default (2099))	compatible group	Embedded Tomcat Server,JBossAS,GHINKLE2 JBossAS 4.2.1.GA default (2099)	1	
DynaGroup - GroupByParentAndType (Embedded Web Application (WAR),JBossAS,rhq.ear)	compatible group	Embedded Web Application (WAR),JBossAS,rhq.ear	2	
DynaGroup - GroupByParentAndType (Enterprise Application (EAR),JBossAS,GHINKLE2 JBossAS 4.2.1.GA default (2099))	compatible group	Enterprise Application (EAR),JBossAS,GHINKLE2 JBossAS 4.2.1.GA default (2099)	1	
DynaGroup - GroupByParentAndType (File System,Platforms,GHINKLE2)	compatible group	File System,Platforms,GHINKLE2	2	
DynaGroup - GroupByParentAndType (Garbage Collector,JBossAS,Memory Subsystem)	compatible group	Garbage Collector,JBossAS,Memory Subsystem	2	
DynaGroup - GroupByParentAndType (Garbage Collector,RHQAgent,Memory Subsystem)	compatible group	Garbage Collector,RHQAgent,Memory Subsystem	2	
DynaGroup - GroupByParentAndType (Hibernate Entity,Hibernate,RHQ Hibernate statistics)	compatible group	Hibernate Entity,Hibernate,RHQ Hibernate statistics	88	
DynaGroup - GroupByParentAndType (Hibernate Statistics,Hibernate,GHINKLE2 JBossAS 4.2.1.GA default (2099))	compatible group	Hibernate Statistics,Hibernate,GHINKLE2 JBossAS 4.2.1.GA default (2099)	1	

Total: 45 Items Per Page: 15 1 2 3

Enhancements: Reliability

- Pervasive reliable agent/server communications avoid problems with unreliable connections
  - Control operation results, configuration and software update actions are not lost
  - Command management and throttling reduces the risk of overloading the server after outages
- Simplified caching improves cluster reliability
- Better installation defaults supports easier growth
- Built-in platform monitoring helps to avoid problems

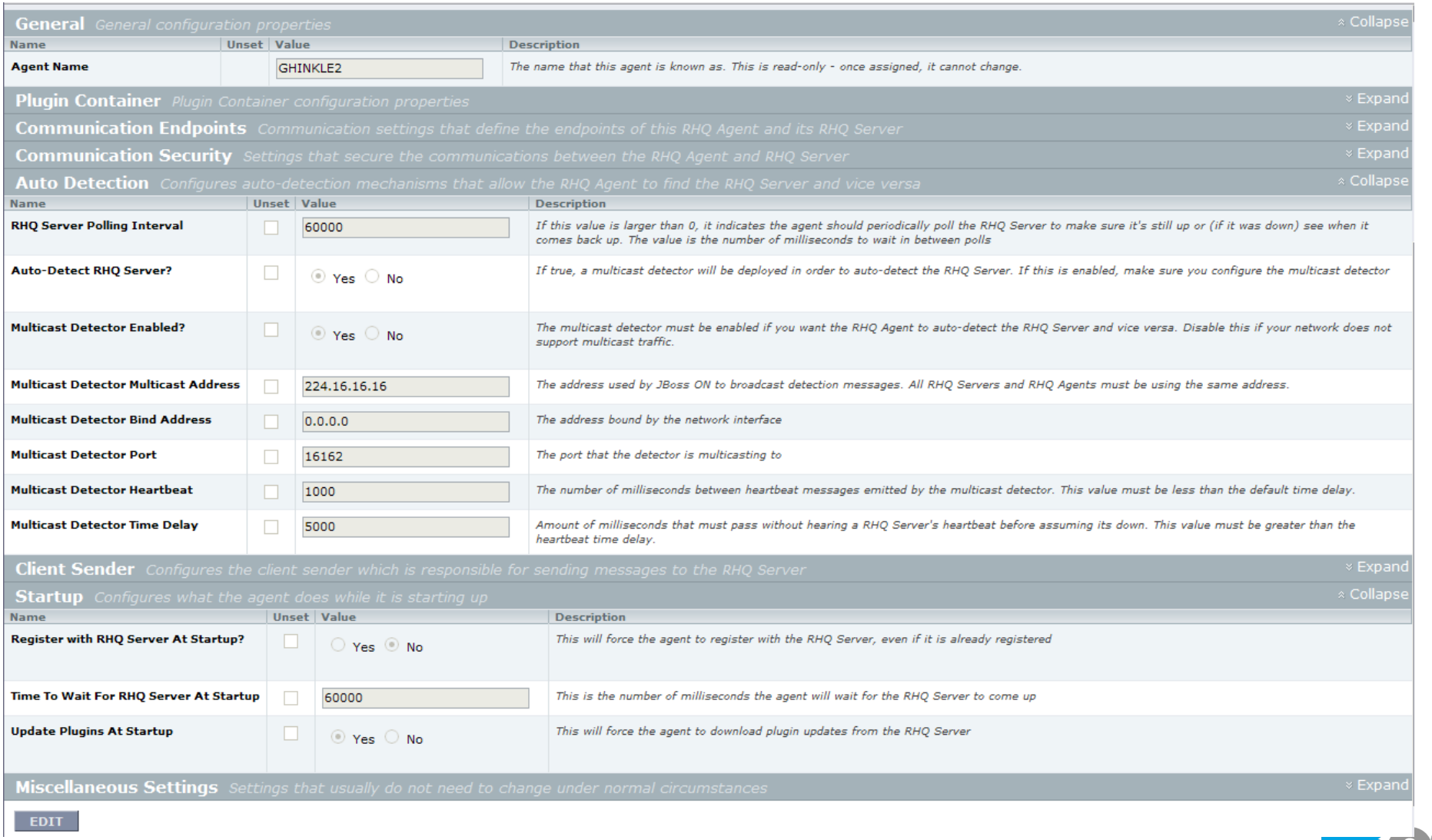
Monitoring JON



Enhancements: Simplification

- Agents will automatically download the latest plugins simplifying deployment
- Recursive and DynaGroup groups-based security greatly reduces authorization maintenance for large environments
- New security model simplifies access control of the inventory and administration features
- Remote agent configuration allows centralized control

Agent Configuration



General *General configuration properties* — Collapse

Name	Unset	Value	Description
Agent Name		GHINKLE2	*The name that this agent is known as. This is read-only - once assigned, it cannot change.*

Plugin Container *Plugin Container configuration properties* — Expand

Communication Endpoints *Communication settings that define the endpoints of this RHQ Agent and its RHQ Server* — Expand

Communication Security *Settings that secure the communications between the RHQ Agent and RHQ Server* — Expand

Auto Detection *Configures auto-detection mechanisms that allow the RHQ Agent to find the RHQ Server and vice versa* — Collapse

Name	Unset	Value	Description
RHQ Server Polling Interval	☐	60000	*If this value is larger than 0, it indicates the agent should periodically poll the RHQ Server to make sure it's still up or (if it was down) see when it comes back up. The value is the number of milliseconds to wait in between polls*
Auto-Detect RHQ Server?	☐	◉ Yes ○ No	*If true, a multicast detector will be deployed in order to auto-detect the RHQ Server. If this is enabled, make sure you configure the multicast detector*
Multicast Detector Enabled?	☐	◉ Yes ○ No	*The multicast detector must be enabled if you want the RHQ Agent to auto-detect the RHQ Server and vice versa. Disable this if your network does not support multicast traffic.*
Multicast Detector Multicast Address	☐	224.16.16.16	*The address used by JBoss ON to broadcast detection messages. All RHQ Servers and RHQ Agents must be using the same address.*
Multicast Detector Bind Address	☐	0.0.0.0	*The address bound by the network interface*
Multicast Detector Port	☐	16162	*The port that the detector is multicasting to*
Multicast Detector Heartbeat	☐	1000	*The number of milliseconds between heartbeat messages emitted by the multicast detector. This value must be less than the default time delay.*
Multicast Detector Time Delay	☐	5000	*Amount of milliseconds that must pass without hearing a RHQ Server's heartbeat before assuming its down. This value must be greater than the heartbeat time delay.*

Client Sender *Configures the client sender which is responsible for sending messages to the RHQ Server* — Expand

Startup *Configures what the agent does while it is starting up* — Collapse

Name	Unset	Value	Description
Register with RHQ Server At Startup?	☐	○ Yes ◉ No	*This will force the agent to register with the RHQ Server, even if it is already registered*
Time To Wait For RHQ Server At Startup	☐	60000	*This is the number of milliseconds the agent will wait for the RHQ Server to come up*
Update Plugins At Startup	☐	◉ Yes ○ No	*This will force the agent to download plugin updates from the RHQ Server*

Miscellaneous Settings *Settings that usually do not need to change under normal circumstances* — Expand

EDIT

Status

- JON 2.0 will be available this spring
- The core of JON in the form of the RHQ project is being released to open source this week (come to the RHQ presentation Tomorrow at 3:00 for more details)
- A hands on session with JON 2.0 beta 2 will be held Friday morning at 9:00

Questions?