JMX in Java & Management Environments (DSOM 2001)

Christophe Ebro
JMX Foundation Spec Lead
Java Technology For Service Providers

JMX in Java & Management Environments
? Status
? JMX and Java environments
? JMX and Management environments
? Evolution
? Q & A

? 60% market share for app server mgt
? 14 Major J2EE App Server vendors in JSR 77 expecting JMX to become a J2EE Standard Service
? Huge amount of JMX downloads
? JMX books:
  – One chapter in a Prentice Hall book (beg 2002)
  – One O'Reilly book (mid-2002)
? In major conferences (NOMS, IEEE, OMG)
Status

JMX and Java environments

JMX and Management environments

Evolution

Q & A

Standard optional package for generic management, widely adopted

Possible following step could be to have JMX delivered along with the VM itself as standard way of doing management:
– For user management
– To monitor and manage the VM itself

Your opinion?
The Management/Monitoring of the VM must be transparent: no need to have additional stuff for management, or to know anything about it:

- A possible solution could be the use of Dynamic MBeans representing the VM Java objects to be managed/monitored
- Your ideas/feedback/requirements?

- Already used by all major app server vendors for app server management
- App server vendors involved in JSR77 have requested that JMX become a J2EE Standard Service
- Possible solution could be to include JMX API into J2EE 1.4 (with added security control)
- Your ideas/feedback/requirements?
Possible solutions to define a Management Interface concept, similar to the Remote one, but dedicated to Management:
- This interface would be referenced in the Deployment Descriptor
- It would automatically register itself into the JMX MBean Server

Proposal: 3 types of EJ Bs:
- Pure Business EJBs
  - Created by a Business client via Home
  - Accessed by Business client using Remote
  - No management
Manageable Business EJBs
- Created by a Business client via Home
- Accessed by Business client using Remote
- Manageable using operations described in their Management Interface (MBean), through the Container Management infrastructure (MBS)

Management EJBs (logs, monitors, etc.)
- Created by a Management client via Home
- Accessed by Management client using Remote
- Manageable using operations described in their Management Interface (MBean), through the Container Management infrastructure (MBS)
Using JMX for J2EE Application Deployment?

J2EE Connectors exposing the generic JMX Client API?
Allow J2EE developers to interface EISs in middle-tier apps by:
- Using a single interface
- Abstracting the underlying protocol

Your ideas/feedback/requirements?

J2ME: opportunities
### Status

- **JMX and Java environments**
- **JMX and Management environments**
- **Evolution**
- **Q & A**

### Management Standards in the Industry

- Because JMX is part of the Java platform, it is adopted along with Java, across industries, from Java-based Internet Appliances to the largest telecommunications systems.
- Because JMX is protocol and InfoModel agnostic, Java developers can plug into the management systems that are prevalent in their industry segment.

<table>
<thead>
<tr>
<th>Consumer Devices</th>
<th>Enterprise</th>
<th>Java Applications</th>
<th>Internet Infrastructure</th>
<th>Telecom Infrastructure</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Existing</strong></td>
<td>Existing</td>
<td>Existing JMX</td>
<td>Existing SNMP</td>
<td>Existing TMN</td>
</tr>
<tr>
<td>No management</td>
<td>SNMP</td>
<td></td>
<td></td>
<td>TL1</td>
</tr>
<tr>
<td>RMON</td>
<td>Proprietary</td>
<td></td>
<td></td>
<td>Proprietary</td>
</tr>
<tr>
<td><strong>Proprietary</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Emerging</strong></td>
<td>Emerging</td>
<td>Emerging CIM/WBEM</td>
<td></td>
<td>Emerging J2EE-based</td>
</tr>
<tr>
<td>Java &amp; Web-based</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Take It to the Next Level

- **Sun Management**

---

*Note: The table above shows the evolution of management standards across different domains, highlighting the adoption of JMX in Java environments.*
JSR 146:
- CIM <-> JMX mapping
- JMX WBEM provider

Possible to model using CIM and manage using any protocol
Possible to manage existing JMX agents using CIM/WBEM by:
- Modeling in CIM extending CIM schemas
- Generating “meditation” Mbeans in agents

(JMX->CIM mapping not reusing CIM schemas!)
OSS/J: opportunities

Application A
- J2EE Driver
- OSS/J Interface
- JMX or JMX/X Client API

Application B
- J2EE Connector
- J2EE Driver
- J2EE Adapter
- J2EE Connector

JMX to manage EJBs

- Status
- JMX and Java environments
- JMX and Management environments
- Evolution
- Q & A
- Bug fixing (ModelMBean, etc.)
- Complete OpenMBean implementation
- Minor extensions (to be identified)

- JMX 1.0 has security problem in MBeanServerFactory class
- Static methods allow anyone to get a handle on MBean servers
- Should add permission checking
Insert any object implementing MBeanServer between connector and real MBean server
Object can check permissions before forwarding operation

Need to define a Client interface for tunneling to the JMX MBean Server, in order to standardise the way a Java Manager (local or remote) can connect to a JMX Agent:
- Expose a single interface to client
- Hide the underlying tunneling protocol
- Provide Discovery and Naming
- Use the RemoteMBeanServer client API contribution as a basis
- Provide a Discovery mechanism based on a generic independent mechanism or JINI/JXTA
- Define Naming Conventions, for a manager to be able to access several managed applications simultaneously
Launch of a component market place, the JMXperience, for anybody to provide components around JMX to the community. The component provider is free to choose format (source or binary) and licencing conditions (free or for a fee).

- Status
- JMX and Java environments
- JMX and Management environments
- Evolution
- Q & A