

Leveraging the power of JBoss Portal with JBPM

Munwar Shariff
CTO, VP of Business Development



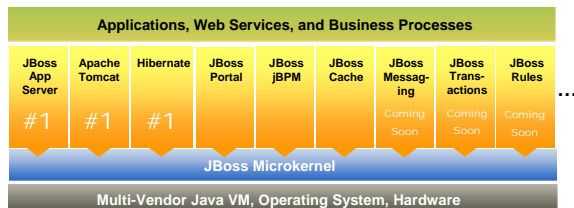
© JBoss Inc. 2006

Agenda

- JEMS in Enterprise – Challenges & Advantages of the stack
- JBoss jBPM Advantages
- JBoss Portal Features
- Motivation to integrate jBPM with JBoss Portal...**Business Case**
- Design Decisions
- Current Status of Development
- Next Steps...Future

2

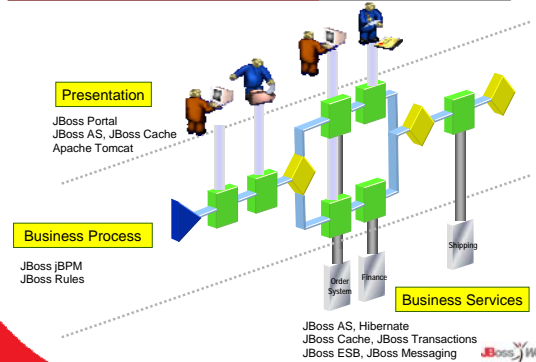
JEMS - JBoss Enterprise Middleware Suite



- JEMS is "The Open-Source Platform for SOA"
- Market-leading products for enhancing business agility
- Modularity enables standardizing on JEMS at your own pace
- Simplicity decreases time to develop and deploy new business solutions

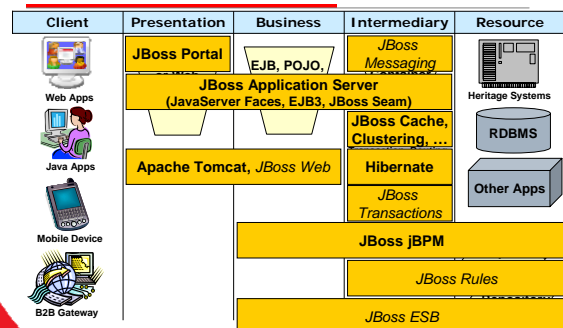
3

JEMS in Enterprise – Challenges & Advantages



4

JEMS – Part of JBoss SOA Strategy



5

JBoss jBPM

- Creates Business Processes to coordinate between
 - ✓ People
 - ✓ Applications
 - ✓ Services
- Implements various process languages
 - ✓ Workflow
 - ✓ Business Process Management (BPM)
 - ✓ Service orchestration (BPEL)
 - ✓ Web Application Pageflow

6

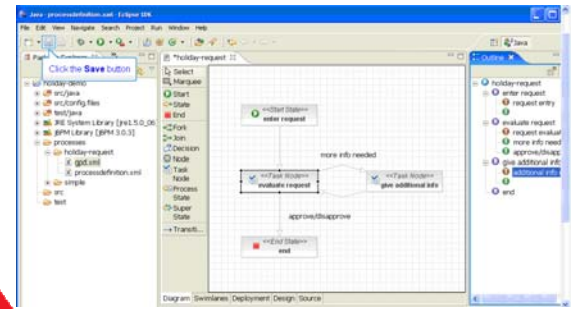
JBoss jBPM Advantages

- JBPM is built on solid foundation
 - ✓ Graph Oriented Programming
 - ✓ Task Management
 - ✓ Clean Java Integration
- JBPM can be easily deployed as a component in an enterprise application
- Open Source
- Open Standards based - BPEL
- Scales well to the enterprise



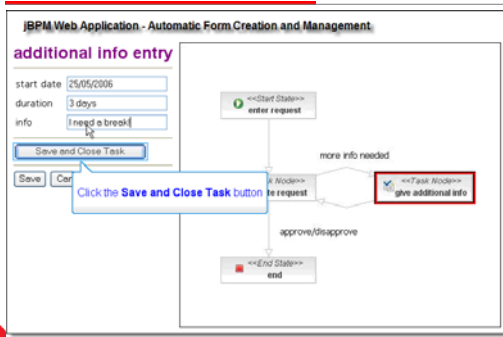
7

jBPM Graph Design Eclipse Plug-in



8

jBPM Graph deployed as Web App



9

JBoss Portal

- Well developed portal
 - ✓ Full portlet container implementation
 - ✓ CMS uses JCR (JSR-170) standards
 - ✓ Seamless integration with JBoss
 - Portlet deployment integrated at the JBoss deployment layer
 - Integrated Security
 - JSF integration
- Runs on JBoss AS
 - ✓ Inherits cluster capability from JBoss AS
 - ✓ Supports most popular databases



10

JBoss Portal - Features

- Membership System with Groups & Permissions
- Content Management System
- Portal with Portlets Integration



11

JBoss Portal - Portlets

- Out of the box
 - ✓ CMS Portlet
 - ✓ CMS Admin Portlet
 - ✓ User Portlet
 - ✓ Forum Portlet...etc
- Add-on
 - ✓ Weather
 - ✓ News Feed
 - ✓ Many more...refer Portlet Swap
- Highly customizable and extensible



12

Motivation

- Started in June 2005
 - ✓ when one of our existing jBPM customers required J2EE based open source CMS and Portal as an integrated solution
- Discussed with Leaders
 - ✓ Tom Baeyens (JBoss jBPM)
 - ✓ Roy Russo, Julien Viet (JBoss Portal)
 - ✓ CIGNEX Engineering Team
- Evaluated options to integrate jBPM with JBoss Portal



13

Motivation - JBoss CMS

- Missing: powerful, flexible workflow engine for complex document and content management
- **There is a Need for multi-party approval and collaboration business process**
 - ✓ Review/approve/publish process of documents by different people in an organization



14

Motivation - Workflow Portlets

- JBoss portal currently include portlets for Forums, Content Management, User Management etc

There is need for workflow portlets that provides front end to the workflow functionality within portal



15

Motivation - jBPM

- JBPM, a very viable solution as workflow engine within the JBoss Portal
 - ✓ Both as part of JEMS can share common underlying resources
 - ✓ Better User Management and Security
 - ✓ JBoss Portal can become front-end web application user interface for jBPM
 - ✓ Tighter integration between Portal and jBPM provides more value to end customers



16

Design Decisions

- Make JBPM a portlet inside the portal
 - ✓ Identity Module
 - Fetch Users & Roles from Portal Data Store
 - ✓ Process Portlet
 - List of existing Processes
 - Deploy New Process
 - ✓ Task Portlet
 - List of tasks associated with logged in user
 - Work on a task item
- JBPM Portlet integration with CMS portlet



17

Design - JSF UI Components

- The UI for the Portlets and associated Webapp / business process is developed using JSF.
- JSF framework Advantages
 - ✓ Follows the MVC framework
 - ✓ Rich and Reusable component model
 - ✓ Event driven model
 - ✓ Aids rapid dynamic UI interface development
 - ✓ Removes JSP binding
 - ✓ Use components developed by third-party developers



18

Design – jBPM JSF UI Components

- Developing jBPM JSF UI components allows for the components to be re-used in both a plain Webapp and a portal environment
- The following are the JSF components envisioned
 - ✓ Task List Component
 - ✓ Task Form Component
 - ✓ Process Start Component

19



Status – Identity Module

- The user assignments differ in jBPM and portal. So, we need to change the identity component of jBPM to use the portal user base
 - ✓ The identity module of jBPM is changed to define a new expression language which consisted of only "user" and "role" as identities
 - ✓ The user and roles are fetched from the portal data store
 - ✓ Any further custom assignment is achieved using delegation

20



Status – Identity Module Session

- Sharing of session between portal and portlet
 - ✓ Fetching of user roles needed using the existing session of portal
 - ✓ JBPM maintains its own session for jBPM specific data
- To achieve this changed hibernate settings in
 - ✓ Jboss-service.xml
- ✓ Hibernate.cfg.xml

```
<mbean code="org.jboss.portal.core.hibernate.SessionFactoryBinder"
name="jbpmporletcfg:service=Hibernate"
mbean-id="org.jboss.portal.core.hibernate.SessionFactoryBinder.xml">
<depends> jboss:jca:service=DataSourceBinding,name=PortalDS </depends>
```

```
<hibernate-configuration>
<session-factory name="java:/jbpmpSessionFactory">
```

21



Status - Portlets developed

- Process Portlet
 - ✓ The user will be able to see the list of processes available
 - ✓ The user can deploy a new process
- Task Management Portlet
 - ✓ When the user logs in, he will be able to see the list of tasks associated with him
 - ✓ The user should be able to select a particular task

22



Status - Process Portlet

- Deploying a new process

```
// creating JBPM session Factory
jbpmSessionFactory = JbpmSessionFactory.buildJbpmSessionFactory();

ProcessDefinition newProcessDefinition =
ProcessDefinition.parseXmlResource("WEBINF/processdefinition.xml");

// Let's open a new persistence session
JbpmSession jbpmSession = JbpmSessionFactory.openJbpmSession();

// ... and begin a transaction on the persistence session
jbpmSession.beginTransaction();

// Save the process definition in the database
jbpmSession.getGraphSession().saveProcessDefinition(newProcessDefinition);
```
- List of all Process Definitions

```
JbpmSession jbpmSession = Context.getPersistenceContext().getJbpmSession();
GraphSession graphSession = jbpmSession.getGraphSession();
List allProcessDefinitions = graphSession.findAllProcessDefinitions();
```

23



Status - Task Portlet

- To find task for the individual user

```
JbpmSessionFactory jbpmSessionFactory =
JbpmSessionFactory.buildJbpmSessionFactory();
JbpmSession jbpmSession = JbpmSessionFactory.openJbpmSession();
jbpmSession.beginTransaction();
TaskMgmtSession taskSession = jbpmSession.getTaskMgmtSession();
List tasks = taskSession.findTaskInstances(request.getRemoteUser());
Iterator iterator = tasks.iterator();
```
- To get the roles in the system

```
// Role module gives the list of roles available in the system
RoleModule roleModule = (RoleModule) new
InitialContext().lookup(ModuleConstants.ROLEMODULE_JNDI_NAME);
Set roles = roleModule.findRoles();
Iterator roleIt = roles.iterator();
```

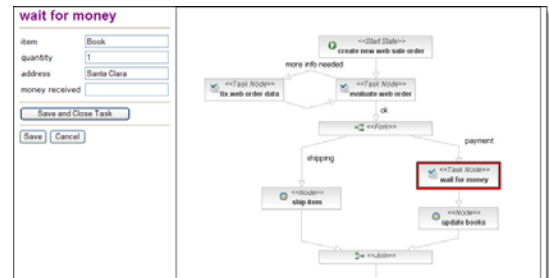
24



Status – Task Portlet for user “Bert”

25

Status – Task edit page for “Bert”



26

Next Steps

- Now that we have the necessary portlets and session integration between portal and portlet, the next goal is provide workflow support to portal CMS
- Requirements
 - ✓ Flexible association of workflow templates to the content management
 - ✓ Proper access and security management
 - ✓ Publish and expire capabilities
 - ✓ External Workflow editors

27

Use Case: Portlets to CMS integration

- The administrator deploys the process definition file for content management
- The CMS uses that process definition to start the execution of the process. The roles, actors are identified for various tasks
- The admin adds the necessary roles, if needed as specified in the process definition
- Once the document is created using the tasks portlet, the workflow kicks in
- The necessary actors are invoked using the tasks portlet to perform the necessary approve/publish process
- If after a preset time, no action takes place on the document, the default timer makes either document invalid or escalates for next appropriate action

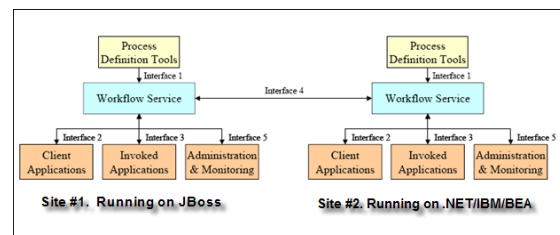
28

Next Steps - Planned

- jBPM Portlet related
 - ✓ Port the completed work to the newer version of jBPM.
 - ✓ Develop user interface components for
 - Task List
 - Task Form
 - Process List
- CMS Portlet integration with jBPM portlet
 - ✓ Implementation details to be determined after the jBPM portlet is completed.

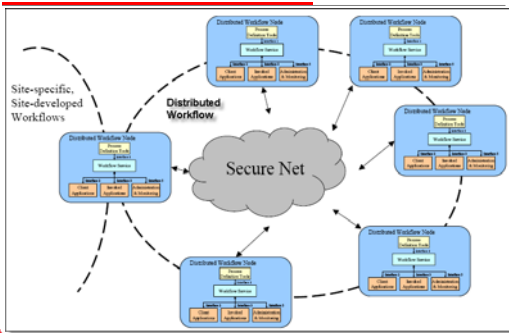
29

Future – Workflow Services



30

Future – Distributed Workflow



31

JBoss World
LAS VEGAS

Summary

- Effectively implemented workflow and CMS solution offers a variety of benefits to an organization
 - ✓ reduced operating costs
 - ✓ improved productivity and services
- JBoss jBPM integration provides much value-added business automation features to the JBoss Portal
 - ✓ Fills the gap in JEMS stack by providing a CMS with powerful and flexible workflow engine

32

JBoss World
LAS VEGAS

Q & A

Contact:

Munwar Shariff

munwar@cignex.com

408.327.9900 x303

33

JBoss World
LAS VEGAS