



jPDL: Simplified Workflow for Java™ Technology

Tom Baeyens

Founder and Lead of JBoss jBPM
JBoss, a Division of Red Hat
<http://jbpm.org>

Session TS-8612



Tom Baeyens

- Founder and Lead of JBoss jBPM
- JBoss, a division of Red Hat
- Participate in Java Community ProcessSM (JCPSM)
- Mission
 - Unify workflow, BPM and orchestration
 - Bring these technologies to developer community



Workflow Business Process Management (BPM) Orchestration

...it all boils down to state machines



Agenda

Introduction

Door in Java technology and jPDL

Human tasks

Binding process to code

Use cases

Persistence



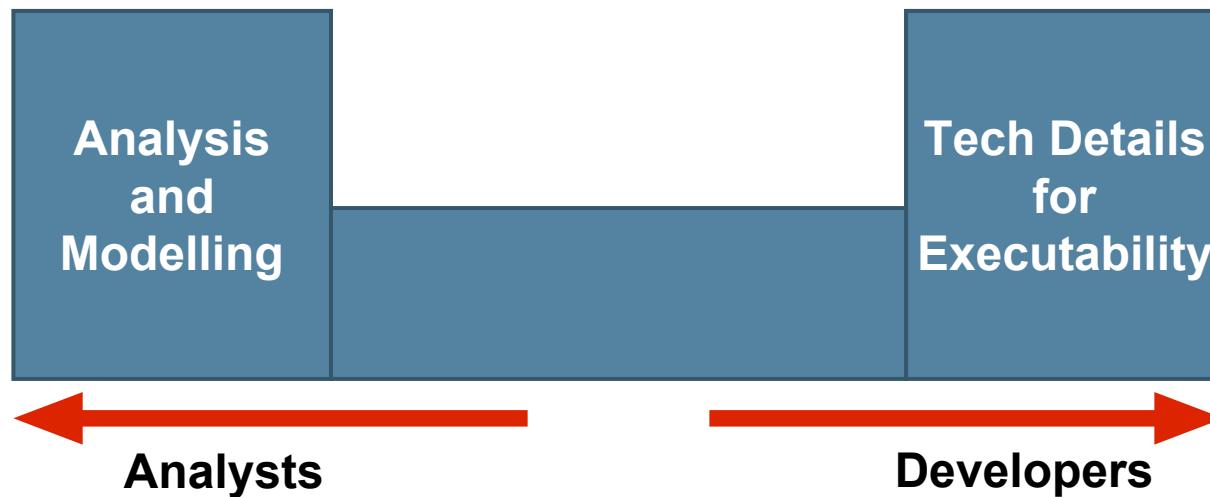
What Is a Business Process?

- Description of how people and/or systems work together
- Typical examples
 - Insurance claim, approvals, legal case



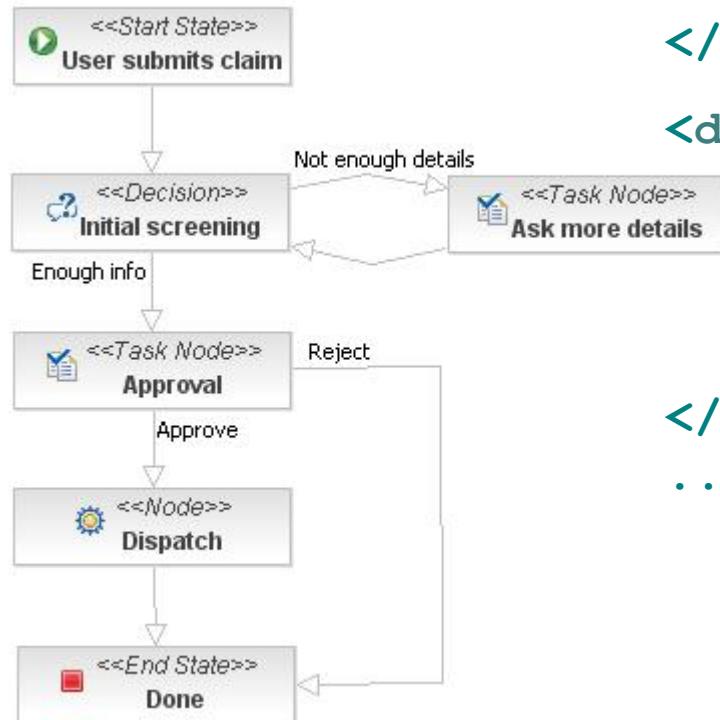
JavaOne

What Is Business Process Management?





What Is a Process Language?



```
<process-definition name="Damage claim">

    <start-state name="User submits claim">
        <transition to="Initial screening"/>
    </start-state>

    <decision name="Initial screening">
        <transition name="Enough info"
                  to="Approval"/>
        <transition name="Not enough details"
                  to="Ask more details"/>
    </decision>
    ...

```



Java

JavaOne

What Is an Executable Process Language?

- Executable process
 - Describe **execution** flow
 - **Graphical** activities
 - Activities can be **wait states**
 - Executed by **process engine**



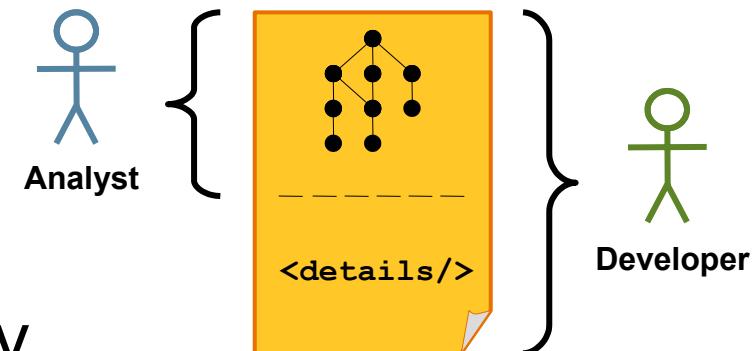


Java

One

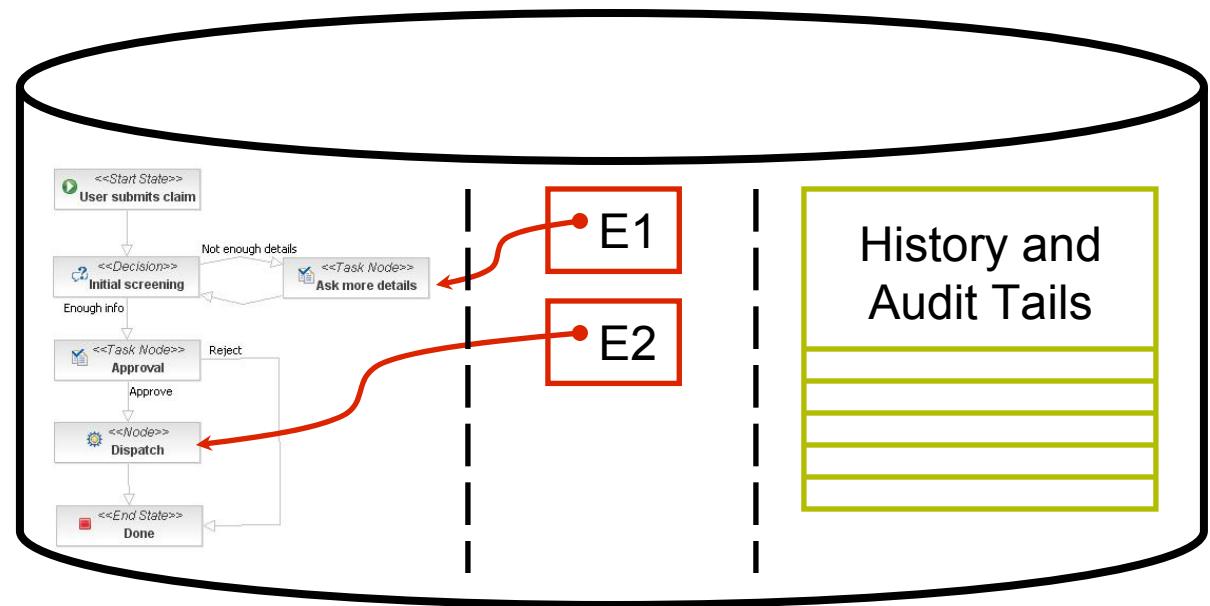
Why an Executable Process Language?

- Simplify implementation
 - Executable process means implementation
 - Some aspects easier as workflow process
 - It extracts state management
- Improve communication
- Automatic persistent history
 - Business intelligence



What Is a BPMS?

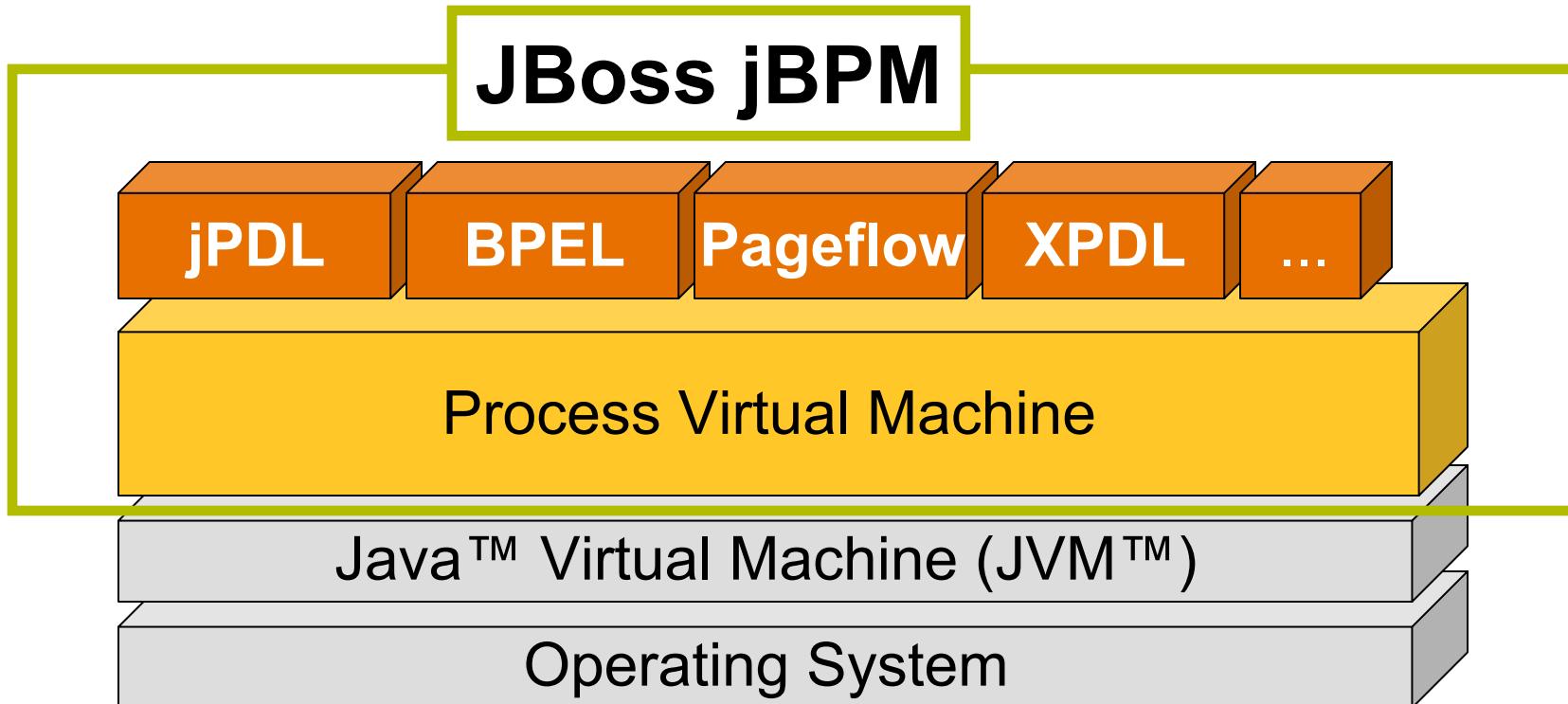
- Business Process Management System
- Executes processes
- Persists executions
- History





JBoss jBPM

- One embeddable technology
- Multiple process languages



The terms “Java Virtual Machine” and “JVM” mean a Virtual Machine for the Java™ platform.



Agenda

Introduction

Door in Java technology and jPDL

Human tasks

Binding process to code

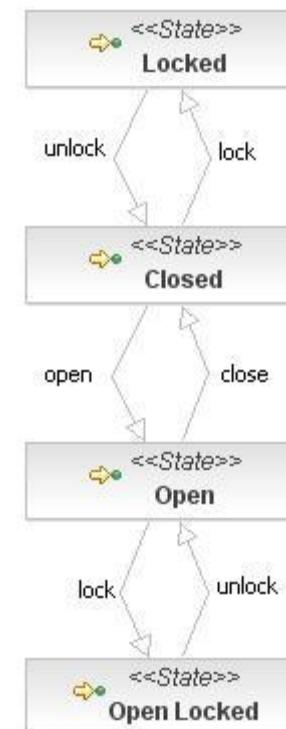
Use cases

Persistence



A First Example

- A door
- 4 states
 - Locked, Closed, Open, Open Locked
- 4 operations
 - unlock, lock
open, close
- First Java code
- Then jPDL





A Door in Java Technology

```
public class Door {  
    static String OPEN = "open";  
    static String CLOSED = "closed";  
    static String LOCKED = "locked";  
    static String OPEN_LOCKED = "open-locked";  
  
    String state = CLOSED;  
}
```

...



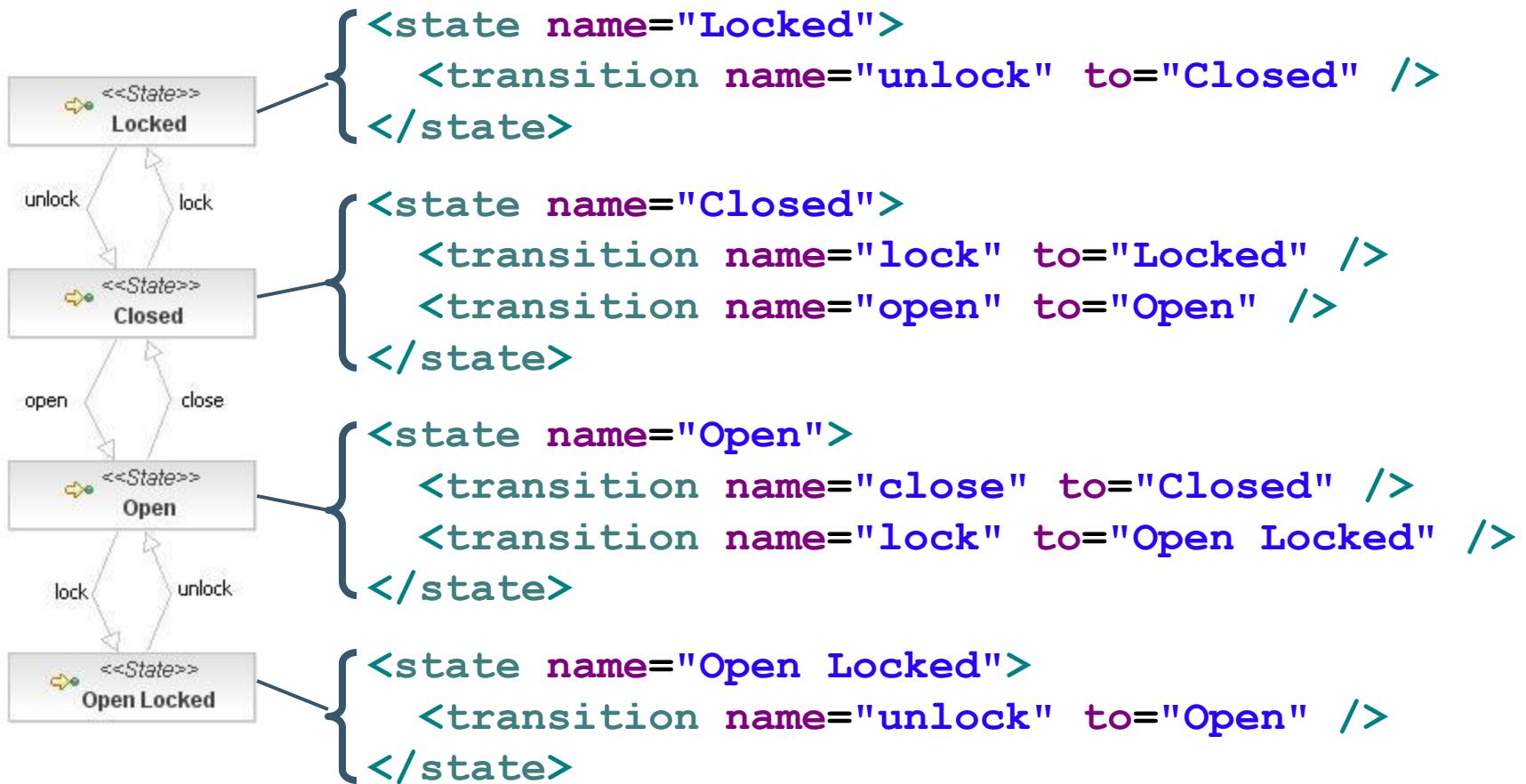
A Door in Java Technology

```
public void lock() {  
    if ( (state==LOCKED) || (state==OPEN_LOCKED) ) {  
        throw new IllegalStateException(  
            "door is already locked"  
        );  
    }  
  
    if (state==CLOSED) {  
        state = LOCKED;  
    } else if (state==OPEN) {  
        state = OPEN_LOCKED;  
    }  
}
```



A Door in jPDL

```
<process-definition name="door" initial="Closed">
```



```
</process-definition>
```



A Door in jPDL

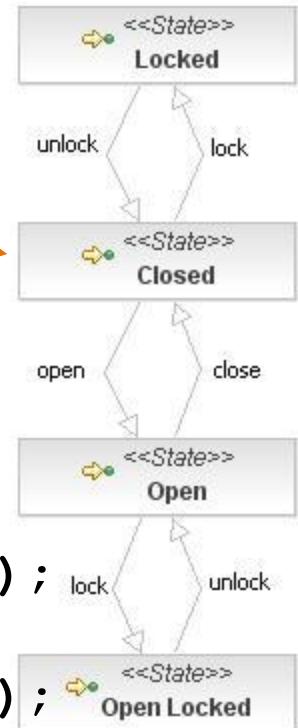
```
public class DoorProcessTest extends TestCase {  
    static ProcessDefinition doorProcess =  
        ProcessDefinition.parseXmlInputStream(  
            DoorProcessTest.class.  
                getResourceAsStream("processdefinition.xml"))  
};  
...
```



A Door in jPDL

```
public void testClosedLock() {  
    ProcessInstance processInstance =  
        new ProcessInstance(doorProcess);  
    Token token = processInstance.getRootToken();  
  
    token.signal("lock");  
    assertEquals("Locked", token.getNode().getName());  
  
    token.signal("unlock");  
    assertEquals("Closed", token.getNode().getName());  
  
    try {  
        token.signal("unlock");  
        fail();  
    } catch (JbpmException e) {  
    }  
}
```

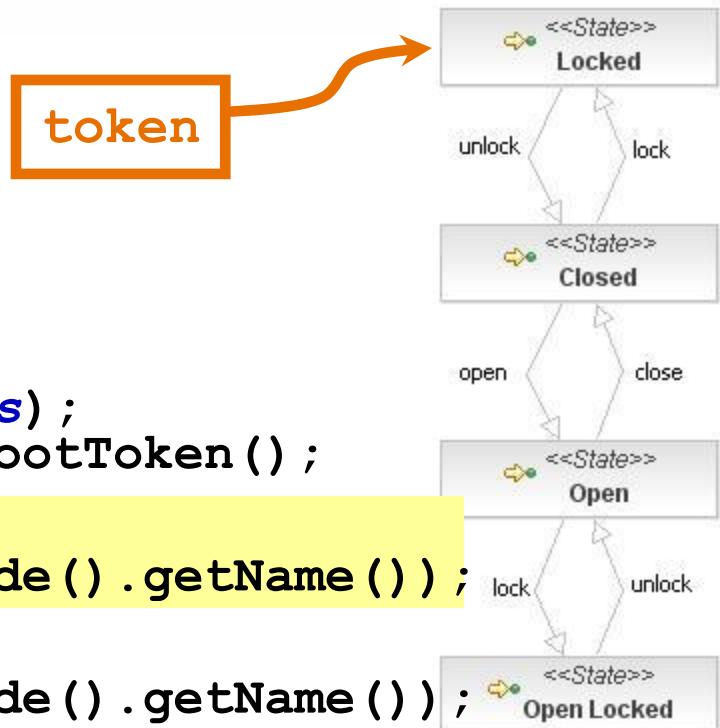
token





A Door in jPDL

```
public void testClosedLock() {  
    ProcessInstance processInstance =  
        new ProcessInstance(doorProcess);  
    Token token = processInstance.getRootToken();  
  
    token.signal("lock");  
    assertEquals("Locked", token.getNode().getName());  
  
    token.signal("unlock");  
    assertEquals("Closed", token.getNode().getName());  
  
    try {  
        token.signal("unlock");  
        fail();  
    } catch (JbpmException e) {  
    }  
}
```

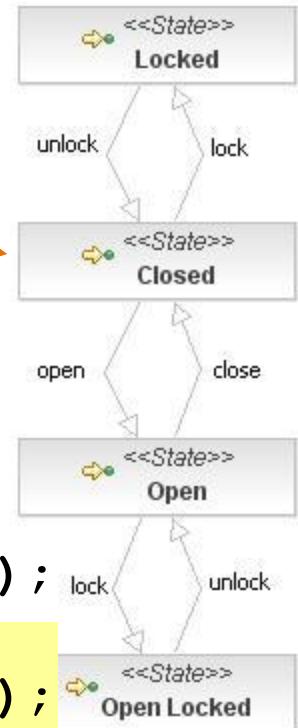




A Door in jPDL

```
public void testClosedLock() {  
  
    ProcessInstance processInstance =  
        new ProcessInstance(doorProcess);  
    Token token = processInstance.getRootToken();  
  
    token.signal("lock");  
    assertEquals("Locked", token.getNode().getName());  
  
    token.signal("unlock");  
    assertEquals("Closed", token.getNode().getName());  
  
    try {  
        token.signal("unlock");  
        fail();  
    } catch (JbpmException e) {  
    }  
}
```

token





A Door in jPDL

```
public void testClosedLock() {  
    ProcessInstance processInstance =  
        new ProcessInstance(doorProcess);  
    Token token = processInstance.getRootToken();  
  
    token.signal("lock");  
    assertEquals("Locked", token.getNode().getName());  
  
    token.signal("unlock");  
    assertEquals("Closed", token.getNode().getName());  
  
    try {  
        token.signal("unlock");  
        fail();  
    } catch (JbpmException e) {  
    }  
}
```

token





Agenda

Introduction

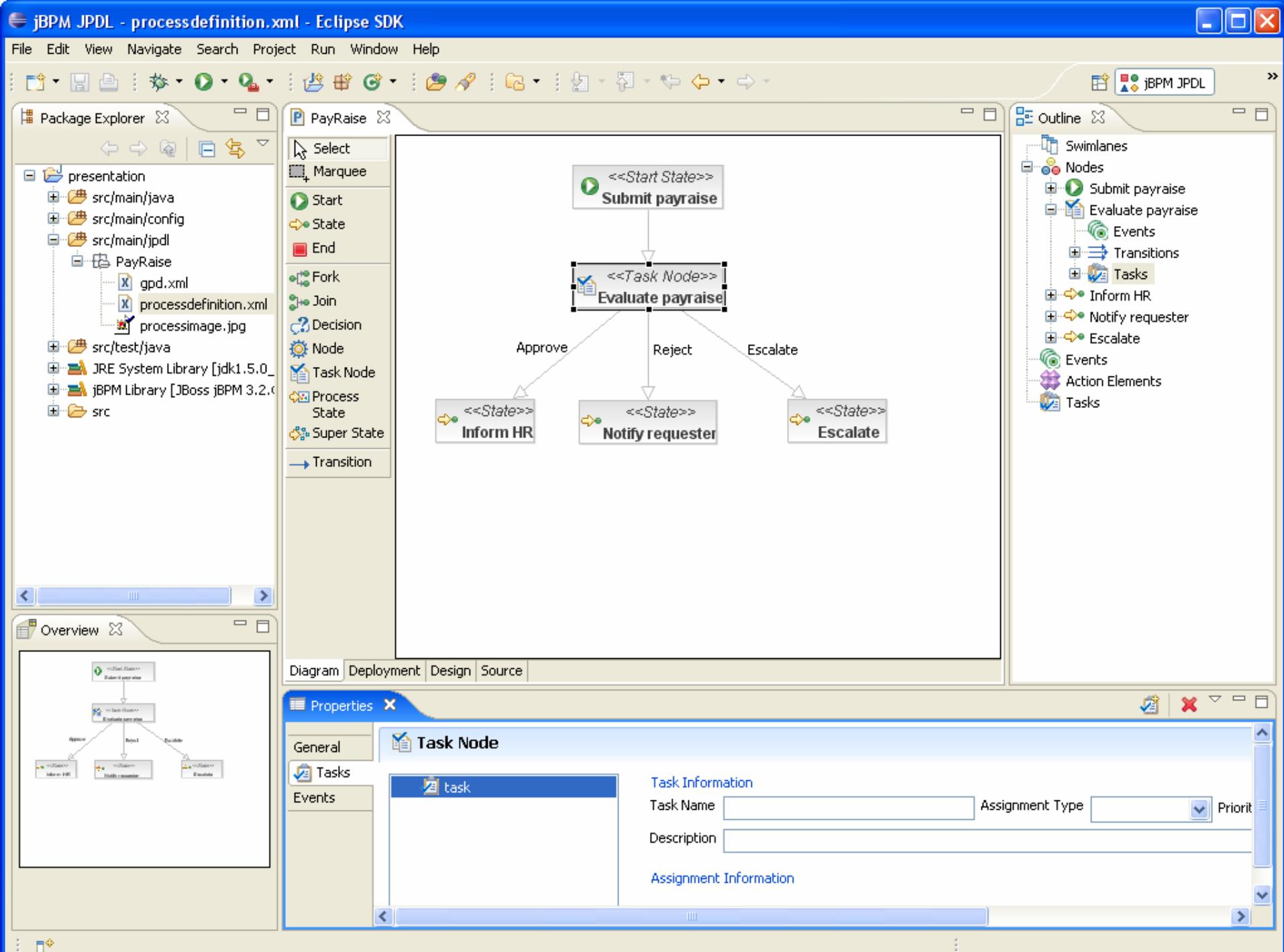
Door in Java technology and jPDL

Human tasks

Binding process to code

Use cases

Persistence



jBPM JPD - processdefinition.xml - Eclipse SDK

File Edit View Navigate Search Project Run Window Help

Package Explorer X

presentation
src/main/java
src/main/config
src/main/jpd
src/test/java
JRE System Library [jdk1.5.0_22]
JBPM Library [JBoss jBPM 3.2.0.GA]
src

*PayRaise X

```
<?xml version="1.0" encoding="UTF-8"?>

<process-definition xmlns="urn:jbpm.org:jpdl-3.2" name="PayRaise">

    <start-state name="Submit payraise">
        <transition to="Evaluate payraise" />
    </start-state>

    <task-node name="Evaluate payraise">
        <task notify="yes" priority="high">
            <>description
            <>event
            <>exception-handler
            <>task
            <>timer
            <>transition
            <!-- comment - xml comment
            <!-- XSL processing instruction - XSL processing instruction
            <>cdata-node>
        </task>
    </task-node>

    <state name="Inform HR" />
    <state name="Notify requester" />
    <state name="Escalate" />
```

Outline X

- Swimlanes
- Nodes
 - Submit payraise
 - Evaluate payraise
 - Events
 - Transitions
 - Tasks
 - Inform HR
 - Notify requester
 - Escalate

Element : event
Content Model : (((action | script | create-timer | cancel-timer | mail)))*

Diagram Deployment Design Source

Properties X

Properties are not available.

```
graph TD
    Start((Start)) --> Evaluate[Evaluate payraise]
    Evaluate --> InformHR[Inform HR]
    Evaluate --> Notify[Notify requester]
    Evaluate --> Escalate[Escalate]
```



A Task in jPDL

```
<task-node name="Evaluate pay raise">
    <task notify="yes" priority="high">
        <assignment class="com.sample.Assigner" />
        <reminder duedate="2 business days"
                   repeat="4 business hours" />
        <timer duedate="4 business days"
                transition="Escalate" />
    </task>
    <transition name="Approve" to="Inform HR" />
    <transition name="Reject" to="Notify requester" />
    <transition name="Escalate" to="Escalate request" />
</task-node>
```

Search For...

Tasks

Processes

Process Instances

Manager Menu

Deploy Process

You are logged in as:
"ernie"

Log Out

Task Instance View

Summary

Task Form Source

Diagram

Evaluate payraise

Requester	Ernie
Amount	\$500
Reason	Learned jBPM
Actions	<input type="button" value="Save"/> <input type="button" value="Cancel"/> <input type="button" value="Reject"/> <input type="button" value="Accept"/> <input type="button" value="Escalate"/>

Task Instance Summary

Instance ID	6	Description	
Current Actor	ernie	Task Created Date	19-mrt-2007 15:22:39
Pooled Actors		Task Start Date	19-mrt-2007 15:22:41
Task Status	Open	Task End Date	
Task Priority	3	Task Due Date	
Process Instance ID	3 (View Instance)		

Actions

Reassign to

cookie monster ▾

Add a comment



Agenda

Introduction

Door in Java technology and jPDL

Human tasks

Binding process to code

Use cases

Persistence



JavaOne

jPDL Is Open Ended

- Traditional BPM/workflow systems
 - **Fixed** process language
- jPDL has process constructs
 - **Base** process language, that can be extended
 - API for runtime behaviour
- jPDL can easily include Java code
 - Process == structure
 - Code as decoration





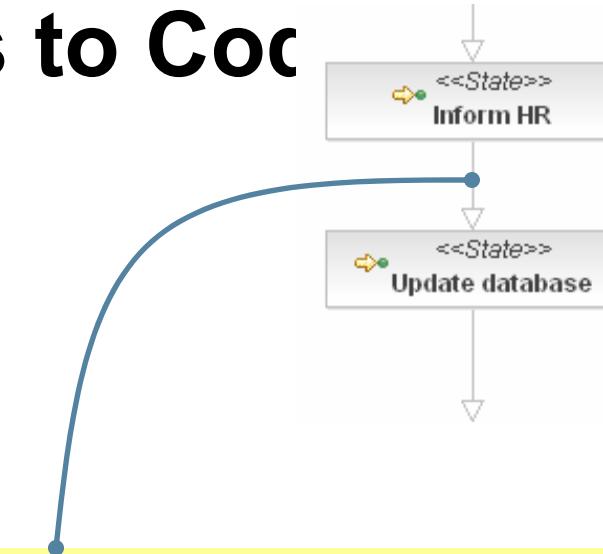
jPDL Adds Features to Java Technology

- Use jPDL for state management
- Use the Java platform as you do now
- Natural binding



Binding Process to Code

- e.g., Action
 - On transition
 - Calculate number



```
public interface ActionHandler extends Serializable {  
    void execute( ExecutionContext ctx )  
        throws Exception;  
}
```



Binding Process to Code

```
<state name="Inform HR">  
  <transition to="Update database">  
    <action class="payraise.CalculateNumber">  
      <var>Salary Increase</var>  
      <factor>5</factor>  
    </action>  
  </transition>  
</state>
```



Binding Process to Code

```
public class CalculateNumber
    implements ActionHandler {
    String var;
    int factor;
    public void execute(ExecutionContext ctx) {
        Integer value = (Integer) ctx.getVariable(var);
        int number = factor * value.intValue();
        ctx.setVariable("number", number);
    }
}
```



Expression Language

```
<state name="Inform HR">  
    <transition to="Update database">  
        <action expression="#{myService.calculateNumber}" />  
    </transition>  
</state>
```

- Variable resolver knows all jBPM objects
 - Process variables
 - Swimlanes (=process roles)
 - Configuration beans



Expression Language

- JBoss Seam/Web beans integration adds:
 - Enterprise JavaBeans™ (EJB™) component architectures
 - Backed beans
 - Conversation beans
 - Any named object in any Seam context



Agenda

Introduction

Door in Java technology and jPDL

Human tasks

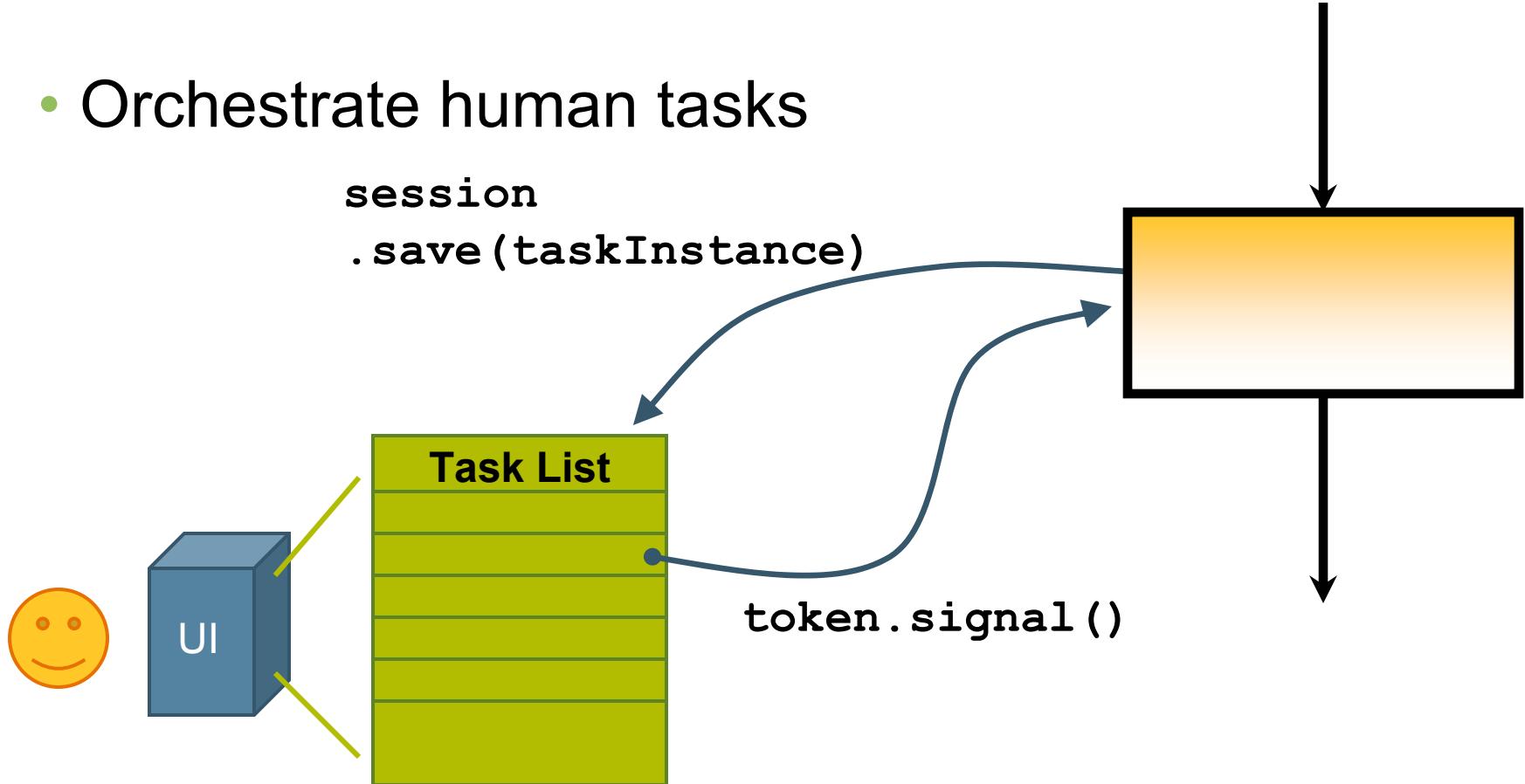
Binding process to code

Use cases

Persistence

jPDL Use Cases

- Orchestrate human tasks



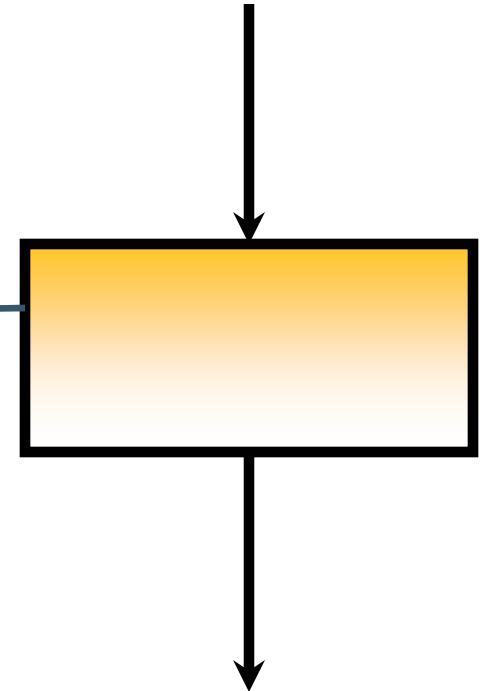


jPDL Use Cases

- Sending async message

`messageProducer`

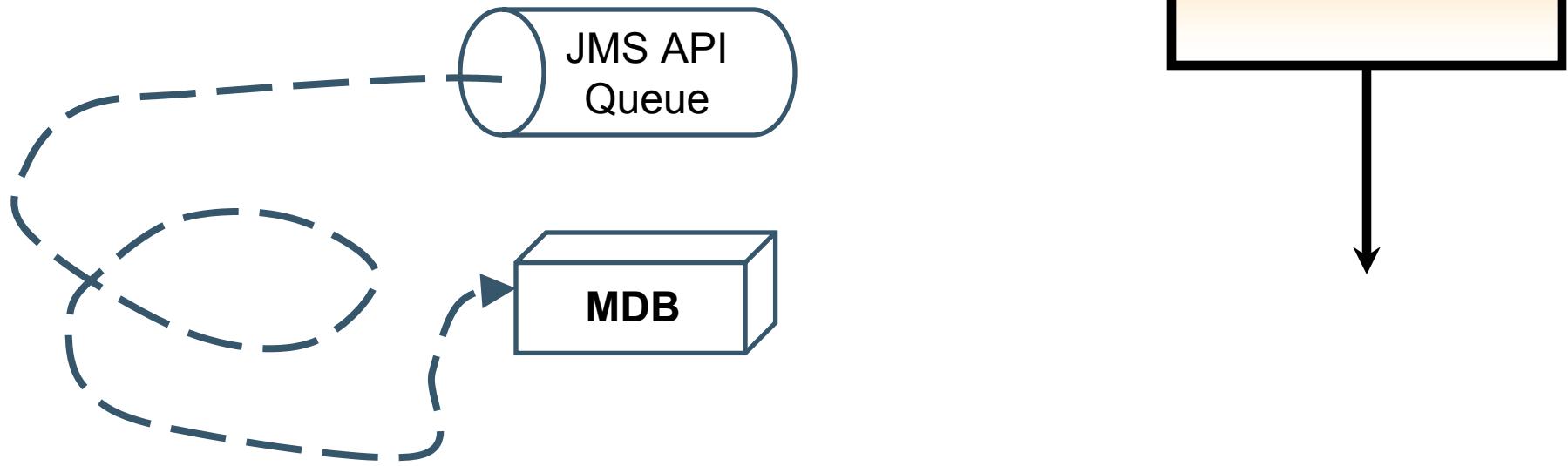
```
. send(destination, message)
```





jPDL Use Cases

- Sending async message





Agenda

Introduction

Door in Java technology and jPDL

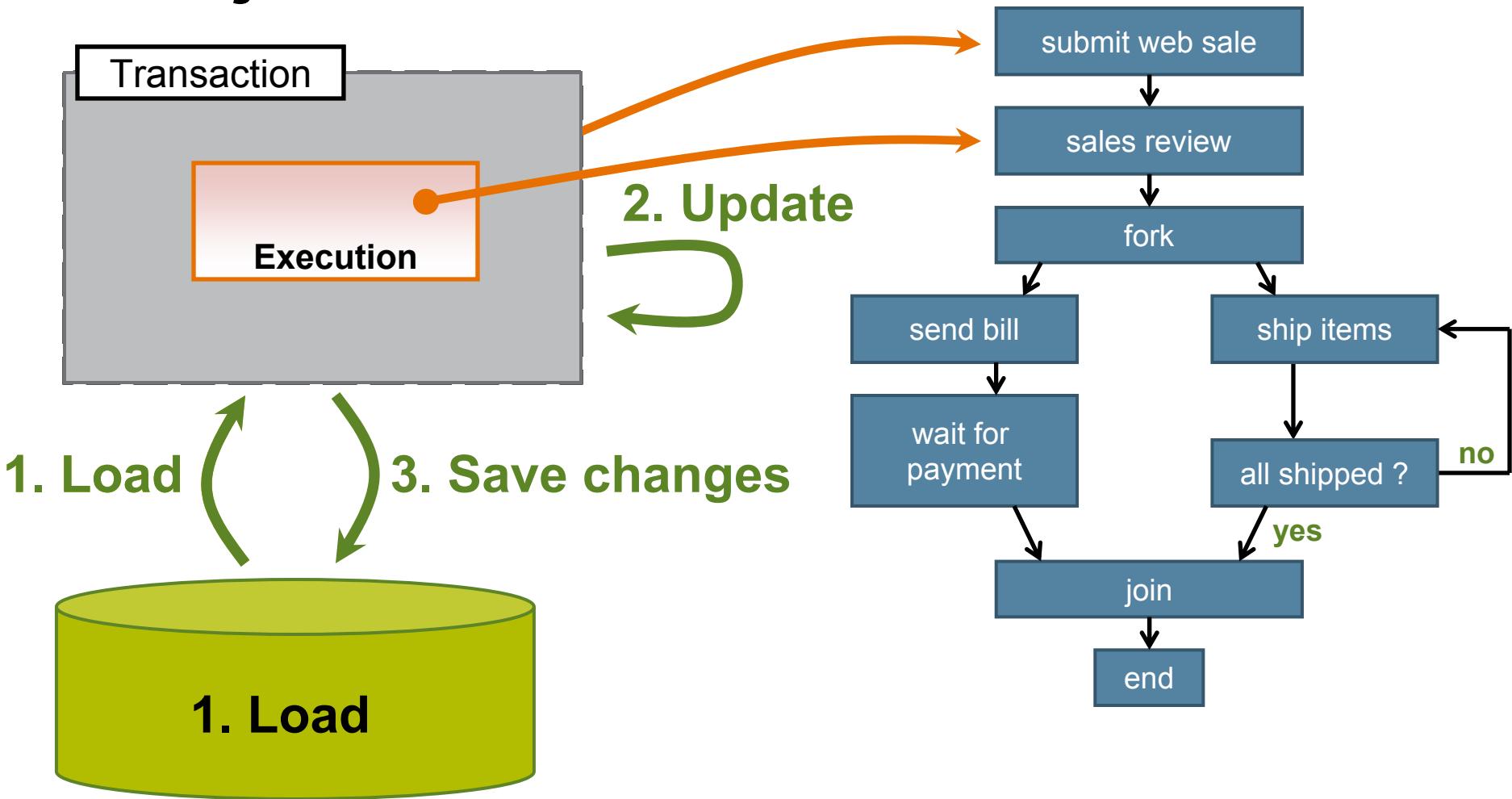
Human tasks

Binding process to code

Use cases

Persistence

jPDL Persistence





jPDL Features

- **Simple**
 - Use only what you need
- **Powerful**
 - Superior modelling and execution
- **Embeddable**
 - Libraries
 - Database
 - Standard and Enterprise Java platform



Conclusion

- JBoss jBPM is a platform for process languages
- jPDL
 - Process language
 - Fits with Java development
 - Extracts state management
 - Improves communication
- Embeddable
 - Standard and Enterprise



jPDL Compared to BPEL

- jPDL
 - Interface and variables are Java code
 - Easy to leverage Java technology
 - Task management
 - Powerful and extensible constructs
- BPEL
 - Interface and variables based on XML
 - Easy to leverage web services
 - Write a new web service as a function of others



Q&A

JBoss jBPM: <http://jbpm.org>

The Process Virtual Machine: <http://jbpm.org/pvm>

JBoss SEAM: <http://www.jboss.com/products/seam>



jPDL Scalability

- Hibernate optimistic locking
 - Version column
 - **UPDATE . . .
SET . . . , version = 2
WHERE id = 5
AND version = 1**
- Synchronization by DB
- Also pessimistic locking possible
- One set of objects per transaction



jPDL: Simplified Workflow for Java™ Technology

Tom Baeyens

Founder and Lead of JBoss jBPM
JBoss, a Division of Red Hat
<http://jbpm.org>

Session TS-8612