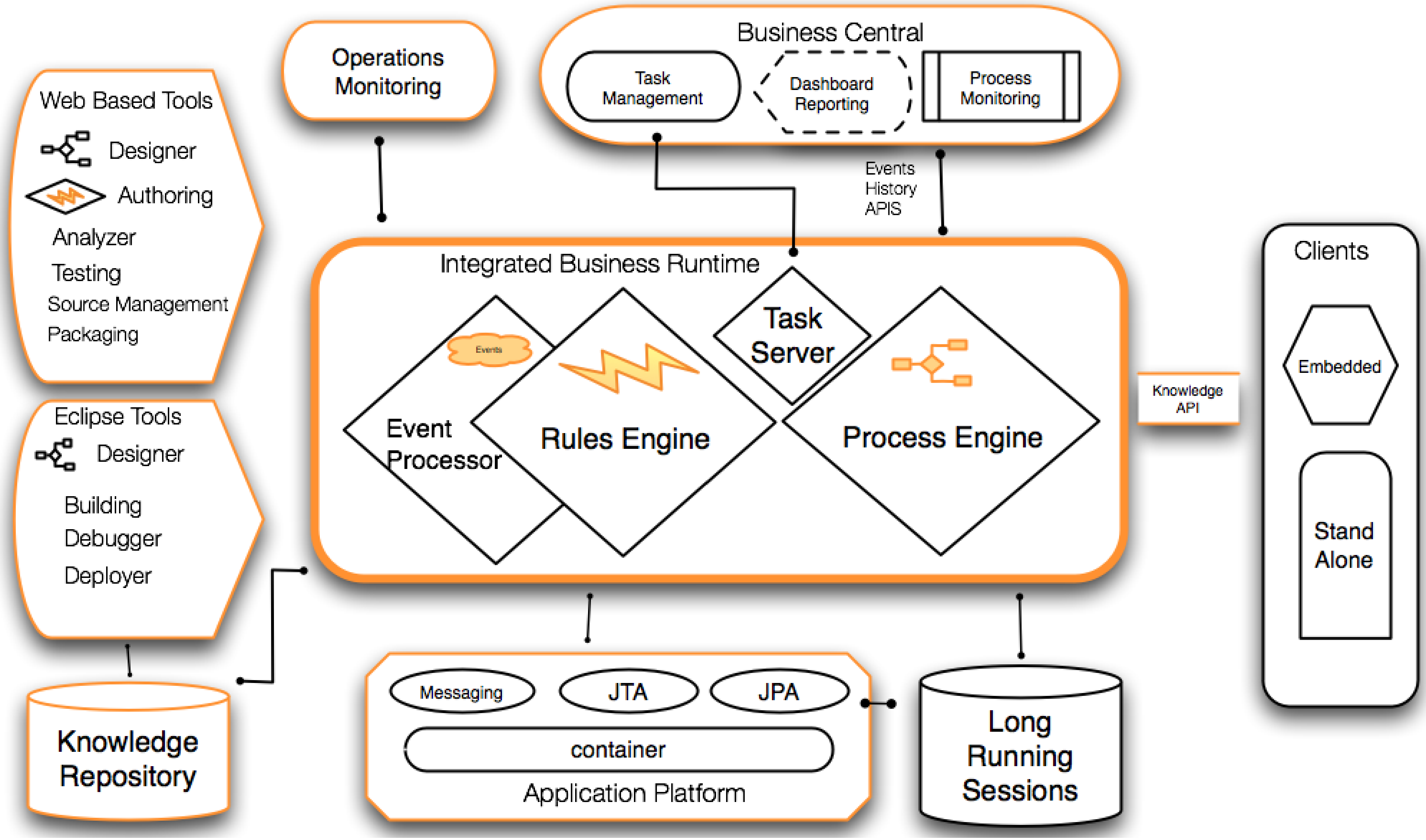
Product Details

Contents

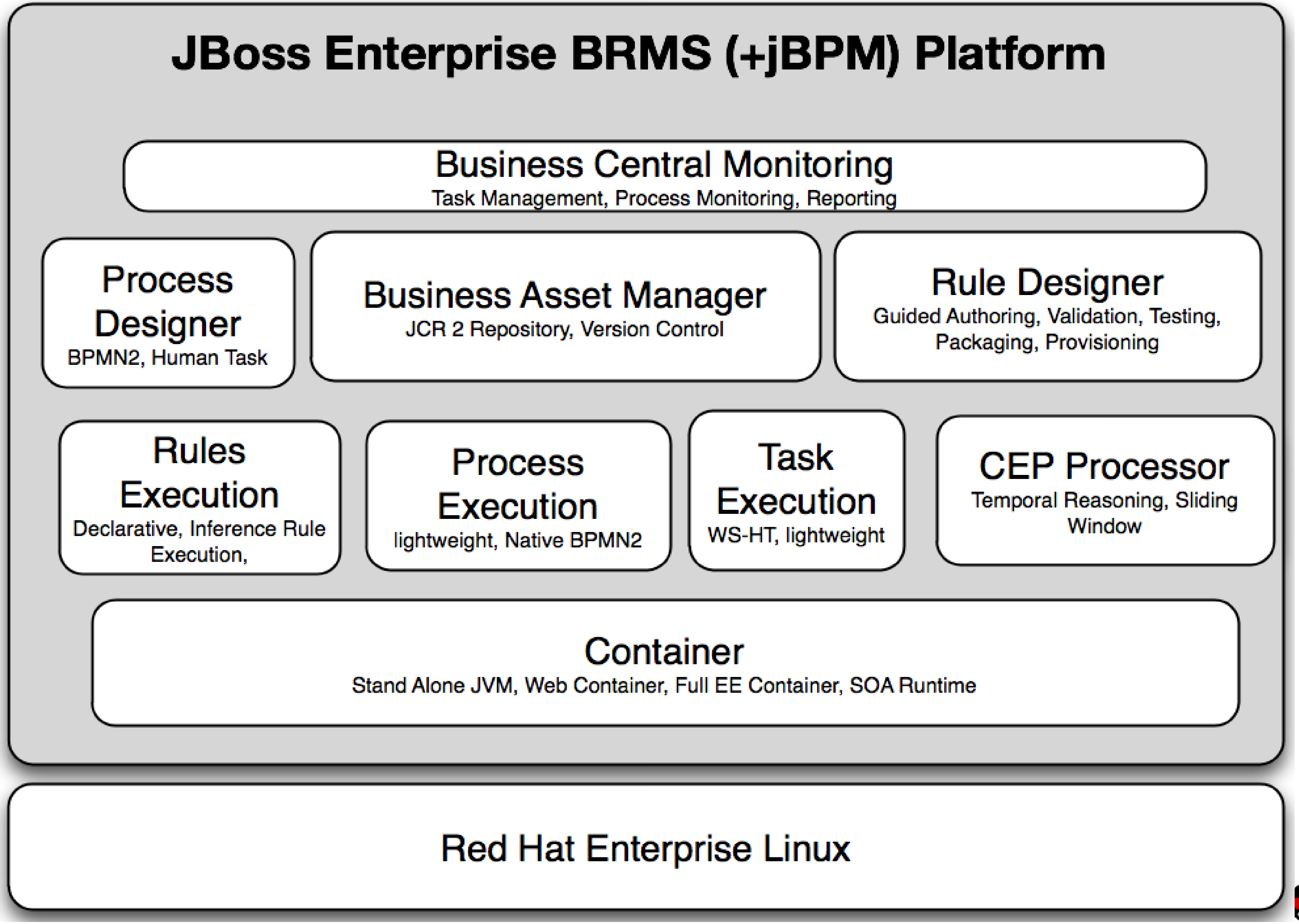
- Platform Features
- Business Process / Workflow Features
- Business Rules and management Features
- Business Events / CEP Features
- Everything else

Platform Features

Platform Architecture



Integrated Intelligent Business Platform



JBoss Enterprise BRMS (+jBPM) Platform

Business Central Monitoring
Task Management, Process Monitoring, Reporting

Process Designer
BPMN2, Human Task

Business Asset Manager
JCR 2 Repository, Version Control

Rule Designer
Guided Authoring, Validation, Testing, Packaging, Provisioning

Rules Execution
Declarative, Inference Rule Execution,

Process Execution
lightweight, Native BPMN2

Task Execution
WS-HT, lightweight

CEP Processor
Temporal Reasoning, Sliding Window

Container
Stand Alone JVM, Web Container, Full EE Container, SOA Runtime

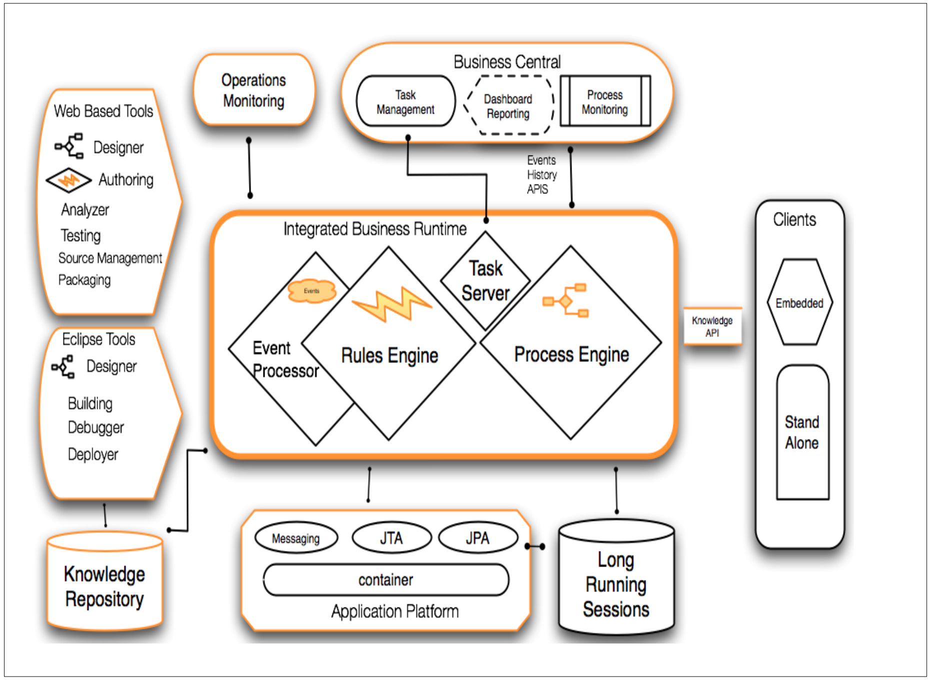
Red Hat Enterprise Linux

Integrated Intelligent Business Platform

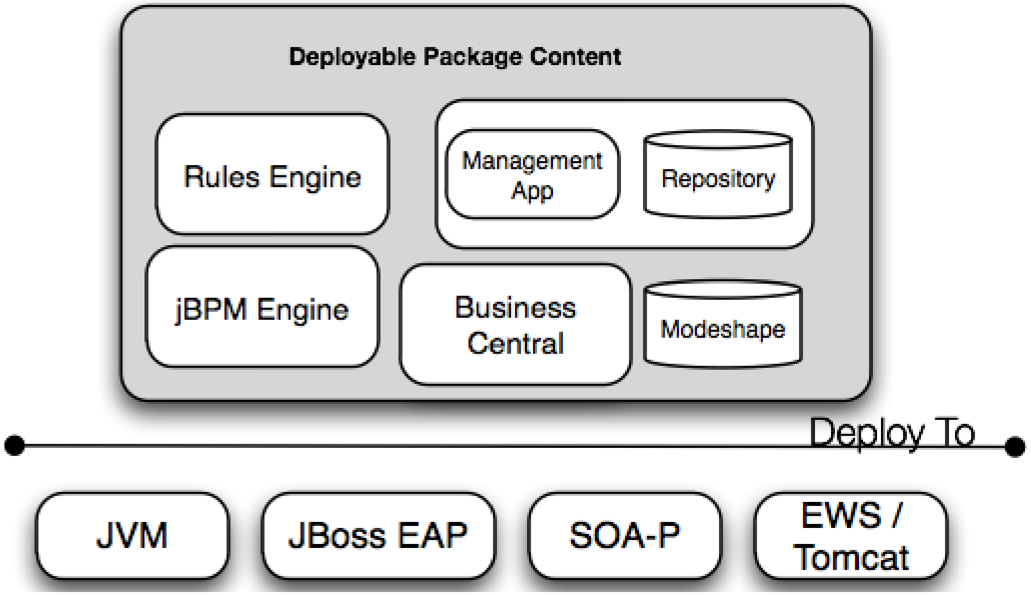
Under Construction

Usecase of using rules, events and process together

Choice of Package: Integrated Vs Components

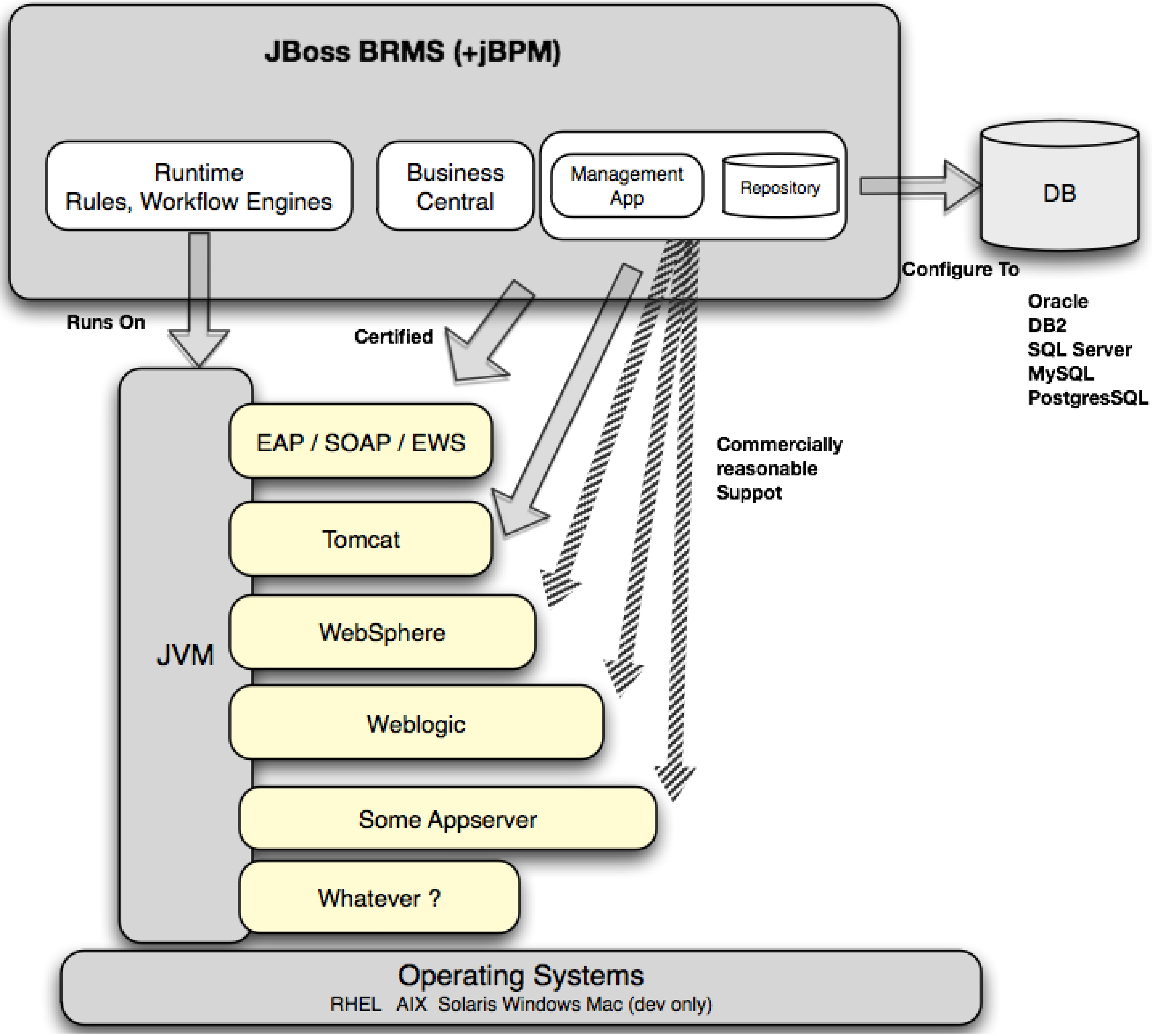


- Stand Alone Package
- Well integrated
- Best Out of the Box Experience



- Lightweight Package
- Flexibility of Container
- Deploy to lightweight containers like Tomcat

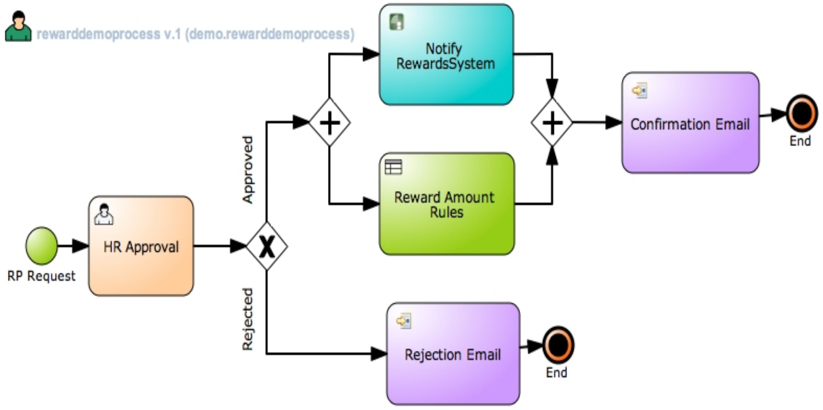
Product Support Configurations

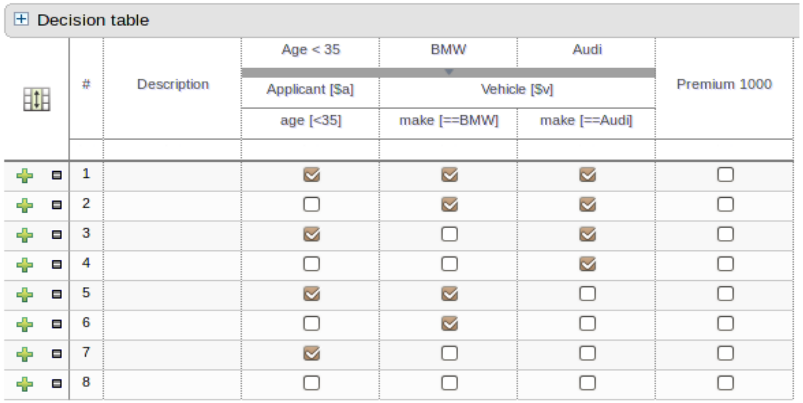


EAP 6 Support ?

Choice of Tools

Business User Friendly – Web Based

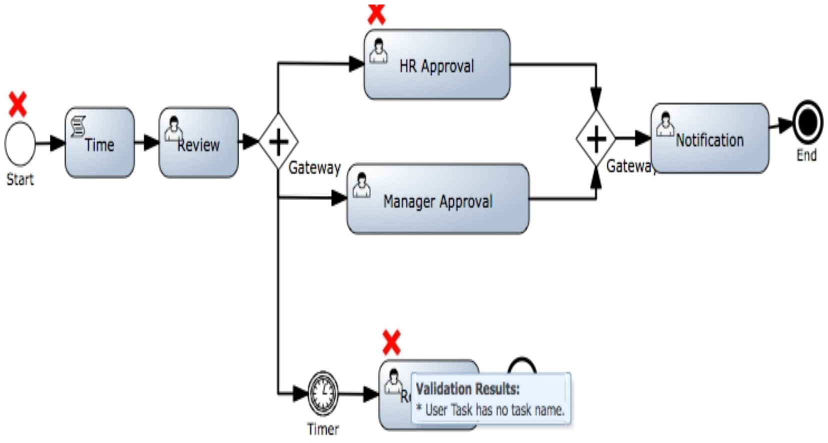


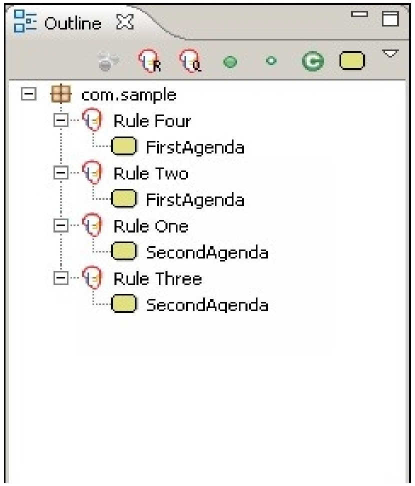


Decision table

	#	Description	Age < 35	BMW	Audi	Premium 1000
			Applicant [$a]	Vehicle [$v]		
			age [<35]	make [==BMW]	make [==Audi]	
	1		☑	☑	☑	□
	2		□	☑	☑	□
	3		☑	□	☑	□
	4		□	□	☑	□
	5		☑	☑	□	□
	6		□	☑	□	□
	7		☑	□	□	□
	8		□	□	□	□

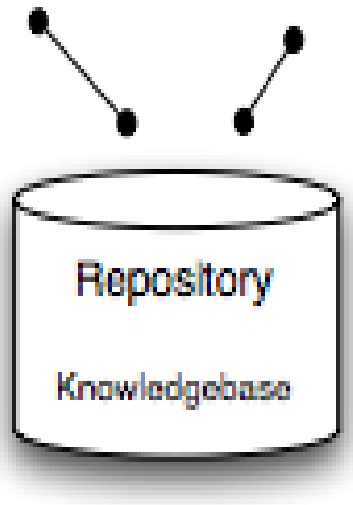
Developer Friendly – IDE Based





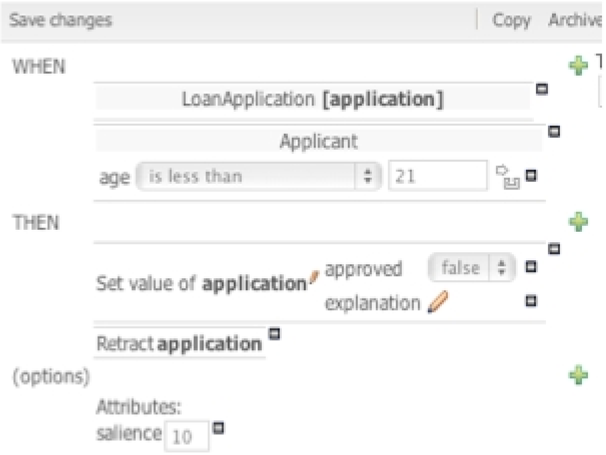
Outline

- com.sample
  - Rule Four
    - FirstAgenda
  - Rule Two
    - FirstAgenda
  - Rule One
    - SecondAgenda
  - Rule Three
    - SecondAgenda



Knowledge Management

Authoring



WHEN

LoanApplication **[application]**

Applicant

age | is less than | 21

THEN

Set value of **application** approved | false

explanation

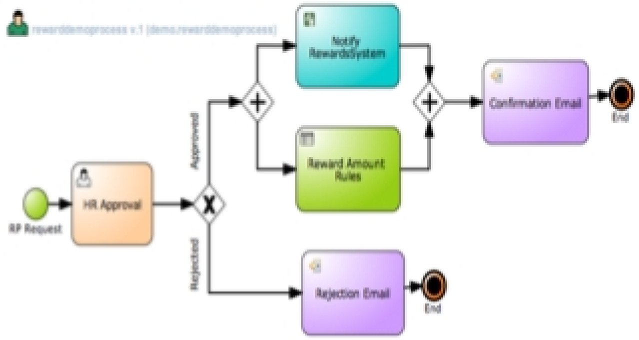
Retract **application**

(options)

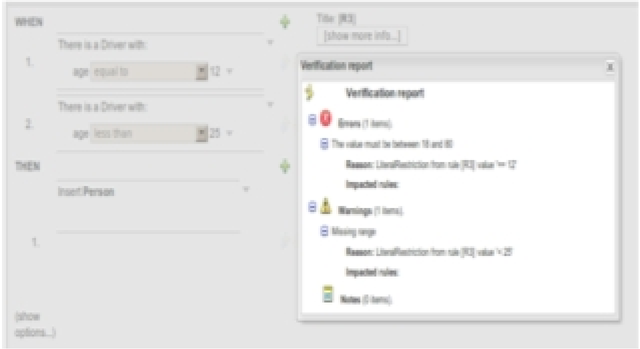
Attributes:

salience | 10

Modeling



Validation



WHEN

1. There is a Driver with:
   age | equal to | 12

2. There is a Driver with:
   age | less than | 25

THEN

Insert Person

1.

(show options...)

Title: [R3]

[show more info...]

Verification report

Verification report

Errors (1 items)

The value must be between 18 and 80

Reason: LiteralRestriction from rule [R3] value '== 12'

Impacted rules:

Warnings (1 items)

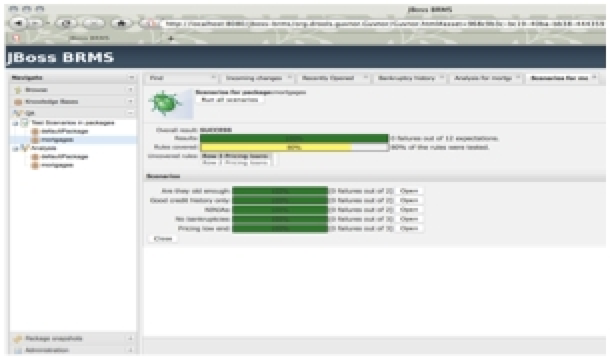
Missing range

Reason: LiteralRestriction from rule [R3] value '< 25'

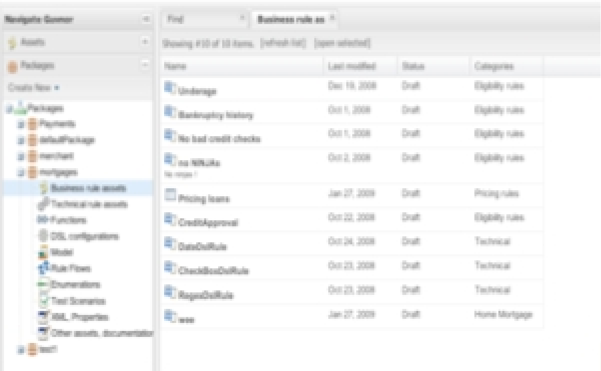
Impacted rules:

Notes (0 items)

Test Scenarios



Asset Management



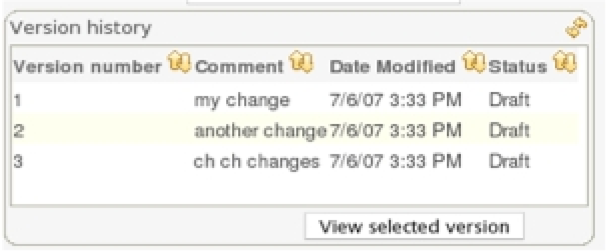
Showing #10 of 10 items. [refresh list] [open selected]

Name	Last modified	Status	Categories
Underage	Dec 19, 2008	Draft	Eligibility rules
Bankruptcy history	Oct 1, 2008	Draft	Eligibility rules
No bad credit checks	Oct 1, 2008	Draft	Eligibility rules
no NINJAs No ninjas !	Oct 2, 2008	Draft	Eligibility rules
Pricing loans	Jan 27, 2009	Draft	Pricing rules
CreditApproval	Oct 22, 2008	Draft	Eligibility rules
DateDslRule	Oct 24, 2008	Draft	Technical
CheckBoxDslRule	Oct 23, 2008	Draft	Technical
RegexDslRule	Oct 23, 2008	Draft	Technical
wee	Jan 27, 2009	Draft	Home Mortgage

Business Asset Manager

JCR 2 Repository, Version Control

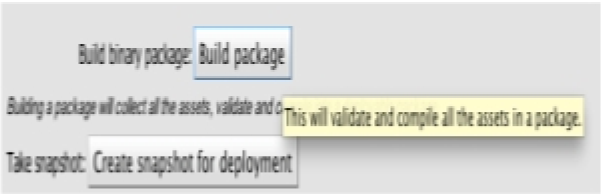
Version Management



Version history

Version number	Comment	Date Modified	Status
1	my change	7/6/07 3:33 PM	Draft
2	another change	7/6/07 3:33 PM	Draft
3	ch ch changes	7/6/07 3:33 PM	Draft

Deployable Packages



Build binary package: Build package

Building a package will collect all the assets, validate and c

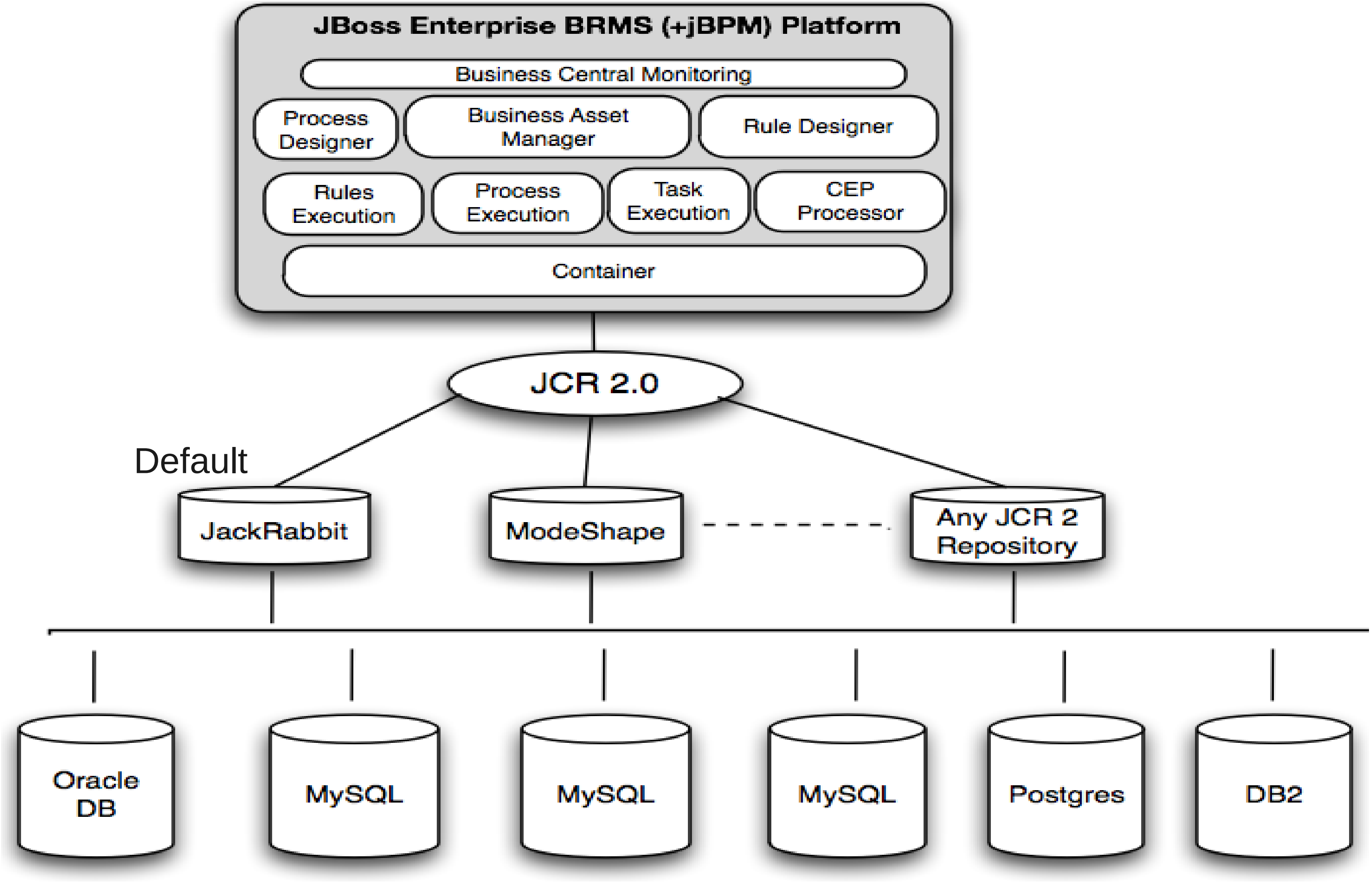
This will validate and compile all the assets in a package.

Take snapshot: Create snapshot for deployment

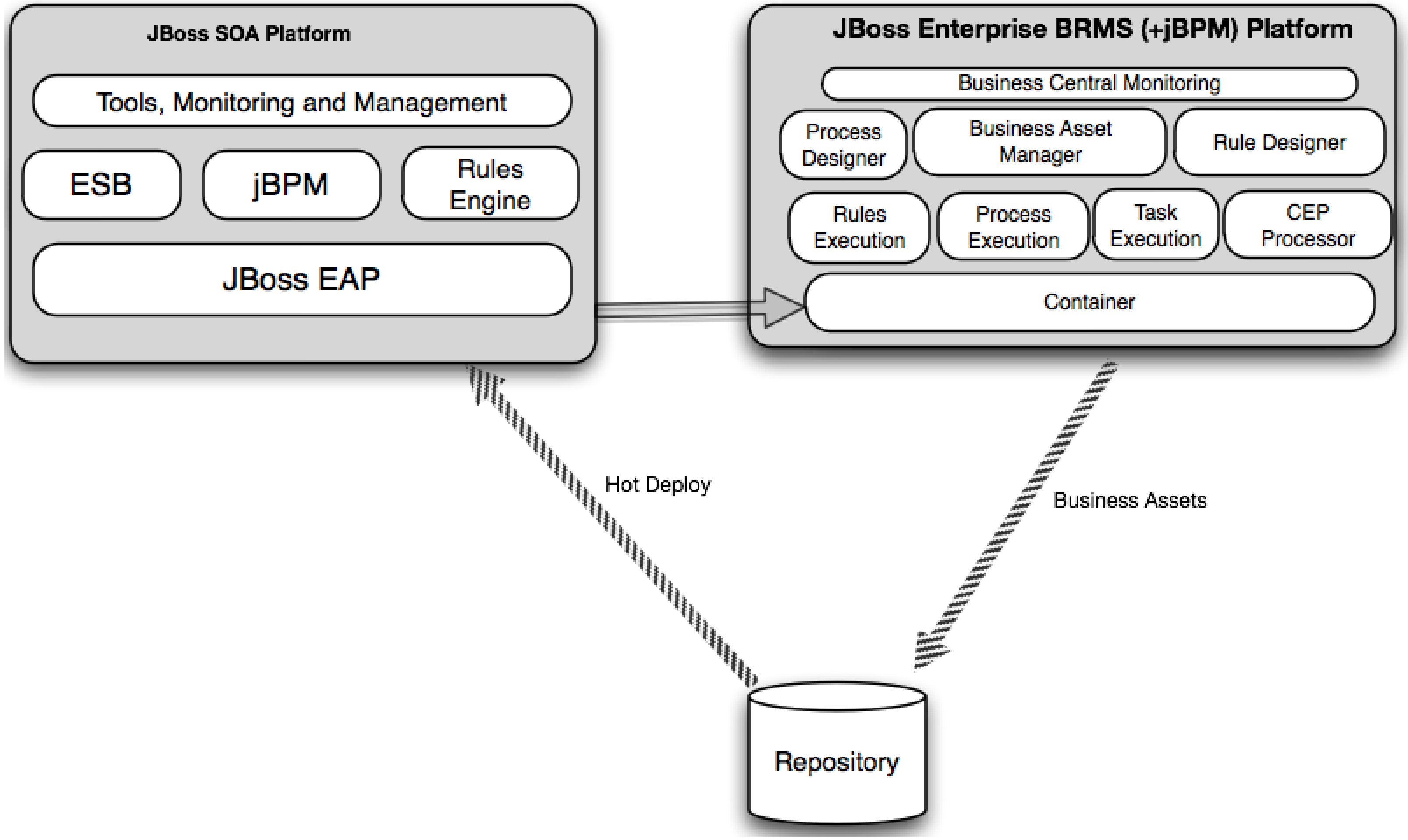
Repository

- Central Knowledge / Asset Repository
- Store – Models, Process Def, Rule Def, Images, Anything Else
- Can store and manage any kind of assets
- Based on JCR 2.0
- Version Control
- Can also be accessed by WebDAV and HTTP
- ***New – Access through REST***
- Further persists to RDBMS

Choice of Repository: Modeshape is Tech Preview



BRMS (+jBPM 5) Interoperability with SOA -P

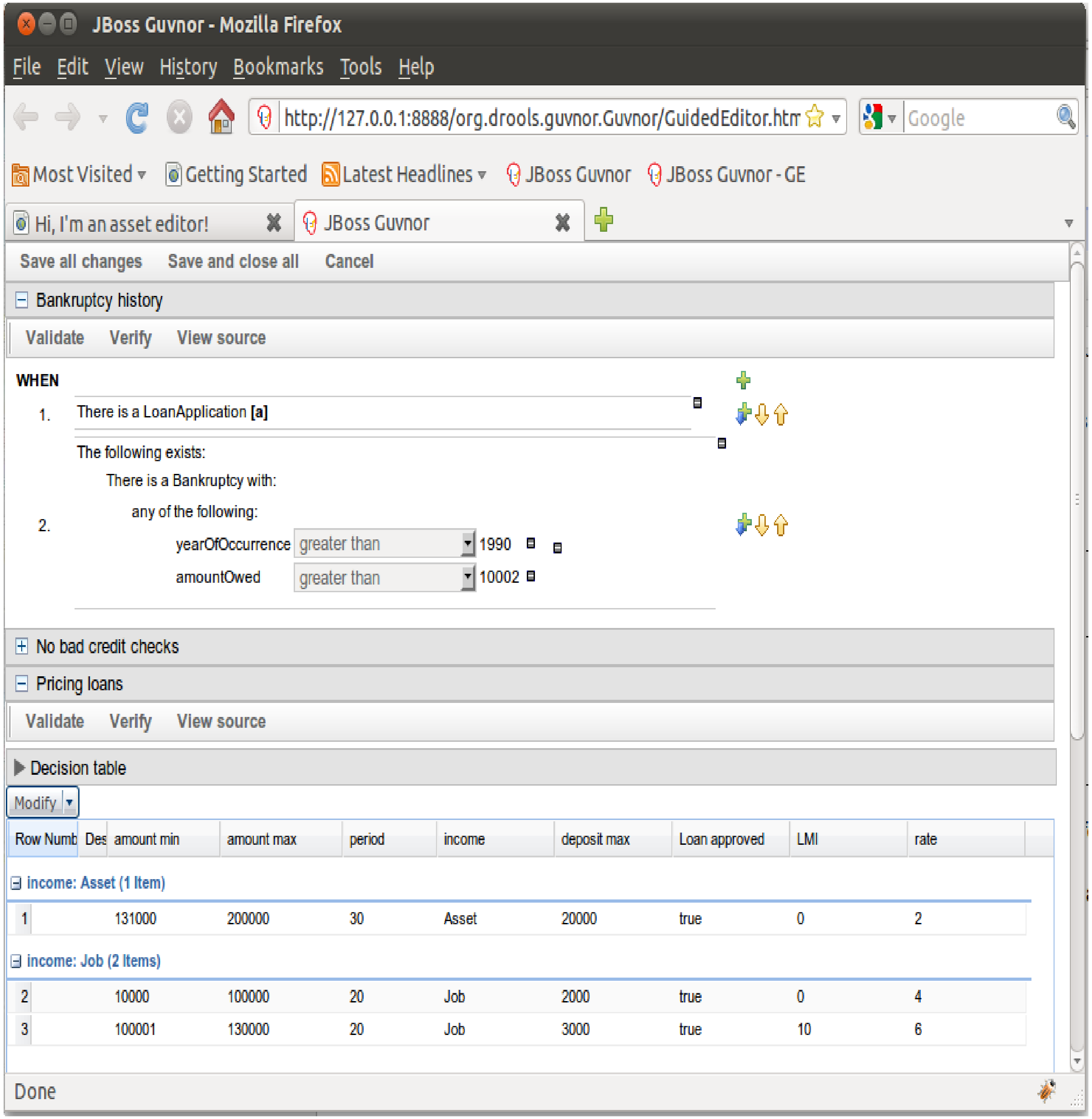


- SOA-P is a certified Container
- Integration of jBPM 5 with ESB in 5.3

Operations Management

- Pluggable to JON
- Monitor long running sessions
- Monitor knowledgebase params for each KB
- View list of running sessions
- List of active rules in sessions
- For each rule session params
- Session params like
  - No. of facts
  - Total firing time
  - Avg firing time
  - Total activations

Embed Guvnor Editors in Custom Applications



http://127.0.0.1:8888/org.drools.guvnor.Guvnor/GuidedEditor.htm

⊟ Bankruptcy history

WHEN

1. There is a LoanApplication **[a]**

2. The following exists:
   There is a Bankruptcy with:
   any of the following:
   yearOfOccurrence greater than 1990
   amountOwed greater than 10002

⊞ No bad credit checks

⊟ Pricing loans

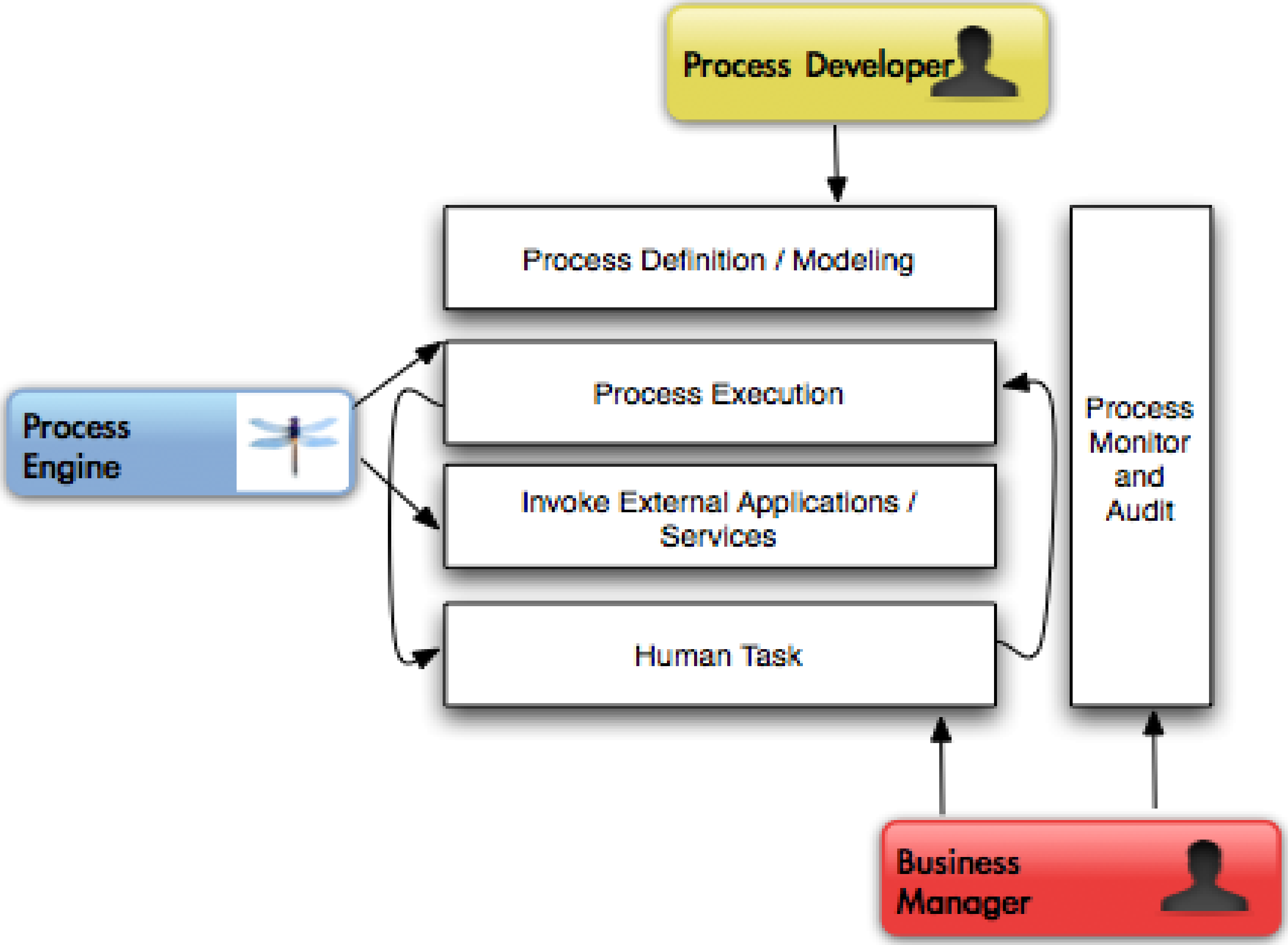
Decision table

Row Numb	Des	amount min	amount max	period	income	deposit max	Loan approved	LMI	rate
⊟ income: Asset (1 Item)									
1		131000	200000	30	Asset	20000	true	0	2
⊟ income: Job (2 Items)									
2		10000	100000	20	Job	2000	true	0	4
3		100001	130000	20	Job	3000	true	10	6

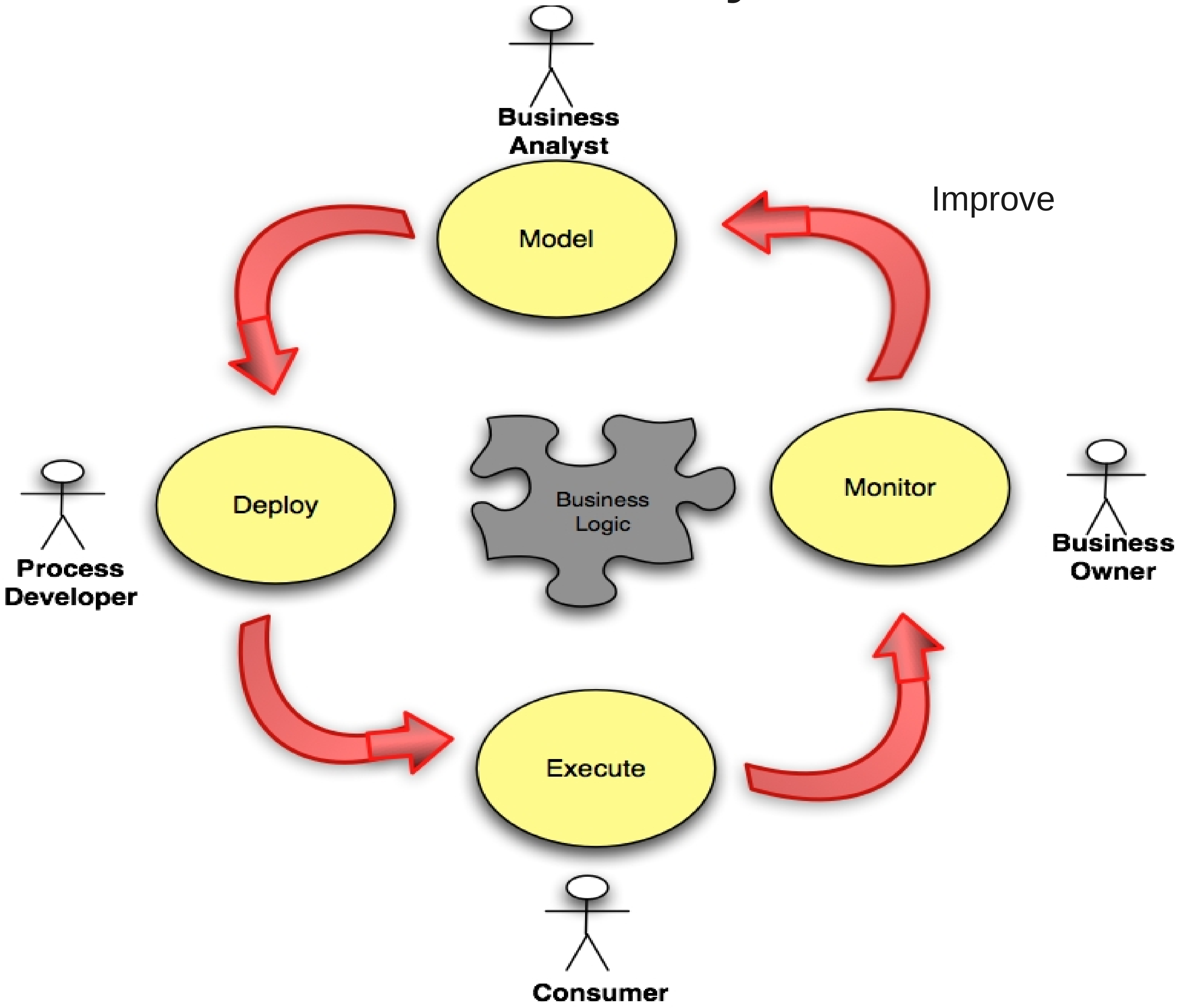
- Embed Any editor in your app
- Guvnor webapp should be running
- BRL Editor mode – no repo saving
- Existing Asset Editor mode – save to repo
- New Asset Editor mode
- Use JavaScript to interact with editor
- Use Http to interact with Guvnor app

Business Process / Workflow Features

Process Management Functions and Roles


Process Developer
Process Definition / Modeling
Process Execution
Invoke External Applications / Services
Human Task
Process Monitor and Audit
Process Engine
Business Manager

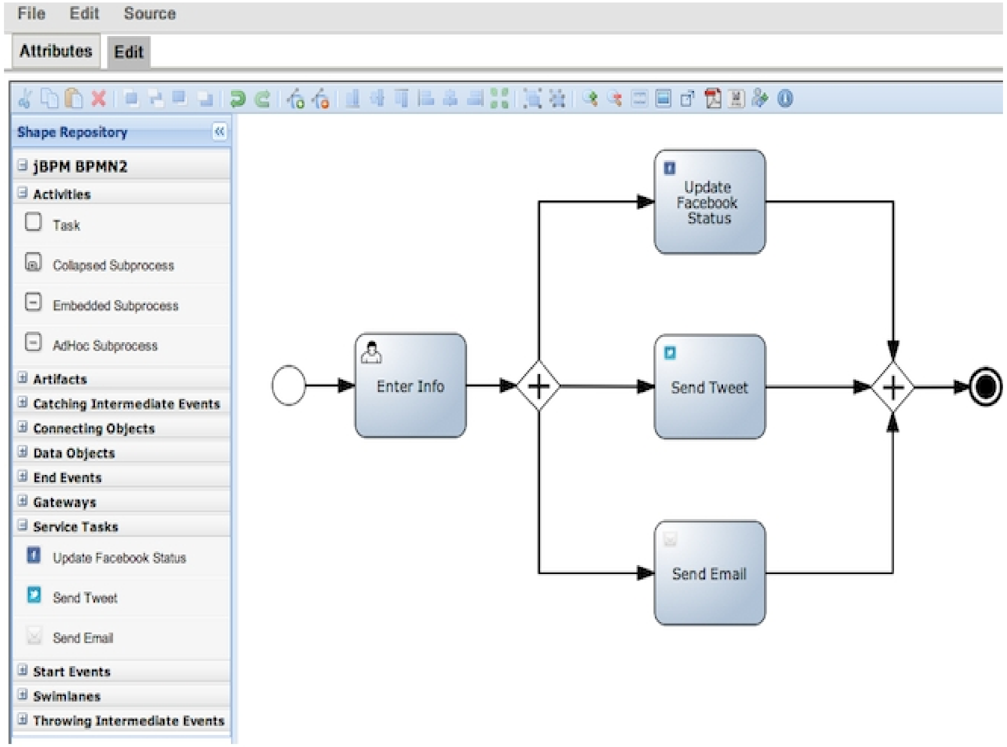
Typical Business Process Lifecycle

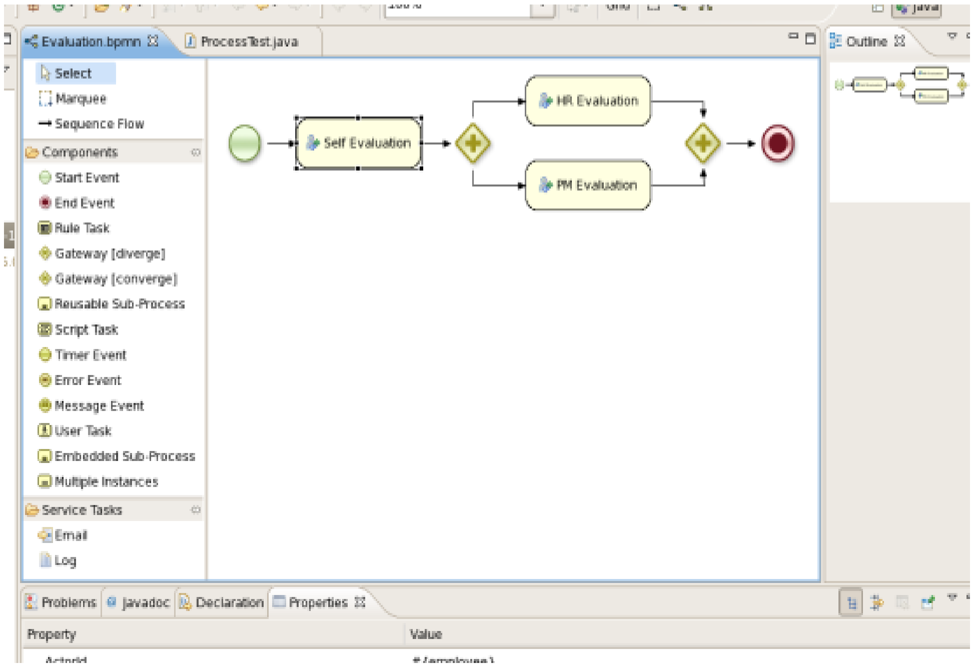


Process Modeling

Web Based Modeling Tools

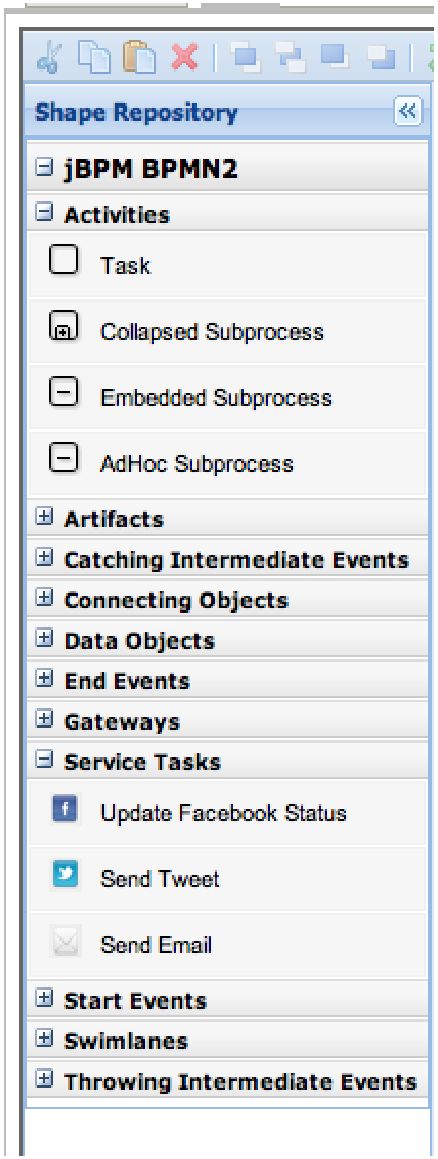
IDE Based Modeling Tools

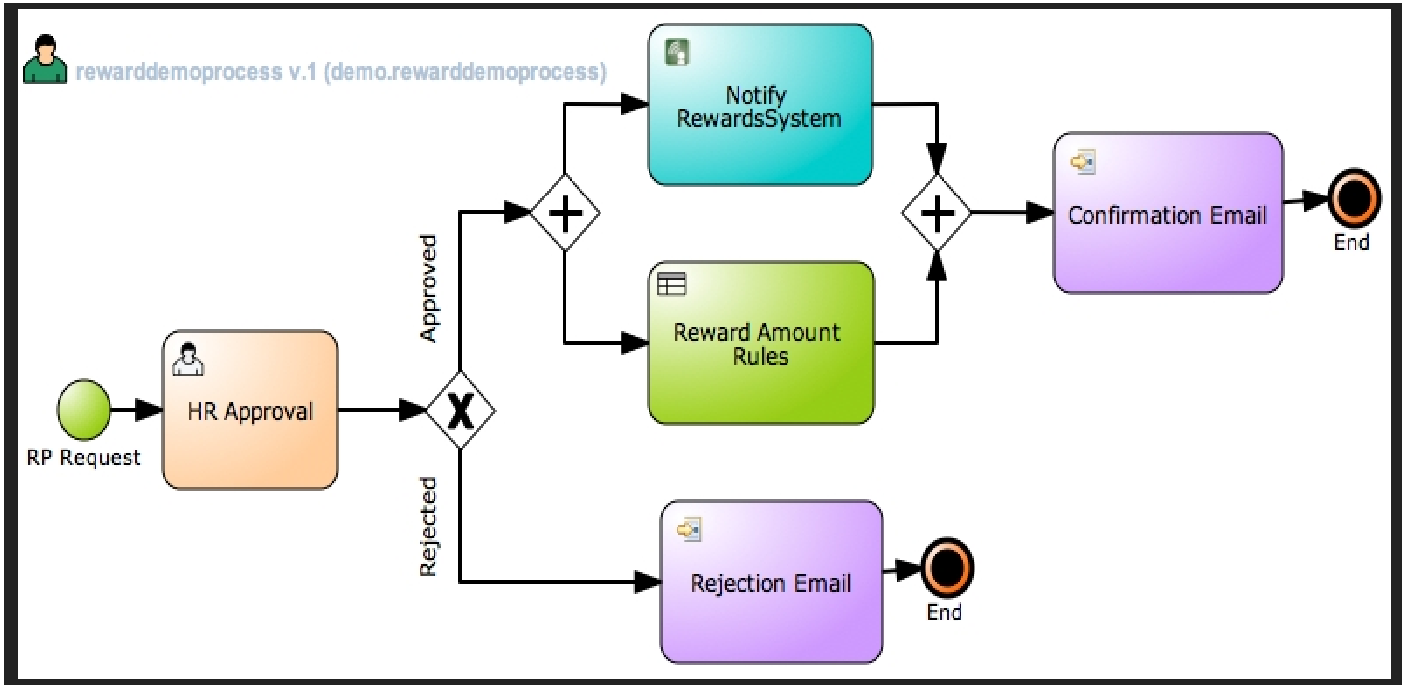




- Create BPMN2 Process Definition
- Common Design Time Repository – Store, Manage, Version
- Create Process Definitions, Validate, Package, Deploy
- Develop Userforms, Integrate other Business Assets (rules etc)
- Import BPMN2 Definitions – Developed by other tools

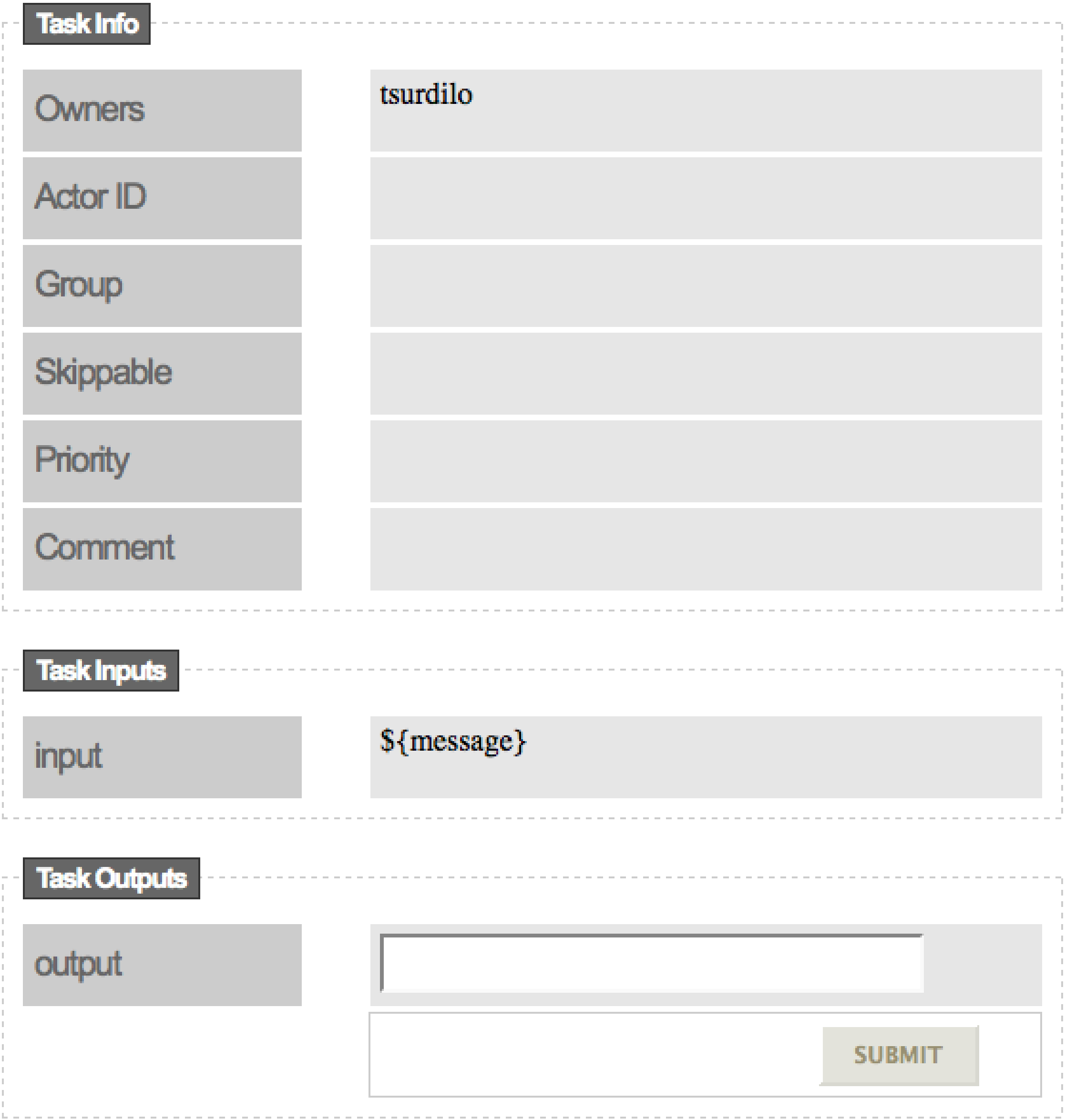
Process Designer Highlights





- Fully Integrated with BRMS Guvnor – On-stop business modeling tool
- Rich set of BPMN2 elements – Drag and drop on canvas
- Domain Specific Nodes – Add your own tasks to the list
- Generate PDF or PNG documents of process definition
- Embeddable Designer – Allows adopting designer in your own tool

Process Designer Highlights – User Forms



Task Info

Owners	tsurdilo
Actor ID	
Group	
Skippable	
Priority	
Comment	

Task Inputs

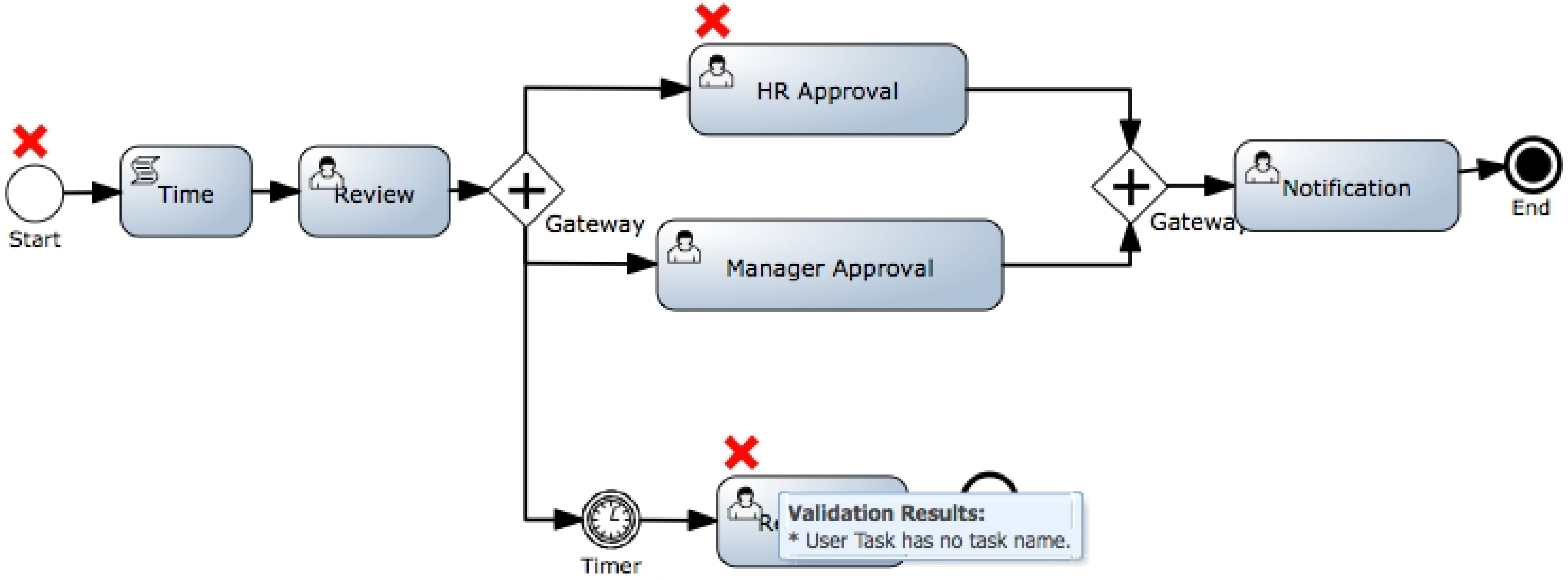
input	${message}

Task Outputs

output	

- Create Task Forms and Process Forms in the designer
- Forms necessary for Workflow kick off and Human task inputs
- These forms are used when process instance is created – Business Central
- Inline editing within the process definition

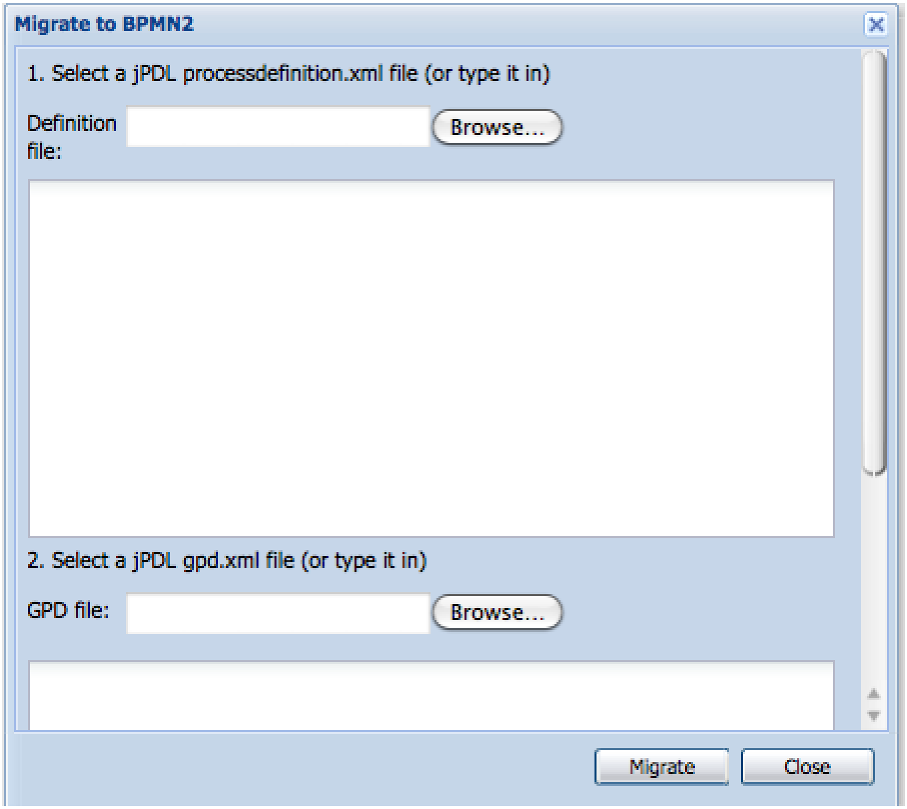
Process Designer Highlights – Visual Process Validation



Validation Results:
* User Task has no task name.

- Visually validate the process definition before packaging

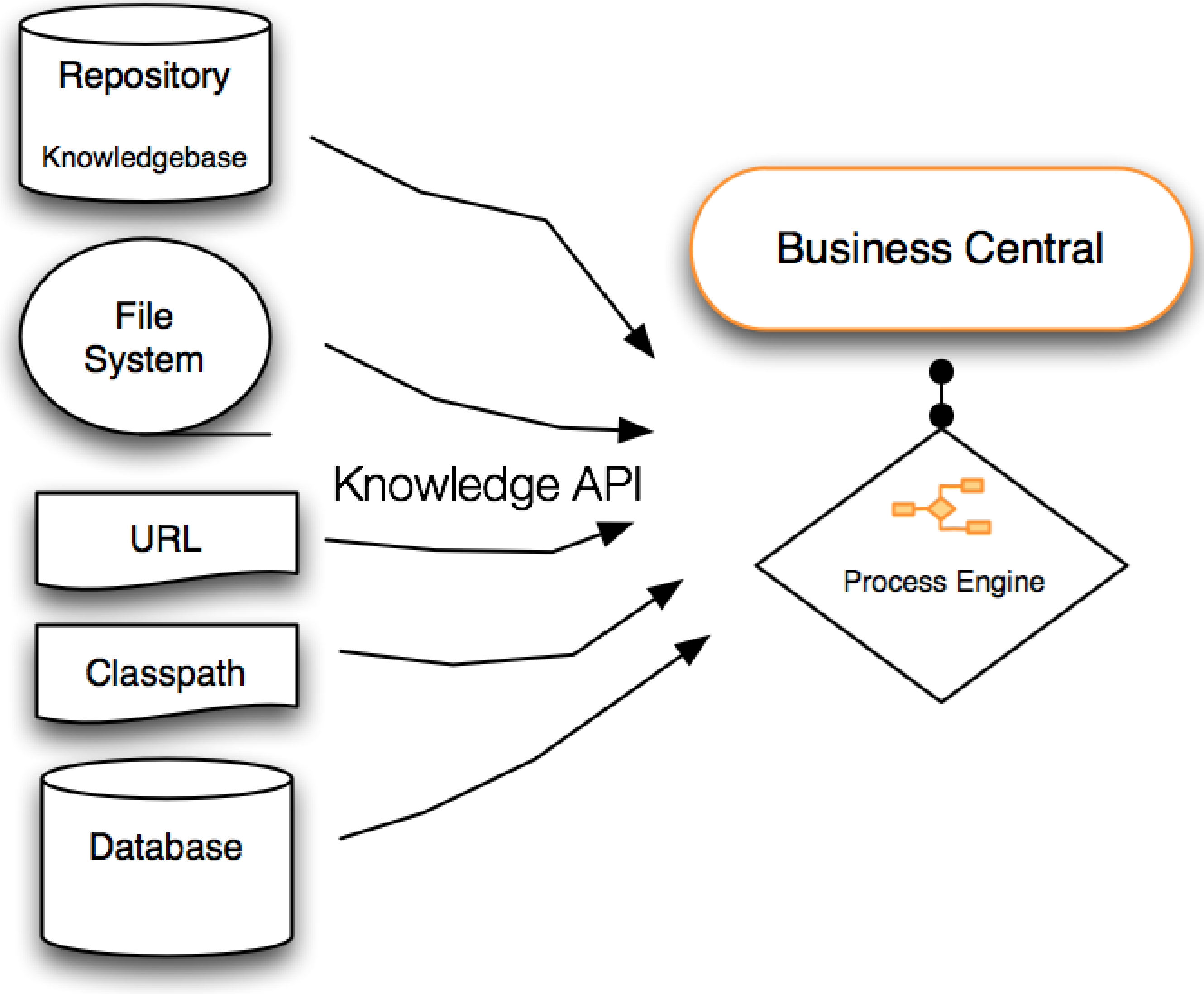
jBPM 3 to 5 Process Migration – Community Supported tool



Migrate to BPMN2
1. Select a jPDL processdefinition.xml file (or type it in)
Definition file:
Browse...
2. Select a jPDL gpd.xml file (or type it in)
GPD file:
Browse...
Migrate
Close

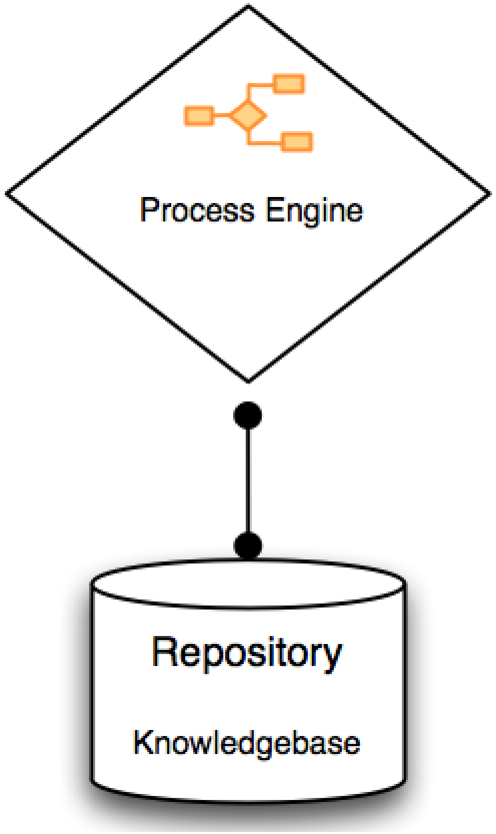
- Not a Supported feature of the product.
- Community Supported Migration Tool - https://github.com/droolsjbpm/jbpmmigration
-

Process Deployment – Loading Process Definition



- Load Definitions through Business Central
- Use Knowledge API to programmatically load
-

Process Execution - Process Engine Highlights

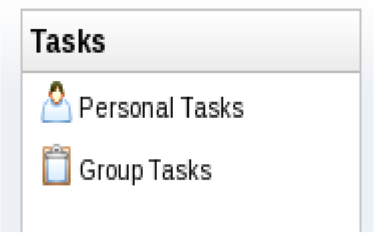


- Lightweight, Embeddable, Extensible
- Manages State Transition
- Native BPMN2 Execution
- Use out of the box Persistence and Transaction
- Or Plugin your Own Persistence and Transaction
- Spring, Camel and OSGi Integration
-

Process Execution - Task Server Highlights

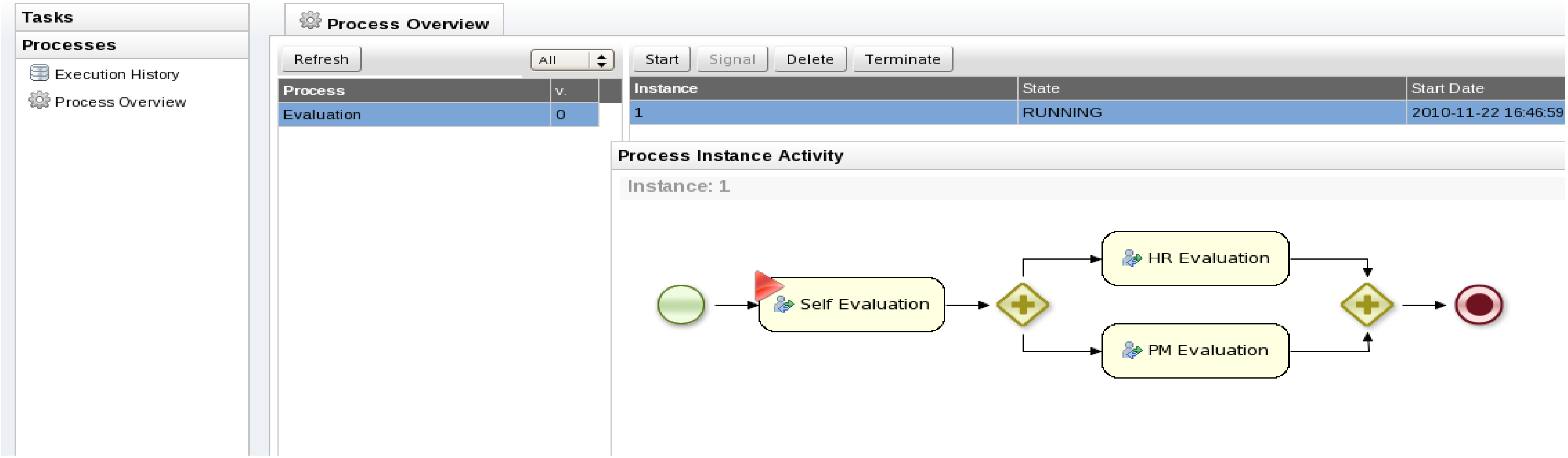
- Independent Task Management Server
- Manages Lifecycle of Human Tasks – Creation, Claim, Completion etc
- Support for Assignment, Delegation, deadlines, calendar integration, internationalization etc.
- Manages its own state persistence – different from process data persistence
- WS-HT Based
- Accessible both by Synchronous and Asynchronous communication
- HornetQ is the default transport.
- Works with Apache Mina
- Accessible through REST interface

Process Monitoring – Business Central



- One-stop Web Based Center for Managing Business Operations
- Manage and Monitor Business Processes
- Manage Human Tasks
- Plug your own reporting tool and create reports
- Role based authentication – configurable to any JaaS / LDAP
- REST Interface for programmatic access to functions

Business Central – Monitoring Processes

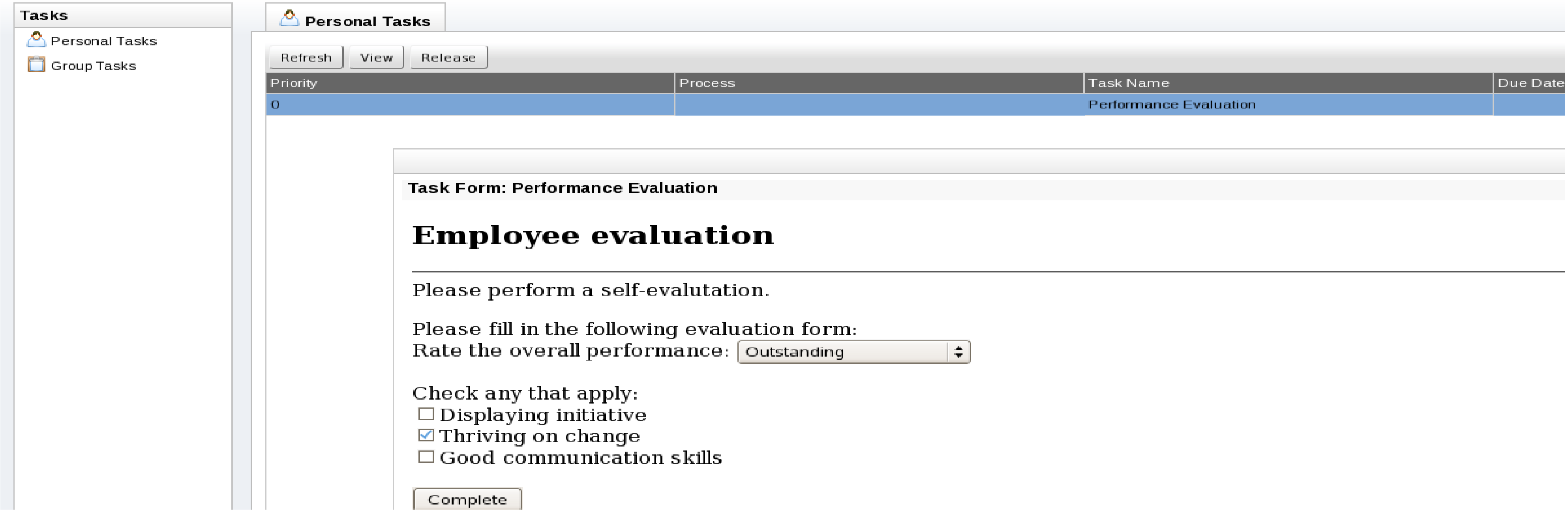


Process	v.
Evaluation	0

Instance	State	Start Date
1	RUNNING	2010-11-22 16:46:59

- View all process definitions that are installed in knowledgebase
- Instantiate a process instance
  - Inline input Process Input form
- Manage running processes – Start, Terminate, Delete, Signal
- Inspect State of a Process Instance and Execution Details
-

Business Central – Managing Tasks



Priority	Process	Task Name	Due Date
0		Performance Evaluation	

Task Form: Performance Evaluation

Employee evaluation

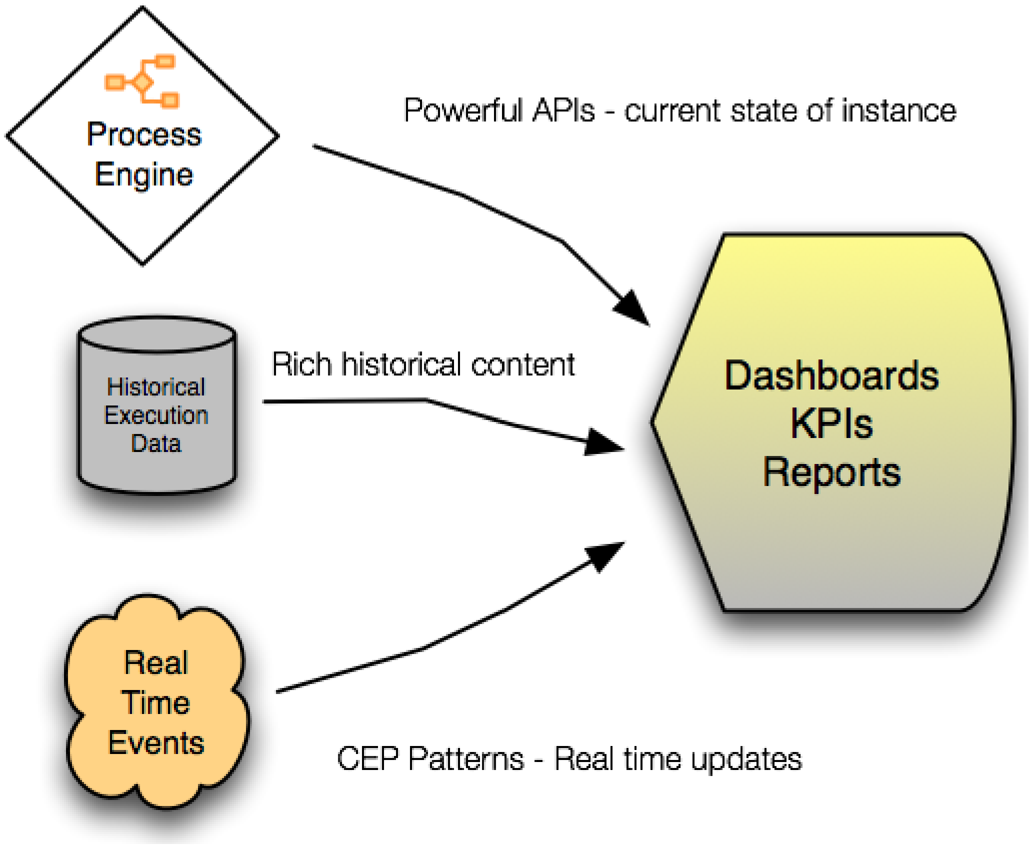
Please perform a self-evalutation.

Please fill in the following evaluation form:
Rate the overall performance: Outstanding

Check any that apply:
□ Displaying initiative
☑ Thriving on change
□ Good communication skills

- View Assigned (personal) tasks and Un-assigned (group) tasks
- Claim a Task, Execute a Task, Release a Task
- Complete inline user task forms

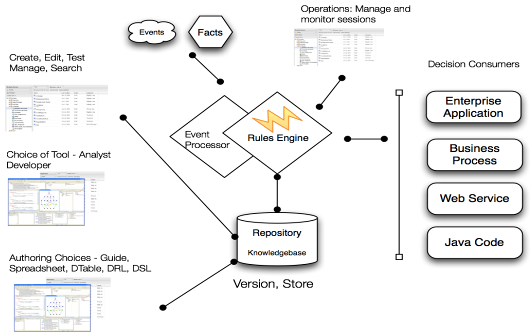
Process Monitoring – Reporting


Process Engine
Powerful APIs - current state of instance
Historical Execution Data
Rich historical content
Dashboards KPIs Reports
Real Time Events
CEP Patterns - Real time updates

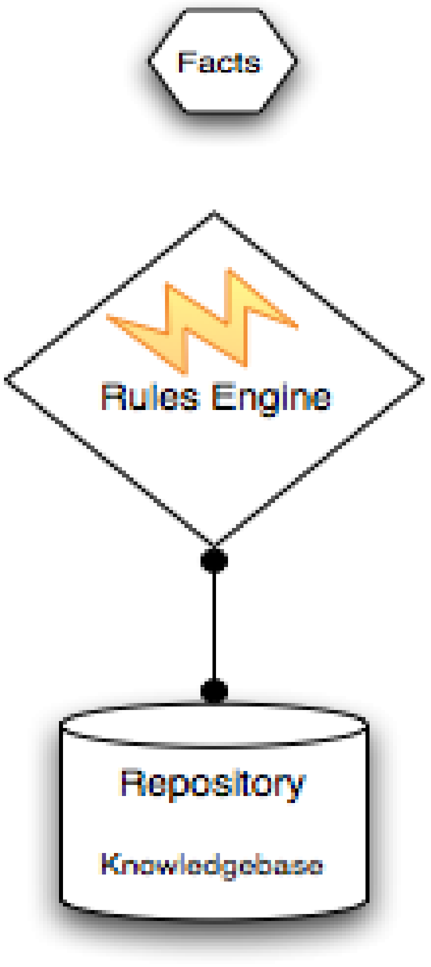
- Powerful Knowledge APIs, REST interface
- Leverage power of CEP for realtime reporting
- Easy to integrate popular 3rd party reporting solutions
- Will supply how to guides
- Placeholder for dashboard. No out of the box reporting in the product

Business Rules and management Features

Rules Engine



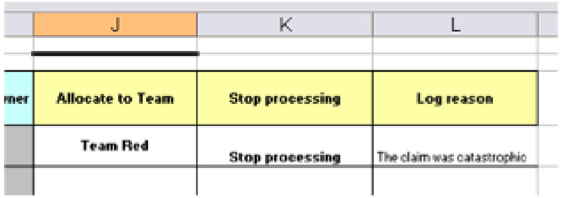
Events
Facts
Operations: Manage and monitor sessions
Create, Edit, Test Manage, Search
Decision Consumers
Event Processor
Rules Engine
Enterprise Application
Business Process
Web Service
Java Code
Choice of Tool - Analyst Developer
Repository
Knowledgebase
Authoring Choices - Guide, Spreadsheet, DTable, DRL, DSL
Version, Store



- Support Inference
- Implements ReteOO algorithm
- Embeddable, lightweight
- Highly performant
- Scalable and most stable
- Supports Declarative Programming

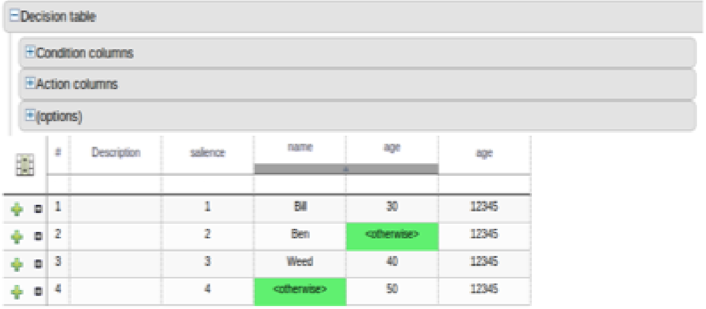
Rule Authoring Choices

Spreadsheet



	J	K	L
mer	**Allocate to Team**	**Stop processing**	**Log reason**
	Team Red	**Stop processing**	The claim was catastrophic

Decision Table



⊟Decision table

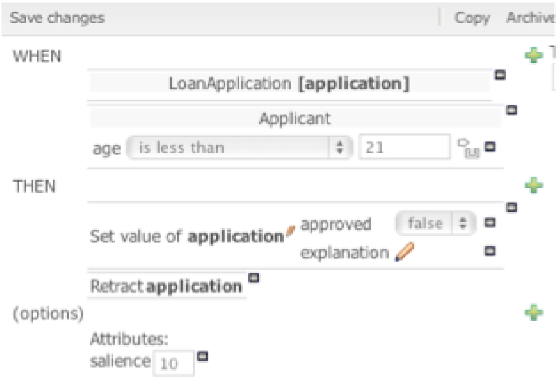
⊞Condition columns

⊞Action columns

⊞(options)

	#	Description	salience	name	age	age
✚ ⊟	1		1	Bill	30	12345
✚ ⊟	2		2	Ben	<otherwise>	12345
✚ ⊟	3		3	Weed	40	12345
✚ ⊟	4		4	<otherwise>	50	12345

Guided Editor



WHEN

LoanApplication **[application]**

Applicant

age | is less than | 21

THEN

Set value of **application** approved | false

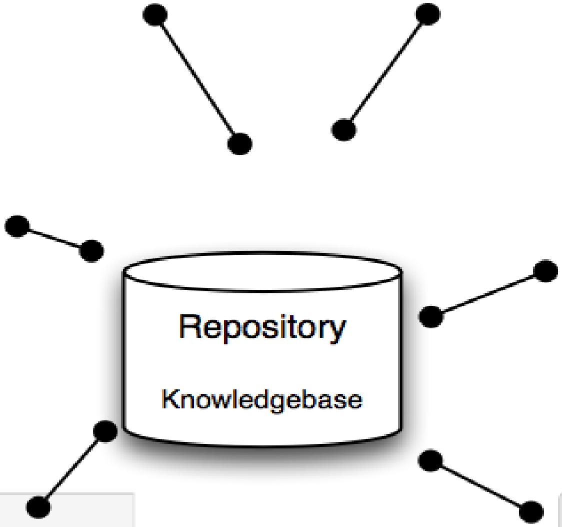
explanation

Retract **application**

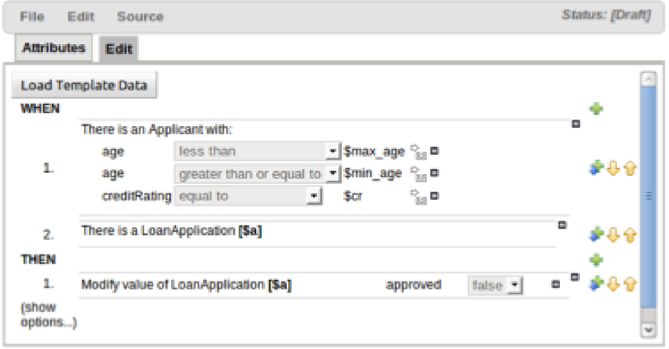
(options)

Attributes:

salience 10



Rule Templates



WHEN

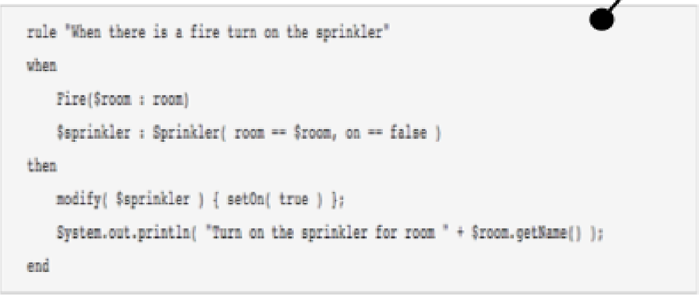
1. There is an Applicant with:
   age | less than | $max_age
   age | greater than or equal to | $min_age
   creditRating | equal to | $cr
2. There is a LoanApplication **[$a]**

THEN

1. Modify value of LoanApplication **[$a]** approved | false

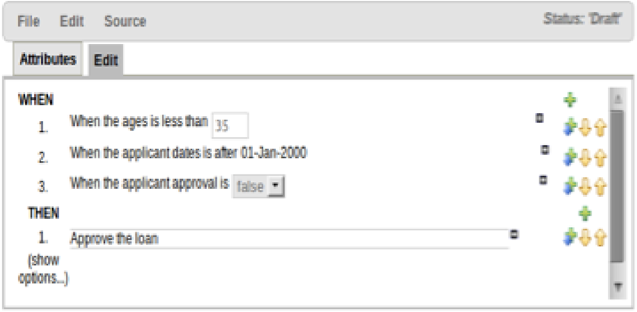
(show options...)

DRL



```
rule "When there is a fire turn on the sprinkler"
when
    Fire($room : room)
    $sprinkler : Sprinkler( room == $room, on == false )
then
    modify( $sprinkler ) { setOn( true ) };
    System.out.println( "Turn on the sprinkler for room " + $room.getName() );
end
```

DSL



WHEN

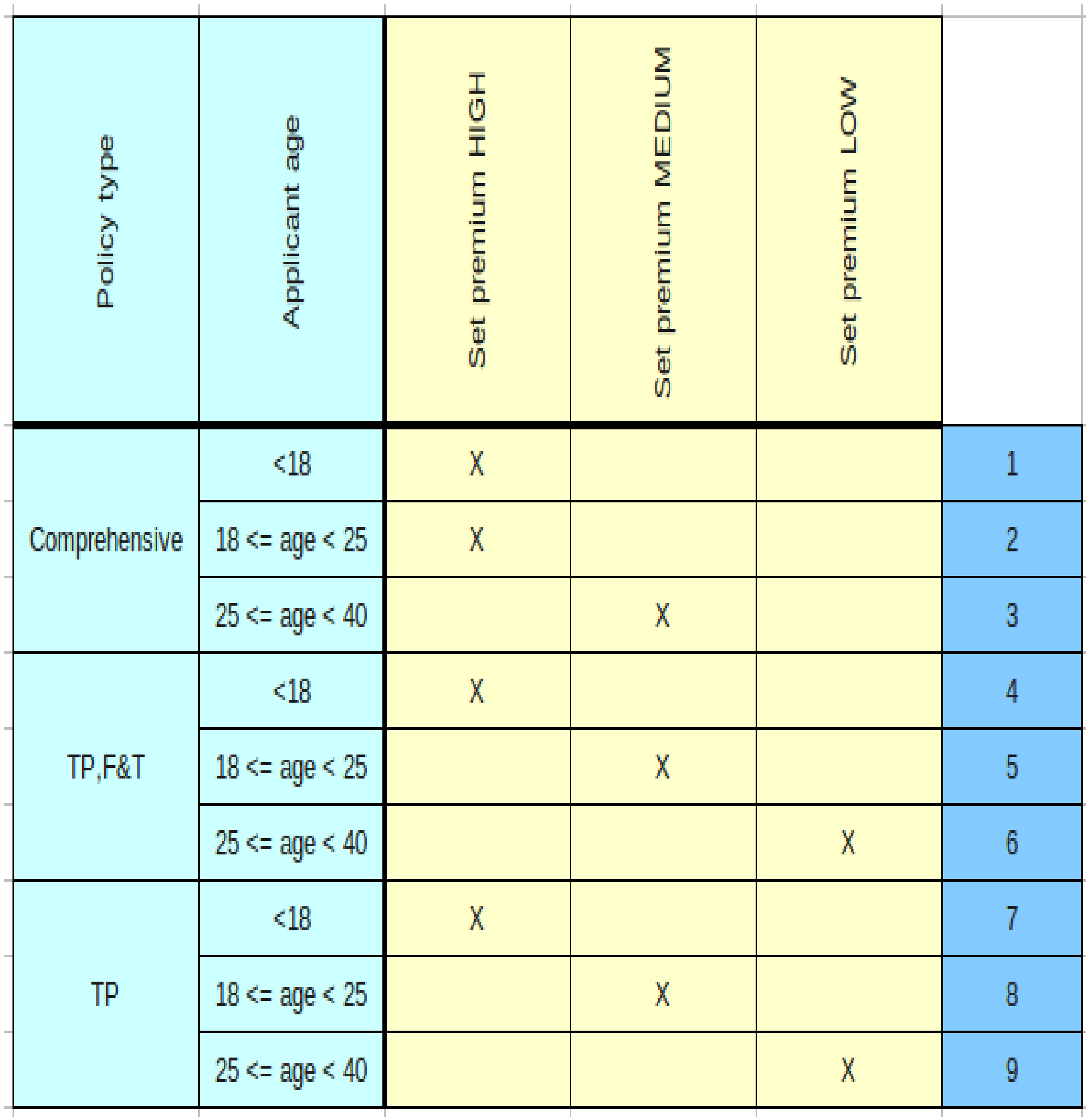
1. When the ages is less than 35
2. When the applicant dates is after 01-Jan-2000
3. When the applicant approval is false

THEN

1. Approve the loan

(show options...)

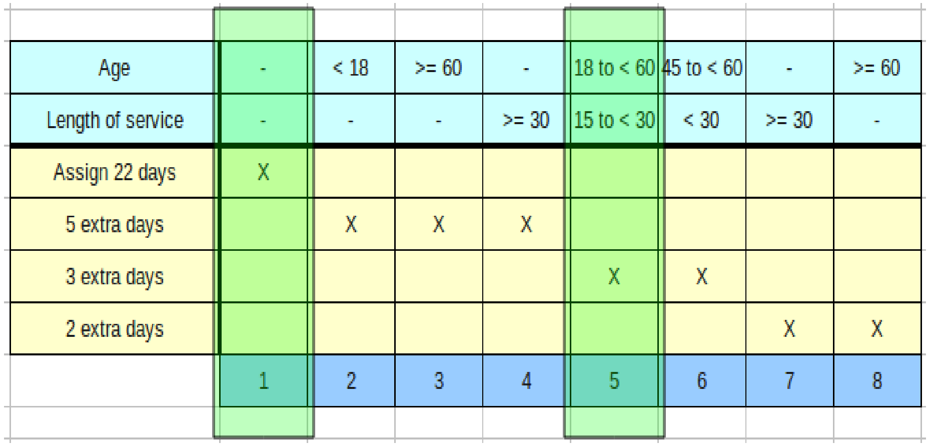
All new Decision Table: Business User Friendly



Policy type	Applicant age	Set premium HIGH	Set premium MEDIUM	Set premium LOW	
Comprehensive	<18	X			1
	18 <= age < 25	X			2
	25 <= age < 40		X		3
TP,F&T	<18	X			4
	18 <= age < 25		X		5
	25 <= age < 40			X	6
TP	<18	X			7
	18 <= age < 25		X		8
	25 <= age < 40			X	9

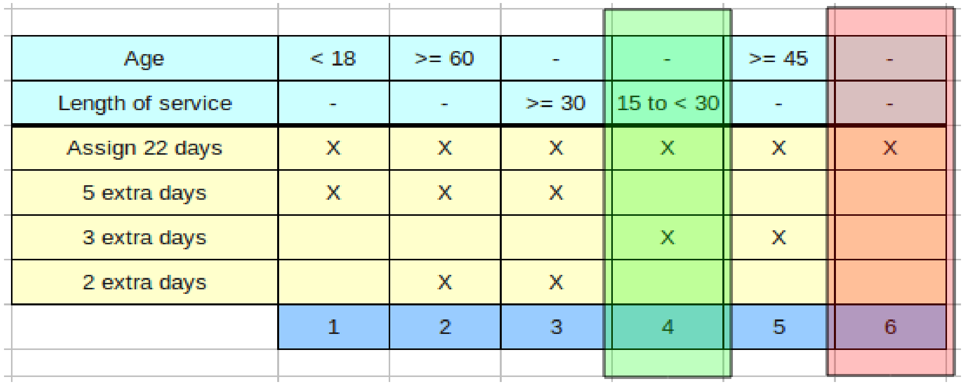
- Intuitive and powerful interface for any users
- Ability to visually create large set of business rules
- Quick validation of rules
- Verify coverage for all conditions

Decision Tables – Supported Categorization



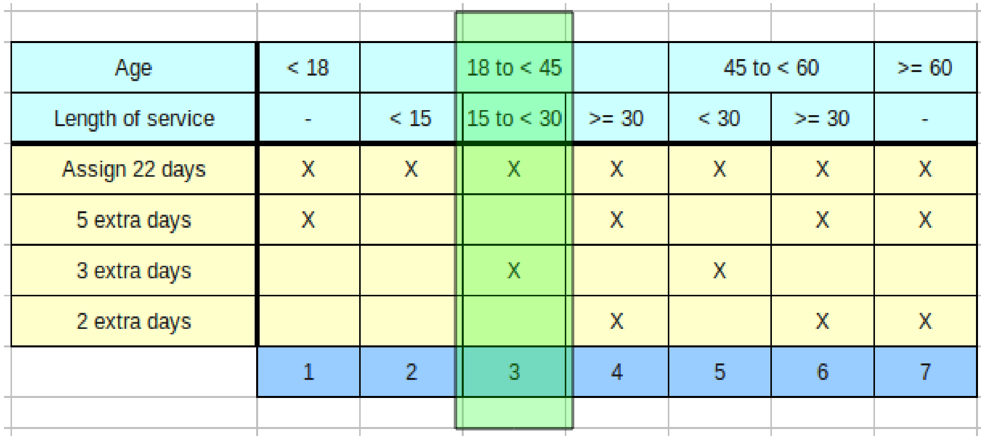
Age	-	< 18	>= 60	-	18 to < 60	45 to < 60	-	>= 60
Length of service	-	-	-	>= 30	15 to < 30	< 30	>= 30	-
Assign 22 days	X							
5 extra days		X	X	X				
3 extra days					X	X		
2 extra days							X	X
	1	2	3	4	5	6	7	8

- Multi hit, all hit
- Conditions are not mutually exclusive
- All matching rules executed



Age	< 18	>= 60	-	-	>= 45	-
Length of service	-	-	>= 30	15 to < 30	-	-
Assign 22 days	X	X	X	X	X	X
5 extra days	X	X	X			
3 extra days				X	X	
2 extra days		X	X			
	1	2	3	4	5	6

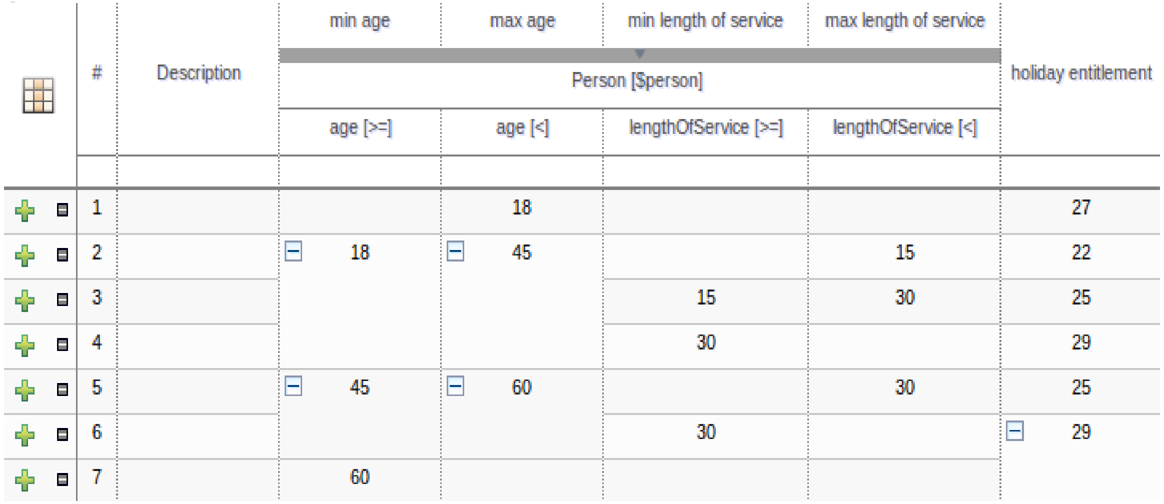
- Multi hit, first hit
- Conditions are not mutually exclusive.
- First matching rule executes



Age	< 18		18 to < 45		45 to < 60		>= 60
Length of service	-	< 15	15 to < 30	>= 30	< 30	>= 30	-
Assign 22 days	X	X	X	X	X	X	X
5 extra days	X			X		X	X
3 extra days			X		X		
2 extra days				X		X	X
	1	2	3	4	5	6	7

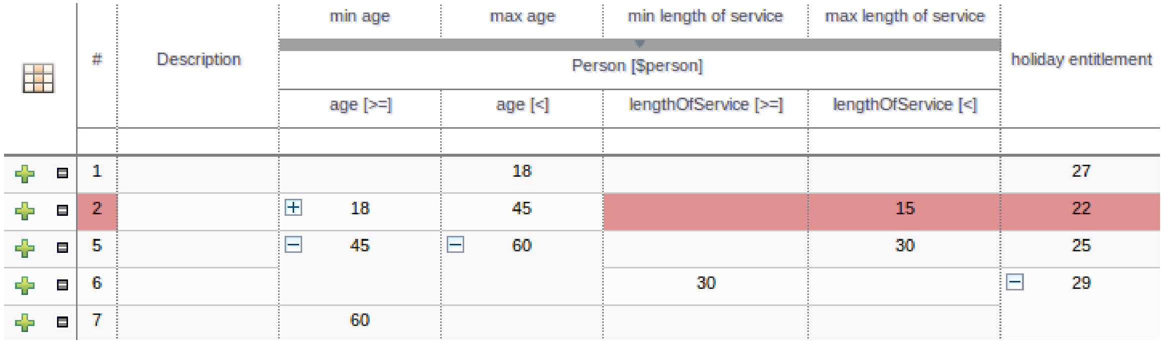
- Single hit
- Exactly One rule fires
- No ambiguity

Decision Tables – Cell merging.. etc



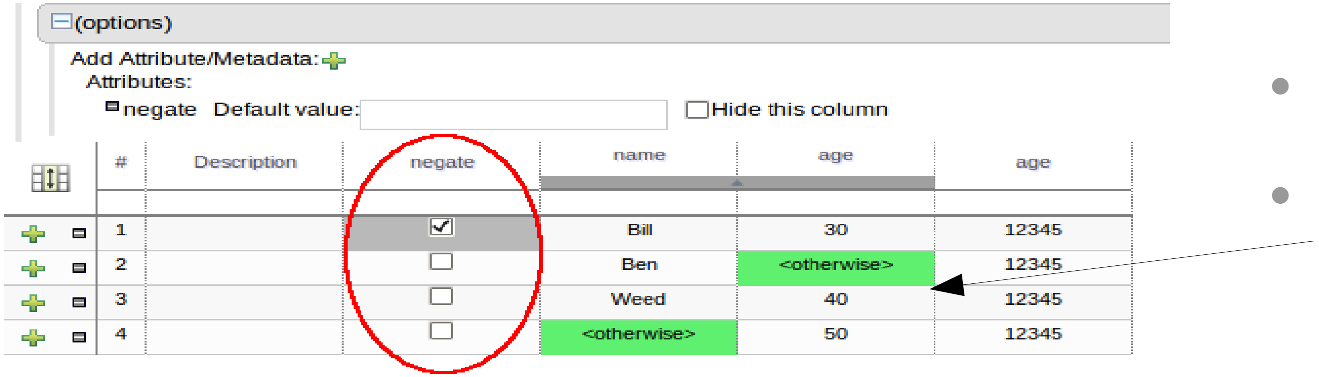
#	Description	min age	max age	min length of service	max length of service	holiday entitlement
		Person [$person]				
		age [>=]	age [<]	lengthOfService [>=]	lengthOfService [<]	
1			18			27
2		18	45		15	22
3				15	30	25
4				30		29
5		45	60		30	25
6				30		29
7		60				

- Cell Merging



#	Description	min age	max age	min length of service	max length of service	holiday entitlement
		Person [$person]				
		age [>=]	age [<]	lengthOfService [>=]	lengthOfService [<]	
1			18			27
2		18	45		15	22
5		45	60		30	25
6				30		29
7		60				

- Cell Grouping
- Typed Columns
- Sorting by column



(options)

Add Attribute/Metadata:

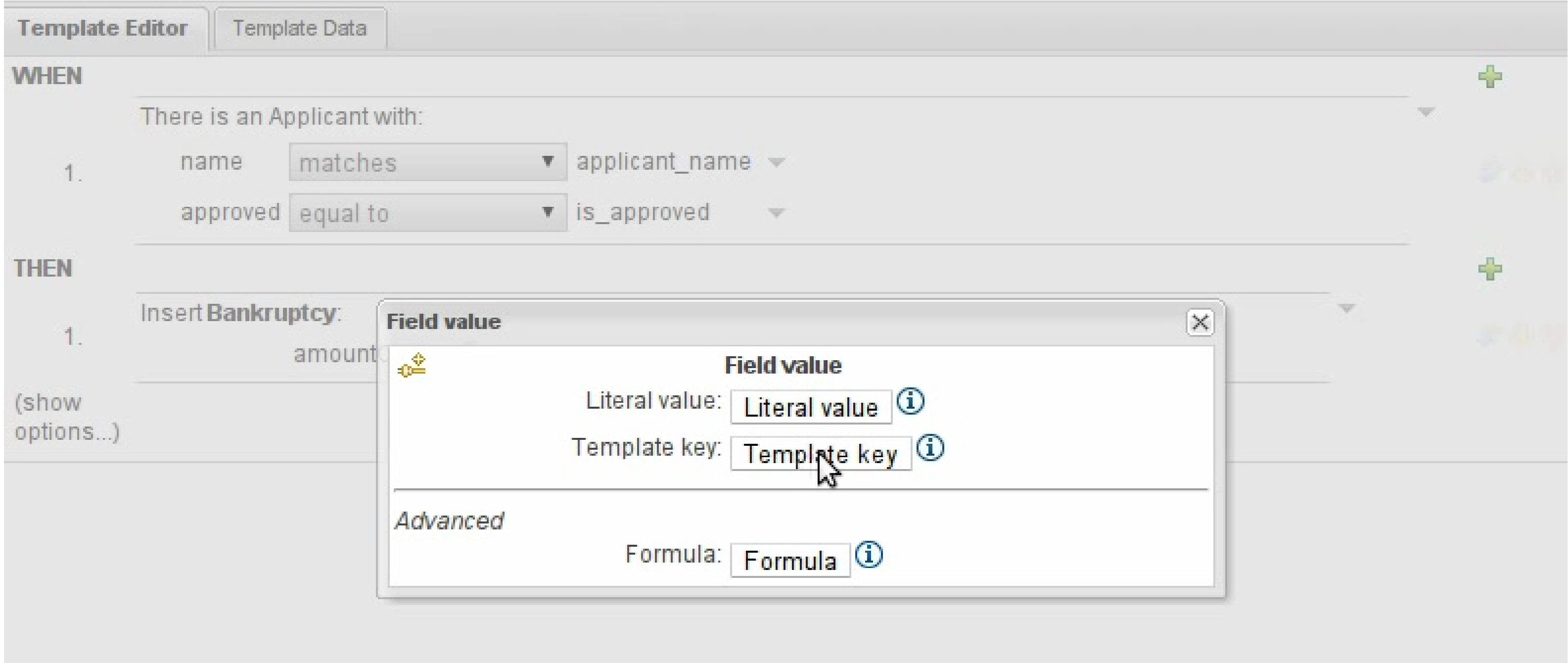
Attributes:

negate Default value: ☐Hide this column

#	Description	negate	name	age	age
1		☑	Bill	30	12345
2		☐	Ben	<otherwise>	12345
3		☐	Weed	40	12345
4		☐	<otherwise>	50	12345

- Negate pattern
- Support for Ohterwise

Rule Templates



WHEN

1. There is an Applicant with:
   - name matches applicant_name
   - approved equal to is_approved

THEN

1. Insert **Bankruptcy**:
   - amount

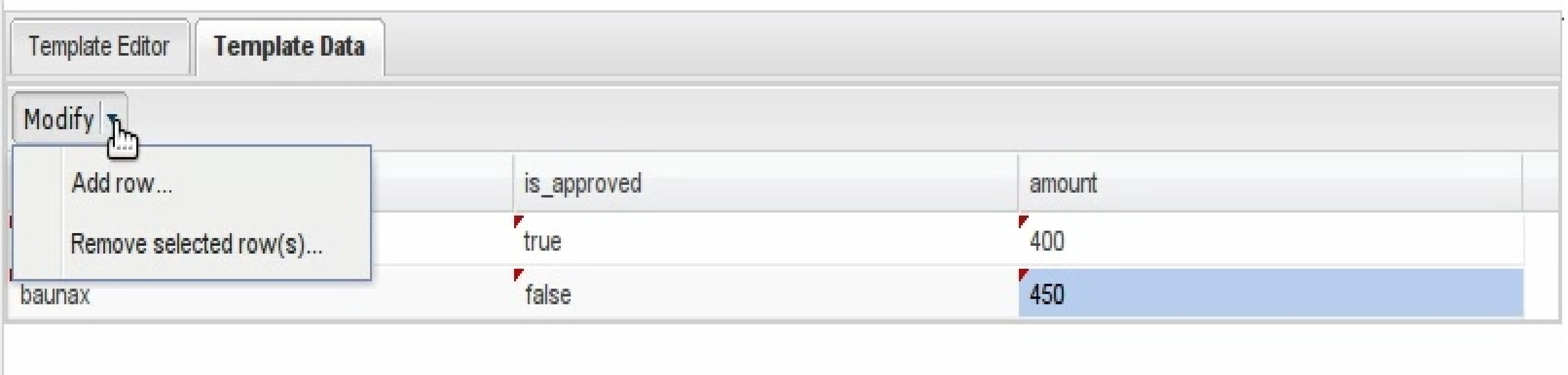
Field value

Literal value: Literal value

Template key: Template key

Formula: Formula

Loading Template Data



	is_approved	amount
	true	400
baunax	false	450

BRMS 5.3 – Other New Features

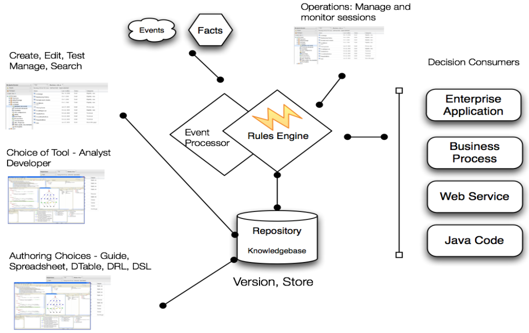
- Backward Chaining – Full Support
- Lazy Truth Maintenance – enabled automagically when needed
- Free form expression in constraints: Enables writing complex expressions and nested accessors

```
Person( age * 2 > $anotherPersonsAge + 2 ) // mathematical expressions

Person( addresses["home"].streetName.startsWith( "High Park" ) ) // method calls and collections simplified syntax

Person( isAdult() ) // boolean expression without relational operator

Cheese( ) from [ $stilton, $brie, $provolone ] // inline list creation and iteration
```

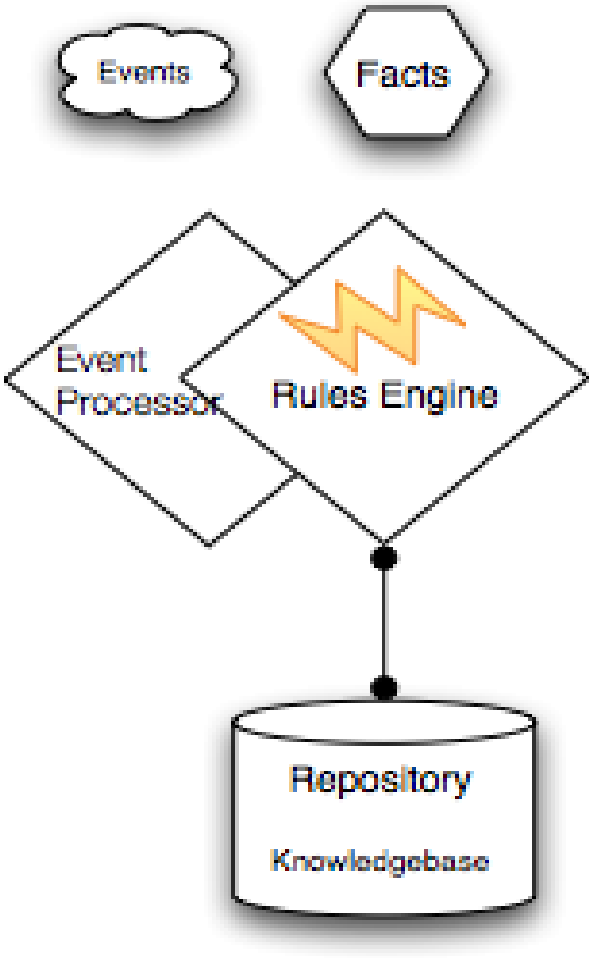
- Performance enhancements
  - Optimized parser enhances
- Modularity enabled for OSGI set up

Business Events / CEP Features

Event Processor



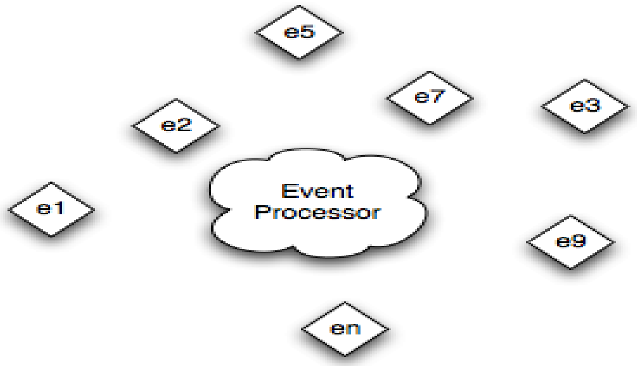
Operations: Manage and monitor sessions
Events
Facts
Create, Edit, Test Manage, Search
Decision Consumers
Enterprise Application
Business Process
Web Service
Java Code
Event Processor
Rules Engine
Choice of Tool - Analyst Developer
Repository
Knowledgebase
Authoring Choices - Guide, Spreadsheet, DTable, DRL, DSL
Version, Store



- CEP Engine enables -
  - Event Detection: Cloud mode and Stream mode
  - Temporal Reasoning (correlation) – reason over aggregation
  - Abstraction – Compose complex events and reason over them
- CEP Engine Supports:
  - Event Semantics (point in time and interval) as first class citizen analogous to rule semantics
  - Support both point in time (as interval with zero duration) and interval semantics
  - Ability to apply temporal constraints
  - Use Session clocks: Support Realtime (system) clock, Psedo Clock (controlled by application)
  - Sliding window support
  - Ability to scale to high volume of events

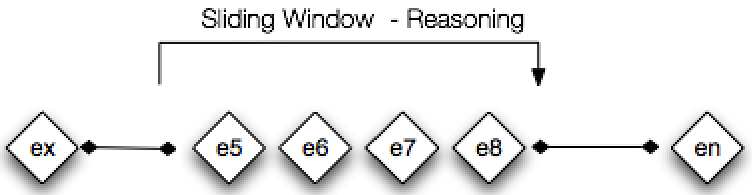
CEP Modes

Cloud Mode



- Default Mode – All facts and events are loaded before reasoning
- Many to many pattern matching by the engine
- No notion of flow of time, no clock synchronization
- Ordering is not required
- Event lifecycle managed by user
- Sliding window is not needed

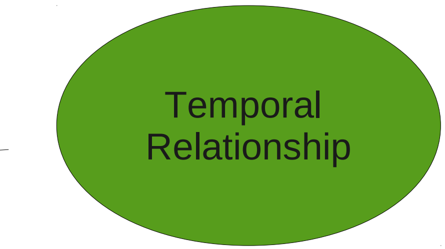
Stream Mode



- Events must be time-ordered
- Engine synchronizes between streams using session clock
- Engine applies the notion of flow
- Engine manages the event lifecycle
- Sliding window option could be used
- Negative patterns could be used. Ex. Fire detected, no sprinkler turned on in 10 sec sound alarm

Temporal Relationships

```
when
    Shipment( $pickupTime : scheduledPickupTime )
    not ShipmentPickup( this before $pickupTime )
then
    // shipment not picked up... Action required.
end
rule “Shipment not picked up in time”
```

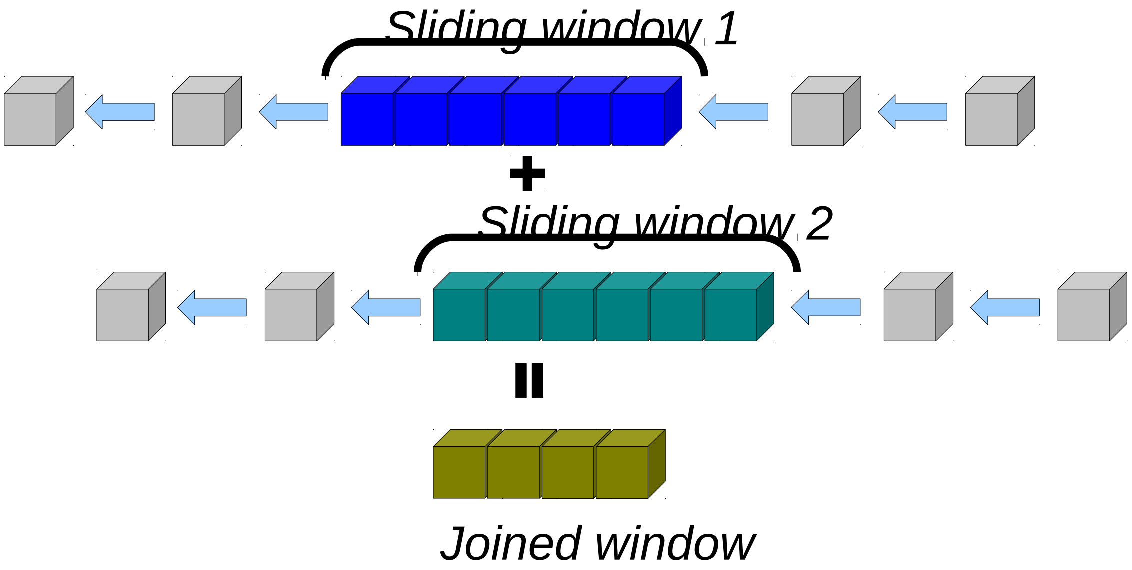


Temporal Relationship

13 Operators are Supported

- Event A before Event B
- Event A meets Event B
- Event A overlaps Event B
- Event A finishes Event B
- Event A includes Event B
- Event A starts Event B
- Event A coincides Event B
- Event A after Event B
- Event A metBy Event B
- Event A overlapedBy Event B
- Event A finishedBy Event B
- Event A during Event B
- Event A finishes Event B

CEP – Sliding Windows



Sliding Time Window

- Reason Over events occurring next set time duration

Example: Raise alarm if avg temp reading from sensor over last 10m is above the threshold

```
rule "Sound the alarm in case temperature rises above
threshold"
when
    TemperatureThreshold( $max : max )
    Number( doubleValue > $max ) from accumulate(
           SensorReading( $temp : temperature ) over
window:time( 10m ),
        average( $temp ) )
then
    // sound the alarm
end
```

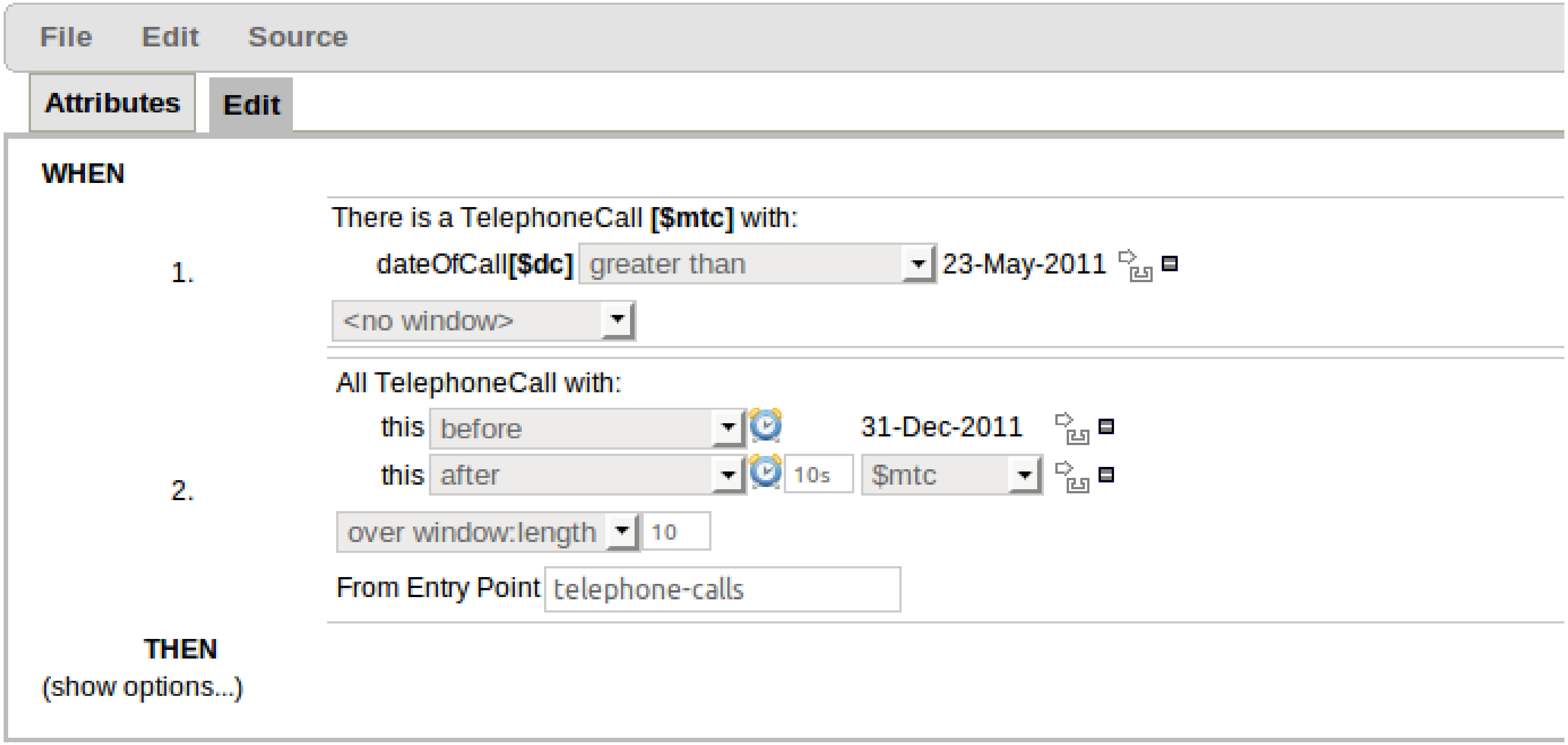
Sliding Length Window

- Reason Over set number of events occuring

Example: Raise alarm if avg temp from last 100 sensor readings is above the threshold

```
rule "Sound the alarm in case temperature rises above
threshold"
when
    TemperatureThreshold( $max : max )
    Number( doubleValue > $max ) from accumulate(
           SensorReading( $temp : temperature ) over
window:length( 100 ),
        average( $temp ) )
then
    // sound the alarm
end
```

Apply CEP operators for Rules



WHEN

1. There is a TelephoneCall **[$mtc]** with:

dateOfCall**[$dc]** greater than 23-May-2011

<no window>

2. All TelephoneCall with:

this before 31-Dec-2011

this after 10s $mtc

over window:length 10

From Entry Point telephone-calls

THEN

(show options...)

- Apply 13 operators available
- Define sliding time window or length

Everything Else