

SUMMIT

**JBoss
WORLD**

PRESENTED BY RED HAT

**LEARN. NETWORK.
EXPERIENCE OPEN SOURCE.**

www.theredhatsummit.com

Business Rules for Mere Mortals

Lamon Gray

JBoss Curriculum Developer, Red Hat

June 22, 2010

SUMMIT

JBoss
WORLD

PRESENTED BY RED HAT



Agenda

- ✓ Business Rules Applied
 - Human-Readable Rules
 - Translating *Domain Specific* Business Rules
- ✓ Understanding Decision Tables
 - Business Rules Design / Maintenance
 - Complex Rules / Multiple Decision Tables
- ✓ JBoss Business Rules Manager IDE & Toolkit
 - Developer Studio
 - Guided Rule Editor
 - Rule Administration – Test – Deploy – Integrate



Introducing Drools



Guvnor 

Expert 

Fusion 

Flow 

SUMMIT

**JBoss
WORLD**

PRESENTED BY RED HAT



Drools Modules



Drools
Expert



Drools
Flow



Drools
Fusion



Drools
Guvnor



(Also Now Drools Planner)

SUMMIT

**JBoss
WORLD**

PRESENTED BY RED HAT



Non-Developers Handle Rules Too...

✓ User Friendliness Factor

Business Analysts

- Easy-to-read
- Simple-to-understand
- Manageable Editing, Creation, and Maintenance



Talk Shop. Explain The Business.

✓ Domain Specific Language - (DSL)

- *Domain* refers to specialized “Business Area”
- Translates *Business Domain Specifics*
 - Translation *Process* defined inside a “.dsl” file
 - Each sentence is stored inside a “.dslr” file
- *Problem-domain* terms converted into “rules”
 - Resulting in a *valid* “.drl” file

Example - **tax.dsl**:

```
[condition] [ ] A taxpayer with id_FEIN  
{taxid} = $taxid : TaxPayer(id_FEIN ==  
{taxid})
```

```
[consequences] [ ] Audit  
TaxPayer=System.out.println("Hello " +  
$taxPayer.getName() );
```

SUMMIT

JBoss
WORLD

PRESENTED BY RED HAT



Rules: Re-Thinking Problem Solving

- A Rule – Basic Features, Attributes

```
package com.gls.jbw.taxes.dsl;  
import com.gls.jbw.tax.model  
expander tax.dsl  
rule "begin taxpayer audit"  
  when  
    A Taxpayer SSN is not entered last year  
  then  
    Flag Taxpayer for Internal Audit  
end
```

- The rule *Name*:
- The *Condition* (Left-Hand Side - LHS)
- The *Consequences* (Right-Hand Side - RHS)
- Keywords: **rule**, **when**, **then**, and **end**
- The **KnowledgeBuilder** as a translator



Decisions, Decisions... Tabling Rules

- Decision tables are a precise yet compact way to model complicated logic (wikipedia).
- Useful when you have lots of rules that follow similar patterns, or templates.
- Want to have a spreadsheet like view for managing rules
- Separate the rule constructs from the data that feeds the rules

	B	C	D	E	F	G
16	Type of New Claim	Is case catastrophic	Allocation code		Insurance Class	Date of accident is after
17	Catastrophic Claim	Y				
18	New Claim with previous Accident num		2			
20	Dependency Claim					
21	Dependency Claim					
22	Interstate Claim					
23	Interstate Claim					
24	Interstate Claim					
25	Interstate Claim					

Each row results in a rule

Each column may be a condition, or action etc.

SUMMIT

JBoss
WORLD

PRESENTED BY RED HAT



Basic Setup

- Header rows contain rule templates, with place holders for data from the rule tables

Microsoft Excel - PolicyPricing.xls

File Edit View Insert Format Tools Data Window Help

Type a question for help

B4 fx

	B	C	D	E	F
1					
2		RuleSet	org.acme.insurance		
3		Notes	This decision table is for working out some basic prices and pretending actuaries		
4					
5		RuleTable Pricing bracket			
6		CONDITION	CONDITION	CONDITION	CONDITION
7			Driver		policy: Policy
8		age >= \$1, age <= \$2	locationRiskProfile	priorClaims == \$param	type
9	Base pricing rules	Age Bracket	Location risk profile	Number of prior claims	Policy type applying for
10			LOW	1	COMPREHENSIVE
11			MED		FIRE_THEFT
12	Young safe package	18, 24	MED	0	COMPREHENSIVE
13			LOW		FIRE_THEFT

Tables Lists

Ready

SUMMIT

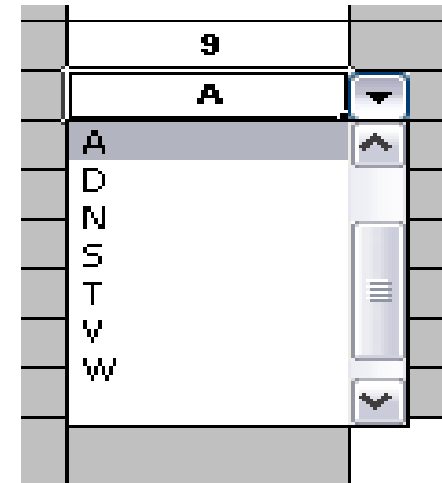
JBoss
WORLD

PRESENTED BY RED HAT



Imports, Variables, and Functions

RuleSet	Control Cajas[1]
Import	foo.Bar, bar.Baz
Variables	Parameters parametros, RulesResult resultado, EvalDate fecha
Functions	<pre>function boolean isRango(int iValor, int iRangoInicio, int iRangoFinal) { if (iRangoInicio <= iValor && iValor <= iRangoFinal) return true; return false; } function boolean isIgualTipo(TipoVO tipoVO, int p_tipo, boolean isNull) { if (tipoVO == null) return isNull; return tipoVO.getSecuencia().intValue() == p_tipo; }</pre>



	9	
	A	▼
A		▲
D		
N		
S		
T		
V		
W		
		▼

- Imports are called Imports
- Globals are called Variables
- Functions are called Functions
- They are all defined under the RuleSet section of the spreadsheet.

SUMMIT

**JBoss
WORLD**

PRESENTED BY RED HAT



Decision Table Anatomy

The screenshot shows a spreadsheet window titled "IntegrationExampleTest - OpenOffice.org Calc". The spreadsheet contains a decision table with the following structure:

	B	C	D	E
7				
8				
9				
10		RuleSet	Some business rules	
11		Import	org.drools.decisiontable.Cheese, org.drools.dec	
12		Sequential	true	
13		RuleTable Cheese fans		
14		CONDITION	CONDITION	ACTION
15		Person	Cheese	list
16	(descriptions)	age	type	add(" \$param")
17	Case	Persons age	Cheese type	Log
18	Old guy	42	stilton	Old man stilton
19	Young guy	21	cheddar	Young man cheddar
20				
21				
22		Variables	java.util.List list	

- Rule Table designates the start of a set of rules with the same template.
- Conditions and Actions are columns.
- A fact with multiple constraints can span multiple columns.



Parameters within Decision Tables

The screenshot shows a Microsoft Excel spreadsheet titled "PolicyPricing.xls". The spreadsheet contains a decision table with the following structure:

1	B	C	D	E	F
2		RuleSet	org.acme.insurance		
3		Notes	This decision table is for working out some basic prices and pretending actuaries		
4					
5		RuleTable Pricing bracket			
6		CONDITION	CONDITION	CONDITION	CONDITION
7			Driver		policy: Policy
8		age >= \$1, age <= \$2	locationRiskProfile	priorClaims == \$param	type
9	Base pricing rules	Age Bracket	Location risk profile	Number of prior claims	Policy type applying for
10			LOW	1	COMPREHENSIVE
11			MED		FIRE_THEFT
12	Young safe package	18, 24	MED	0	COMPREHENSIVE
13			LOW		FIRE_THEFT

- Constraints can be expressed with parameters that are resolved with values from the cells.
- Parameters are prefixed with the \$ sign.
- Multiple parameters can be used in the same constraint.



Multiple RuleTables

- Can have multiple rule tables – each rule table resets the “templates” for the rows below

1	2	3	4	5	6	
1						
2	Module	PRSC[02]				
3	RuleSet	Control Cajas[1]				
8						
9	1.ValidarAperturaCaja (Caja, Registro Estado Sucursal,Transaccion)					
13	ID_Caso de Uso	Caso de Uso	Identificadores de las Reglas	Prioridades de las Reglas	Nombres de las Reglas	Descripciones
14			1	2000	ValidarAperturaCajaSucursal Abierta	Esta Regla tiene por Mision Validar que la sucursal de la se encuentre abierta Trabaja sobre la Caja que se intenta abrir, la Sucursal corresponde a esa caja y la Transacción de Caja apertura
15			2	2000	ValidarAperturaCajaMismaFecha	Esta Regla tiene por Mision Validar que en la sucursal caja se encuentre abierta para la misma fecha de apertura de la caja. Trabaja sobre la Caja que se intenta abrir, la Sucursal corresponde a esa caja y la Transacción de Caja apertura
16						
17						
18	2.ValidarCierreCajasSucursal(Registro Estado Sucursal, TransaccionCaja)					
22	ID_Caso de Uso	Caso de Uso	Identificadores de las Reglas	Prioridades de las Reglas	Nombres de las Reglas	Descripciones
23	C_PRSC_503 C_PRSC_504 C_PRSC_513		1	1000	ValidarCierreCajasSucursal	Esta Regla tiene por Misión Validar que al momento efectuarse el Cierre Contable de una Sucursal de FOL todas las Cajas de esta última se encuentren en Estado Cerrado, es decir la Fecha de Cierre de Caja debe ser a la Fecha de cierre de la entidad Registro_Cierre_Suc
24						
25						
26	3.ValidarTransaccionCaja(Caja, Transaccion_Caja)					
27	RuleTable[3] ValidarTransaccionCaja(CajaVO caja, MovimientoCajaVO movimientoCaja)					
28	ID_Caso de Uso	Caso de Uso	Identificador	Prioridad	Nombre	Descripcion

SUMMIT

JBoss
WORLD

PRESENTED BY RED HAT



Rule Attributes

- The following are added as a column with this name and a value in the cell below:
- PRIORITY - Indicates that this columns values will set the 'salience' values for the rule row. Over-rides the 'Sequential' flag. Optional
- DURATION - Indicates that this columns values will set the duration values for the rule row. Optional
- UNLOOP - Indicates that if there are cell values in this column, the no-loop attribute should be set. Optional
- XOR-GROUP - Cell values in this column mean that the rule-row belongs to the given XOR/activation group . An Activation group means that only one rule in the named group will fire (i.e. the first one to fire cancels the other rules activations).



An Example

Applications Places System 8:00 PM

PolicyPricing.xls - OpenOffice.org Calc

File Edit View Insert Format Tools Data Window Help

Tahoma 7

H3 f0 Σ =

	B	C	D	E	F	G	H	I	J	K
1										
2		RuleSet	org.acme.insurance							
3		Variables	org.jboss.bpm.context.exe.ContextInstance ci							
4		Import	org.acme.insurance.Driver, org.acme.insurance.Policy							
5		Notes	This decision table is for working out some basic prices and pretending actuaries don't							
6										
7		RuleTable Pricing Rules								
8		CONDITION	CONDITION	CONDITION	ACTION	ACTION	ACTION	ACTION	LINLOOP	
9			Driver	policy: Policy	policy					
10		age >= \$1, age <= \$2	locationRiskProfile	priorClaims >= \$1, priorClaims < \$2	type	setBasePrice(\$param)	System.out.println("\$param");	modify(\$policy);	System.out.println(policy.getBasePrice());	
11	Base pricing rules	Age Bracket	Location risk profile	Number of prior claims	Policy type applying for	Base \$ AUD	Record Reason	Modify policy		
12	Young safe package	18, 24	LOW	1,3	COMPREHENSIVE	450		x	x	TRUE
13			MED		FIRE_THEFT	200	Priors not relevant	x	x	TRUE
14			MED	0,1	COMPREHENSIVE	300		x	x	TRUE
15			LOW		FIRE_THEFT	150		x	x	TRUE
16			LOW	0,1	COMPREHENSIVE	150	Safe driver discount	x	x	TRUE
17	Young risk	18,24	MED	1,3	COMPREHENSIVE	700		x	x	TRUE
18			HIGH	0,1	COMPREHENSIVE	700	Location risk	x	x	TRUE
19			HIGH		FIRE_THEFT	550	Location risk	x	x	TRUE
20	Mature drivers	25,30		0,1	COMPREHENSIVE	120	Cheapest possible	x	x	TRUE
21				1,2	COMPREHENSIVE	300		x	x	TRUE
22				2,3	COMPREHENSIVE	590		x	x	TRUE

Tables / Lists

Sheet 1 / 2 PageStyle_Tables 100% STD Sum=0 Average=

eclipse - File Browser [Module 4 - Decision Tables.odp - OpenOffice.org I... [Java - Eclipse SDK] PolicyPricing.xls - OpenOffice.org Calc

SUMMIT

JBoss
WORLD

PRESENTED BY RED HAT



Introducing Tooling... JBoss Developer Studio

SUMMIT

**JBoss
WORLD**

PRESENTED BY RED HAT



Decision Table in Drools Recap

- Decision tables are a handy way to express rules that have the same structure.
- In a decision table, each column is a condition or an action, and each row is a separate rule.
- Decision tables support imports, globals, and functions at the decision table level.
- Decision tables support rule attributes, where each rule attribute is another column.



Jump Into The Lab

- Please see the ***JBW_LAB_INSTR.pdf*** file inside the courseware folder.
- Have Fun!
-

SUMMIT

JBoss
WORLD

PRESENTED BY RED HAT



FOLLOW US ON TWITTER

www.twitter.com/redhatsummit

TWEET ABOUT IT

[#summitjbw](https://twitter.com/summitjbw)

READ THE BLOG

<http://summitblog.redhat.com/>

SUMMIT

**JBoss
WORLD**

PRESENTED BY RED HAT

