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

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

Drools Modules

Drools Expert

Drools Flow

Drools Fusion

Drools Guvnor

(Also Now Drools Planner)

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() );
```


Rules: Re-Thinking Problem Solving

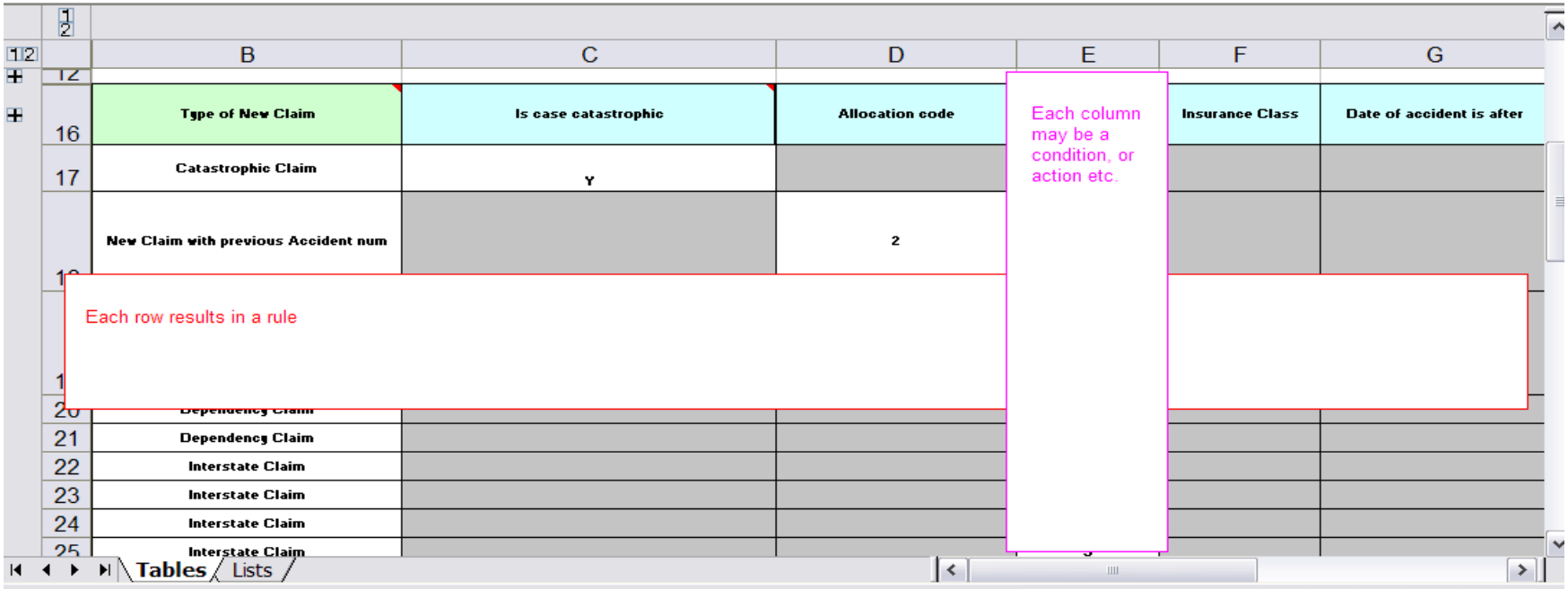
- A Rule – Basic Features, Attributes

```
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

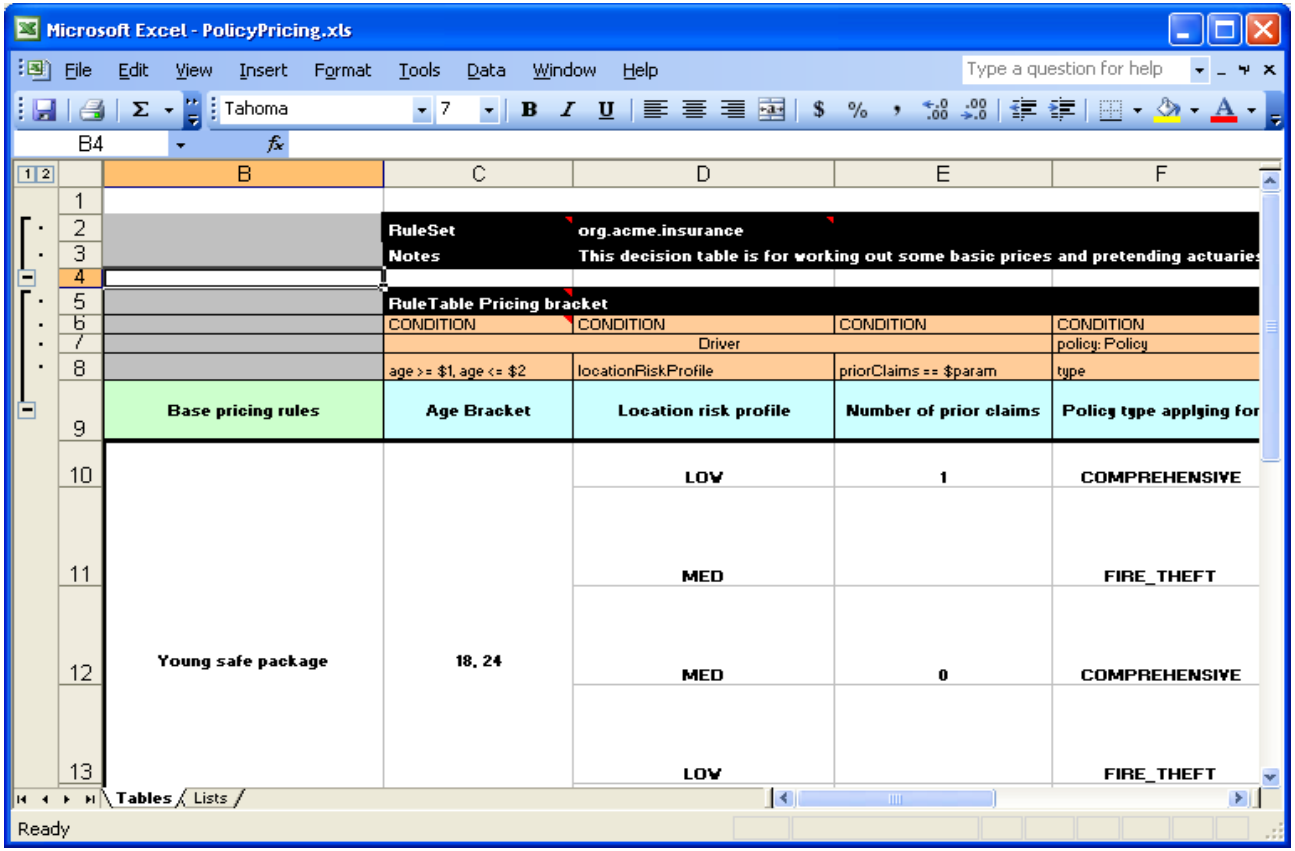
Decisions, Decisions... Tabling Rules

- 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



Basic Setup

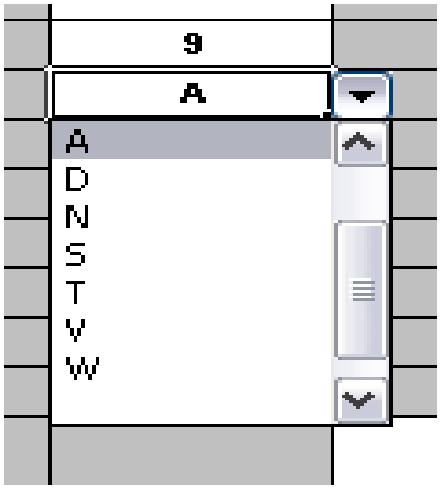
- Header rows contain rule templates, with place holders for data from the rule tables



Imports, Variables, and Functions

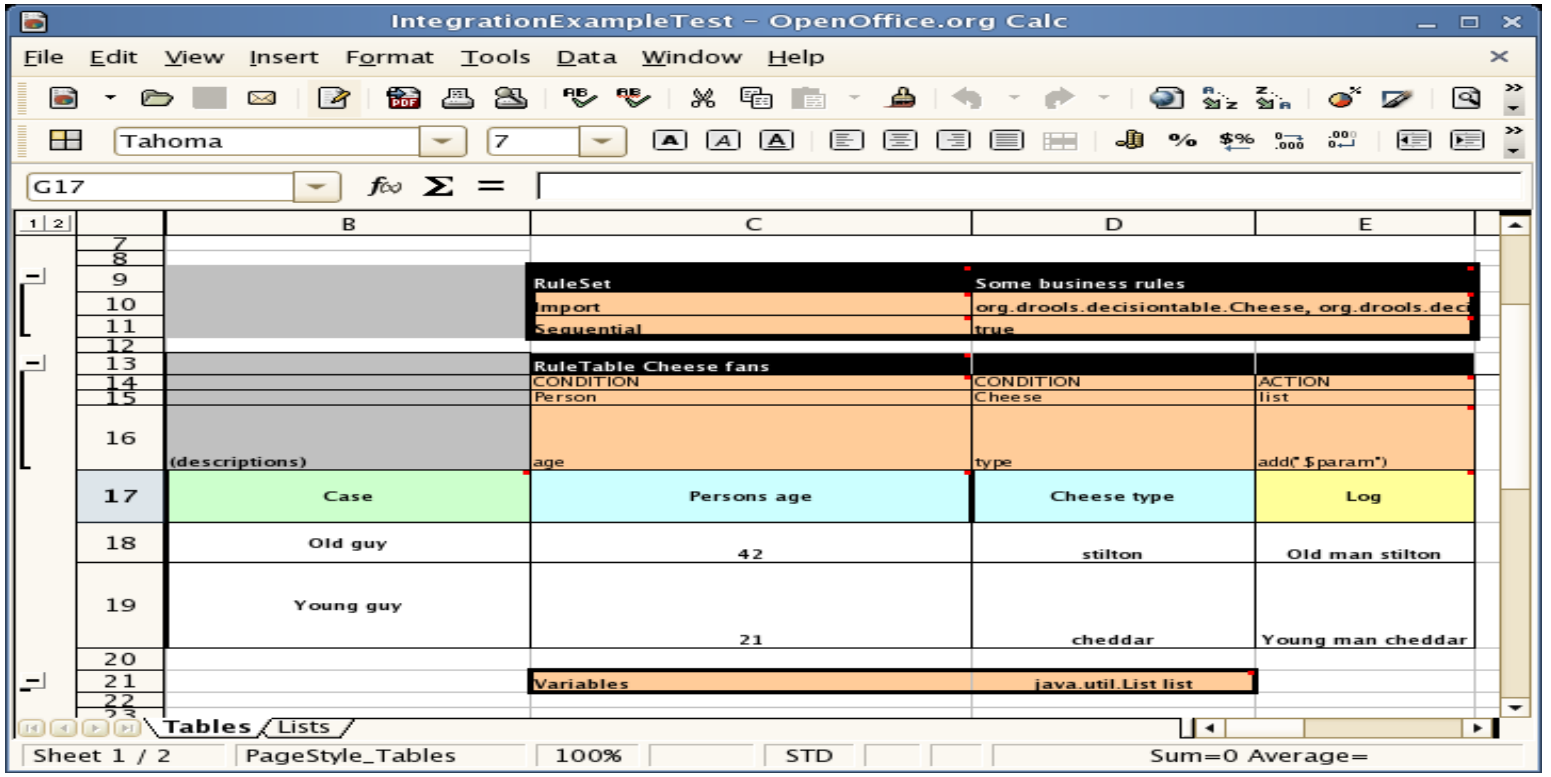
RuleSet	Control Cajas[1]
Import	foo.Bar, bar.Baz
Variables	Parameters parametros, RulesResult resultado, EvalDate fecha

Functions	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;
	}



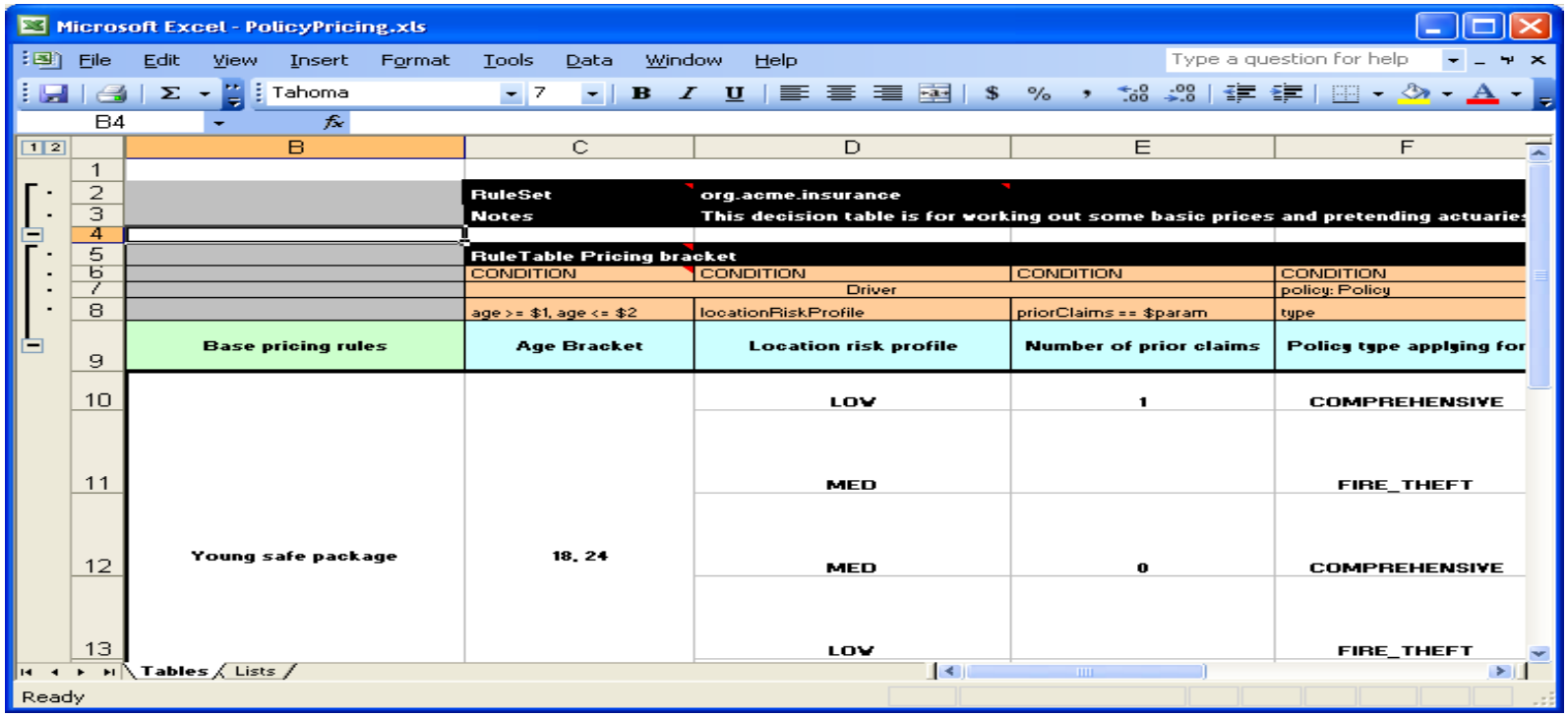
- Imports are called Imports
- Globals are called Variables
- Functions are called Functions
- They are all defined under the RuleSet section of the spreadsheet.

Decision Table Anatomy



- 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



- 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

	Module	PRSC[02]
	RuleSet	Control Cajas[1]

1.ValidarAperturaCaja (Caja, Registro Estado Sucursal,Transaccion)

ID_Caso de Uso	Caso de Uso	Identificadores de las Reglas	Prioridades de las Reglas	Nombres de las Reglas	Descripciones
		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 Sucursa corresponde a esa caja y la Transacción de Caj apertura
		2	2000	ValidarAperturaCajaMismaFe cha	Esta Regla tiene por Mision Validar que en la sucursal caja se encuentre abierta para la misma fecha de ape de la caja. Trabaja sobre la Caja que se intenta abrir, la Sucursa corresponde a esa caja y la Transacción de Caj apertura

2.ValidarCierreCajasSucursal(Registro Estado Sucursal, TransaccionCaja)

ID_Caso de Uso	Caso de Uso	Identificadores de las Reglas	Prioridades de las Reglas	Nombres de las Reglas	Descripciones
C_PRSC_503 C_PRSC_504 C_PRSC_513		1	1000	ValidarCierreCajasSucursal	Esta Regla tiene por Misión Validar que al moment efectuarse el Cierre Contable de una Sucursal de FON todas las Cajas de esta última se encuentren en E Cerrado, es decir la Fecha de Cierre de Caja debe ser a la Fecha de cierre de la entidad Registro_Cierre_Suc

3.ValidarTransaccionCaja(Caja, Transaccion_Caja)

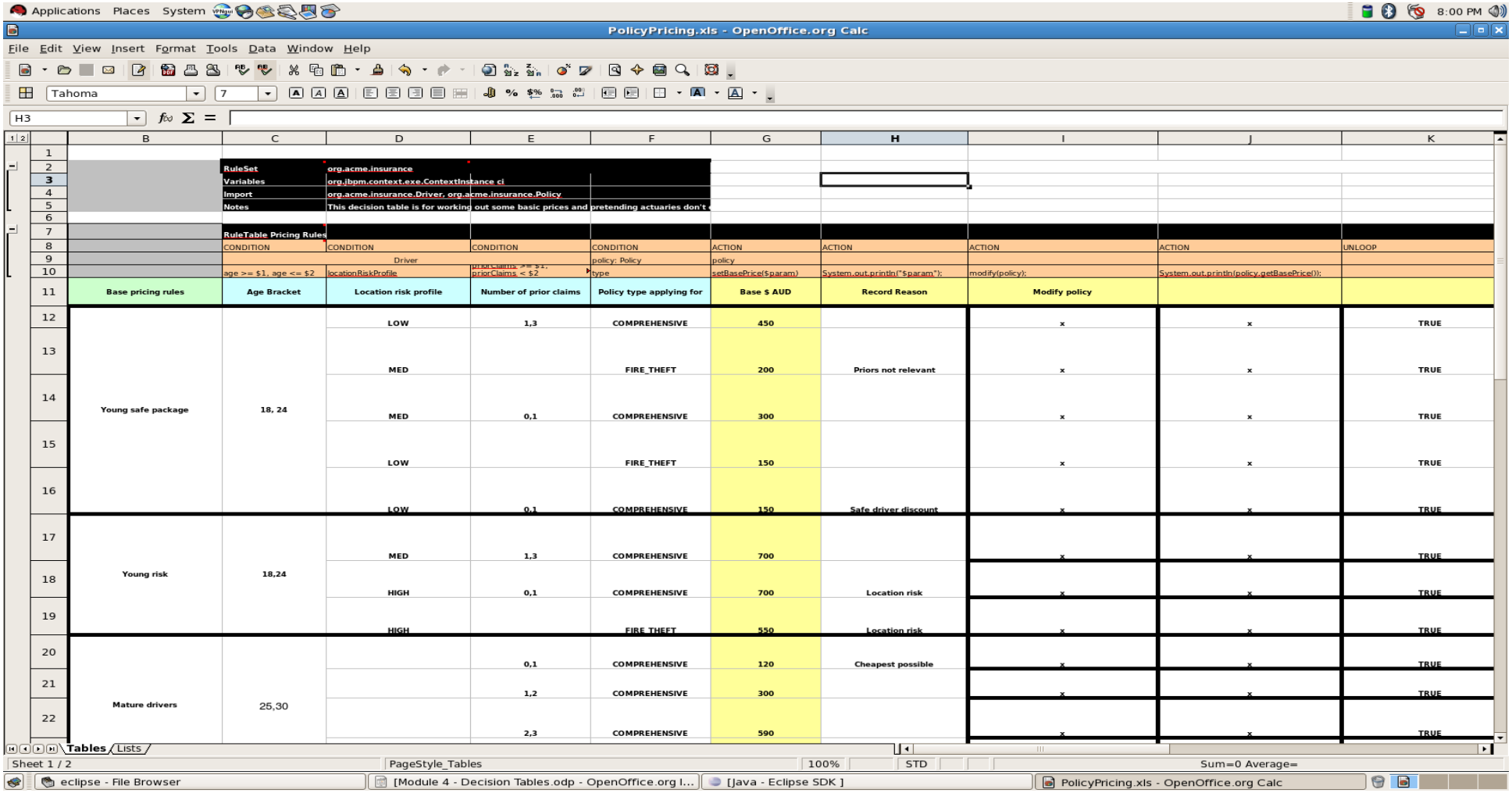
RuleTable[3] ValidarTransaccionCaja(CajaVO caja, MovimientoCajaVO movimientoCaja)

ID_Caso de Uso	Caso de Uso	Identificador	Prioridad	Nombre	Descripcion

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



PolicyPricing.xls - OpenOffice.org Calc

		RuleSet	org.acme.insurance							
		Variables	org.jbpm.context.exe.ContextInstance ci							
		Import	org.acme.insurance.Driver, org.acme.insurance.Policy							
		Notes	This decision table is for working out some basic prices and pretending actuaries don't							
		RuleTable Pricing Rules								
		CONDITION	CONDITION	CONDITION	CONDITION	ACTION	ACTION	ACTION	ACTION	UNLOOP
			Driver		policy: Policy	policy				
		age >= $1, age <= $2	locationRiskProfile	priorClaims >= $1, priorClaims < $2	type	setBasePrice($param)	System.out.println("$param");	modify(policy);	System.out.println(policy.getBasePrice());	
	Base pricing rules	Age Bracket	Location risk profile	Number of prior claims	Policy type applying for	Base $ AUD	Record Reason	Modify policy		
	Young safe package	18, 24	LOW	1,3	COMPREHENSIVE	450		x	x	TRUE
			MED		FIRE_THEFT	200	Priors not relevant	x	x	TRUE
			MED	0,1	COMPREHENSIVE	300		x	x	TRUE
			LOW		FIRE_THEFT	150		x	x	TRUE
			LOW	0,1	COMPREHENSIVE	150	Safe driver discount	x	x	TRUE
	Young risk	18,24	MED	1,3	COMPREHENSIVE	700		x	x	TRUE
			HIGH	0,1	COMPREHENSIVE	700	Location risk	x	x	TRUE
			HIGH		FIRE_THEFT	550	Location risk	x	x	TRUE
	Mature drivers	25,30		0,1	COMPREHENSIVE	120	Cheapest possible	x	x	TRUE
				1,2	COMPREHENSIVE	300		x	x	TRUE
				2,3	COMPREHENSIVE	590		x	x	TRUE

Introducing Tooling... JBoss Developer Studio

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!
-

FOLLOW US ON TWITTER

www.twitter.com/redhatsummit

TWEET ABOUT IT

#summitjbw

READ THE BLOG

http://summitblog.redhat.com/