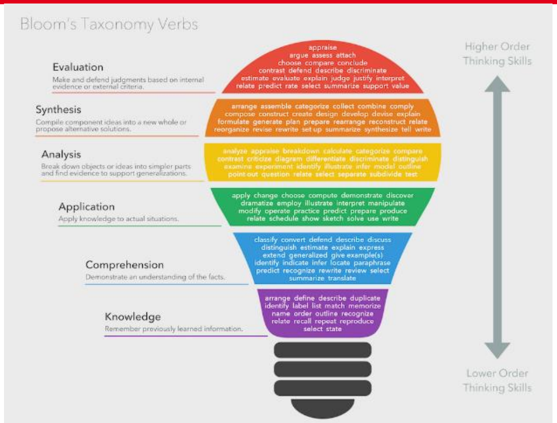
Hochschule für Technik Stuttgart

Master Software Technology
Business Process Technology – [09]
Business Rules Management

Prof. Dr. Lückemeyer, 07.12.2017

Bloom's Taxonomy Verbs
by Fractus Learning,
Lizenz: CC-BY-SA 4.0


Bloom's Taxonomy Verbs
Higher Order Thinking Skills
Evaluation
Make and defend judgments based on internal evidence or external criteria.
appraise argue assess attach choose compare conclude contrast defend describe discriminate estimate evaluate explain judge justify interpret relate predict rate select summarize support value
Synthesis
Compile component ideas into a new whole or propose alternative solutions.
arrange assemble categorize collect combine comply compose construct create design develop devise explain formulate generate plan prepare rearrange reconstruct relate reorganize revise rewrite set up summarize synthesize tell write
Analysis
Break down objects or ideas into simpler parts and find evidence to support generalizations.
analyze appraise breakdown calculate categorize compare contrast criticize diagram differentiate discriminate distinguish examine experiment identify illustrate infer model outline point out question relate select separate subdivide test
Application
Apply knowledge to actual situations.
apply change choose compute demonstrate discover dramatize employ illustrate interpret manipulate modify operate practice predict prepare produce relate schedule show sketch solve use write
Comprehension
Demonstrate an understanding of the facts.
classify convert defend describe discuss distinguish estimate explain express extend generalized give example(s) identify indicate infer locate paraphrase predict recognize rewrite review select summarize translate
Knowledge
Remember previously learned information.
arrange define describe duplicate identify label list match memorize name order outline recognize relate recall repeat reproduce select state
Lower Order Thinking Skills

Learning Goals

- ✓ Understand the area of Business Rules Management
- ✓ Understand its relation to other BPM areas
- ✓ Understand key tasks in Business Rules Management
- ✓ Model and execute business rules evaluation using Camunda

Agenda

Motivation
Terminology
Rule Types
BRM Tool Examples
BPM, BRM & SOA
Business Rules Management (BRM) using JBoss Drools
Decision Model & Notation (DMN) 1.1
Integrated BPM/BRM using Signavio & Camunda

Motivation

Environment: Companies face rapidly changing market situation and customer demand.

Problem: with legacy systems, the process & business logic are buried deeply in the applications (often hardcoded)

- Processes / application workflows
- Business Rules

Thus, business depends on (often external) developers for system adaptions. Even with agile development, changes require time to production.

Solution approaches (possibly combined):

- Externalization of processes (Business Process Management)
- **Externalization of rules (Business Rules Management)**
- **Direct maintenance of processes and rules by the business departments!**

Terms and Definitions

Business Rule

- Rule which controls / influences the execution or the result of a business process
- Examples:
  - Rules for the approval of credit applications
  - Price building rules (e.g. insurance policies)
  - Rules for the automatic approval or denial of repair applications in the domain of automotive damages

Business Rules Management (BRM)

- Systematic management of business rules
- Important aspects: e.g. versioning, access rights management, …

Business Rule Management System (BRMS)

- A system which supports BRM

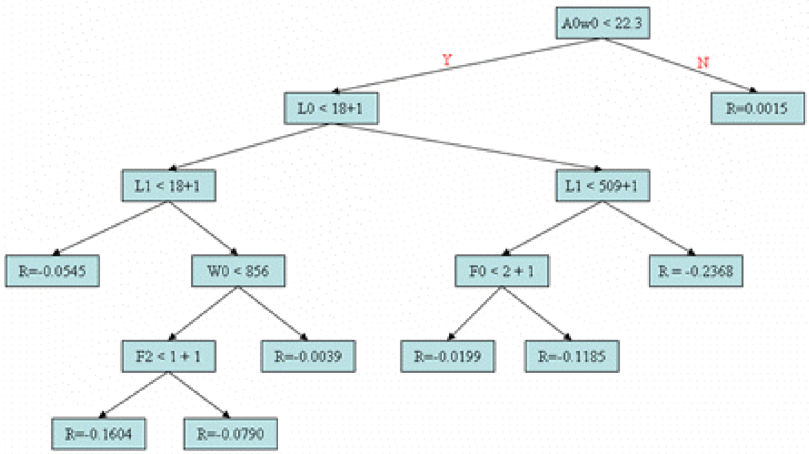
Business Rule Engine

- Core of a BRMS for the execution / evaluation of the rules

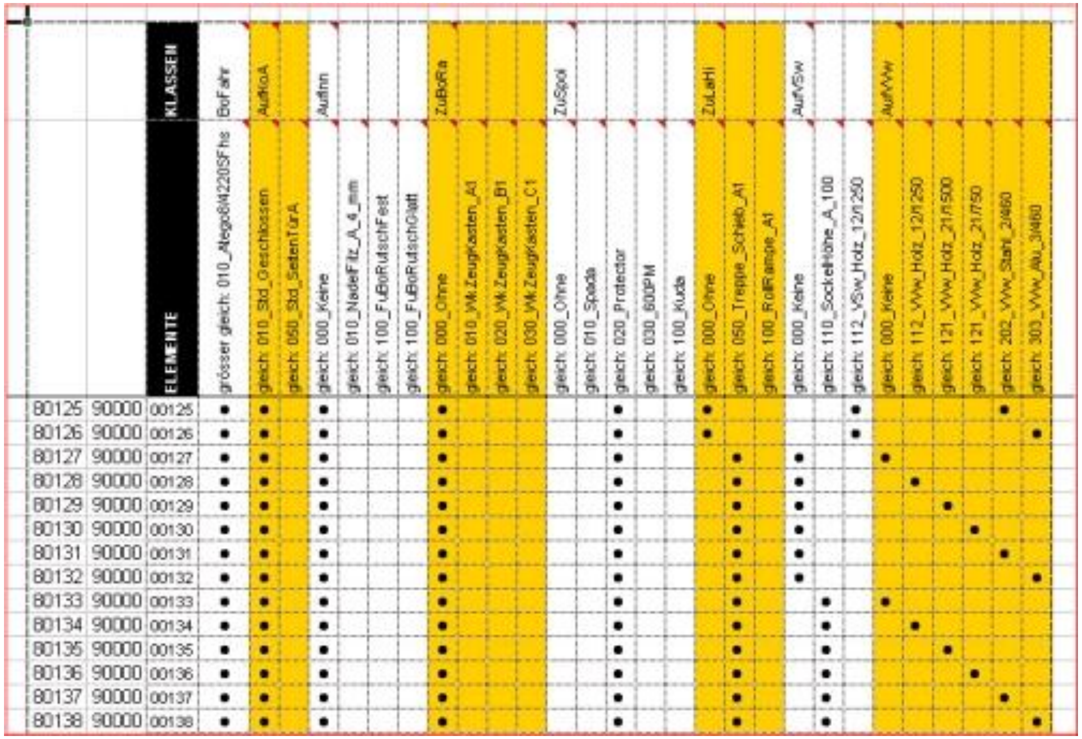
Business Rule Editor

- (Graphical) Editor for business rules

Rule Types



Logical "IF-THEN" rules
Decision Tables / Matrices
Decision Trees
Decision / Rules processes
Complex rule engines a la PROLOG



		KLASSEN	BoFahr	AufKoA		Aufnn				ZuBoRa				ZuSpoi					ZuLaHi			AufVSw			AufVw					
		ELEMENTE	grösser gleich: 010_Atego8/42205Fhs	gleich: 010_Std_Geschlossen	gleich: 050_Std_SeitenTürA	gleich: 000_Keine	gleich: 010_NadelFilz_A_4_mm	gleich: 100_FußBoRutschFest	gleich: 100_FußBoRutschGlatt	gleich: 000_Ohne	gleich: 010_WkZeugKasten_A1	gleich: 020_WkZeugKasten_B1	gleich: 030_WkZeugKasten_C1	gleich: 000_Ohne	gleich: 010_Spoila	gleich: 020_Protector	gleich: 030_600PM	gleich: 100_Kuda	gleich: 000_Ohne	gleich: 050_Treppe_Schieb_A1	gleich: 100_RollRampe_A1	gleich: 000_Keine	gleich: 110_SockelHöhe_A_100	gleich: 112_VSw_Holz_12/1250	gleich: 000_Keine	gleich: 112_Vw_Holz_12/1250	gleich: 121_Vw_Holz_21/1500	gleich: 121_Vw_Holz_21/750	gleich: 202_Vw_Stahl_2/460	gleich: 303_Vw_Alu_3/460
80125	90000	00125	●	●		●				●						●			●					●					●	
80126	90000	00126	●	●		●				●						●			●					●						●
80127	90000	00127	●	●		●				●						●				●		●			●					
80128	90000	00128	●	●		●				●						●				●		●				●				
80129	90000	00129	●	●		●				●						●				●		●					●			
80130	90000	00130	●	●		●				●						●				●		●						●		
80131	90000	00131	●	●		●				●						●				●		●							●	
80132	90000	00132	●	●		●				●						●				●		●								●
80133	90000	00133	●	●		●				●						●				●			●		●					
80134	90000	00134	●	●		●				●						●				●			●			●				
80135	90000	00135	●	●		●				●						●				●			●				●			
80136	90000	00136	●	●		●				●						●				●			●					●		
80137	90000	00137	●	●		●				●						●				●			●						●	
80138	90000	00138	●	●		●				●						●				●			●							●

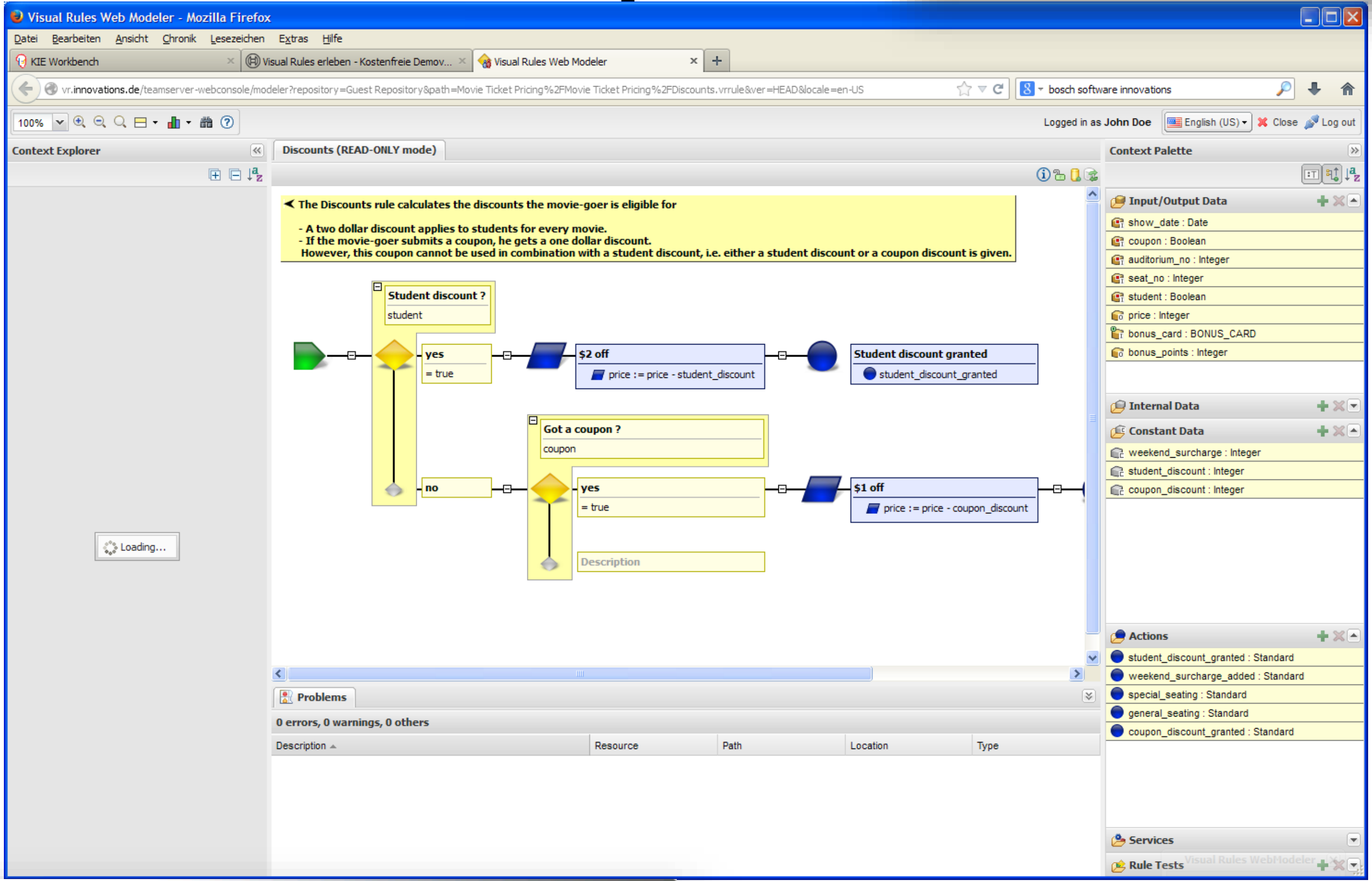
BRM Tool Examples

Automate business decisions and accelerate time-to-market

WebSphere. software
IBM

IBM WebSphere ILOG JRules BRMS

Business rule management for Java, mainframe and SOA environments



Visual Rules Web Modeler - Mozilla Firefox
Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe
KIE Workbench
Visual Rules erleben - Kostenfreie Demov...
Visual Rules Web Modeler
vr.innovations.de/teamserver-webconsole/modeler?repository=Guest Repository&path=Movie Ticket Pricing%2FMovie Ticket Pricing%2FDiscounts.vrrule&ver=HEAD&locale=en-US
bosch software innovations
100%
Logged in as John Doe
English (US)
Close
Log out
Context Explorer
Loading...
Discounts (READ-ONLY mode)
Context Palette

The Discounts rule calculates the discounts the movie-goer is eligible for

- **A two dollar discount applies to students for every movie.**
- **If the movie-goer submits a coupon, he gets a one dollar discount.**
However, this coupon cannot be used in combination with a student discount, i.e. either a student discount or a coupon discount is given.

Student discount ?
student
yes
= true
$2 off
price := price - student_discount
Student discount granted
student_discount_granted
no
Got a coupon ?
coupon
yes
= true
$1 off
price := price - coupon_discount
Description

Problems

0 errors, 0 warnings, 0 others

Description	Resource	Path	Location	Type

Input/Output Data
- show_date : Date
- coupon : Boolean
- auditorium_no : Integer
- seat_no : Integer
- student : Boolean
- price : Integer
- bonus_card : BONUS_CARD
- bonus_points : Integer

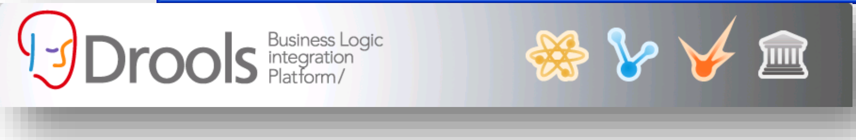
Internal Data

Constant Data
- weekend_surcharge : Integer
- student_discount : Integer
- coupon_discount : Integer

Actions
- student_discount_granted : Standard
- weekend_surcharge_added : Standard
- special_seating : Standard
- general_seating : Standard
- coupon_discount_granted : Standard

Services

Rule Tests

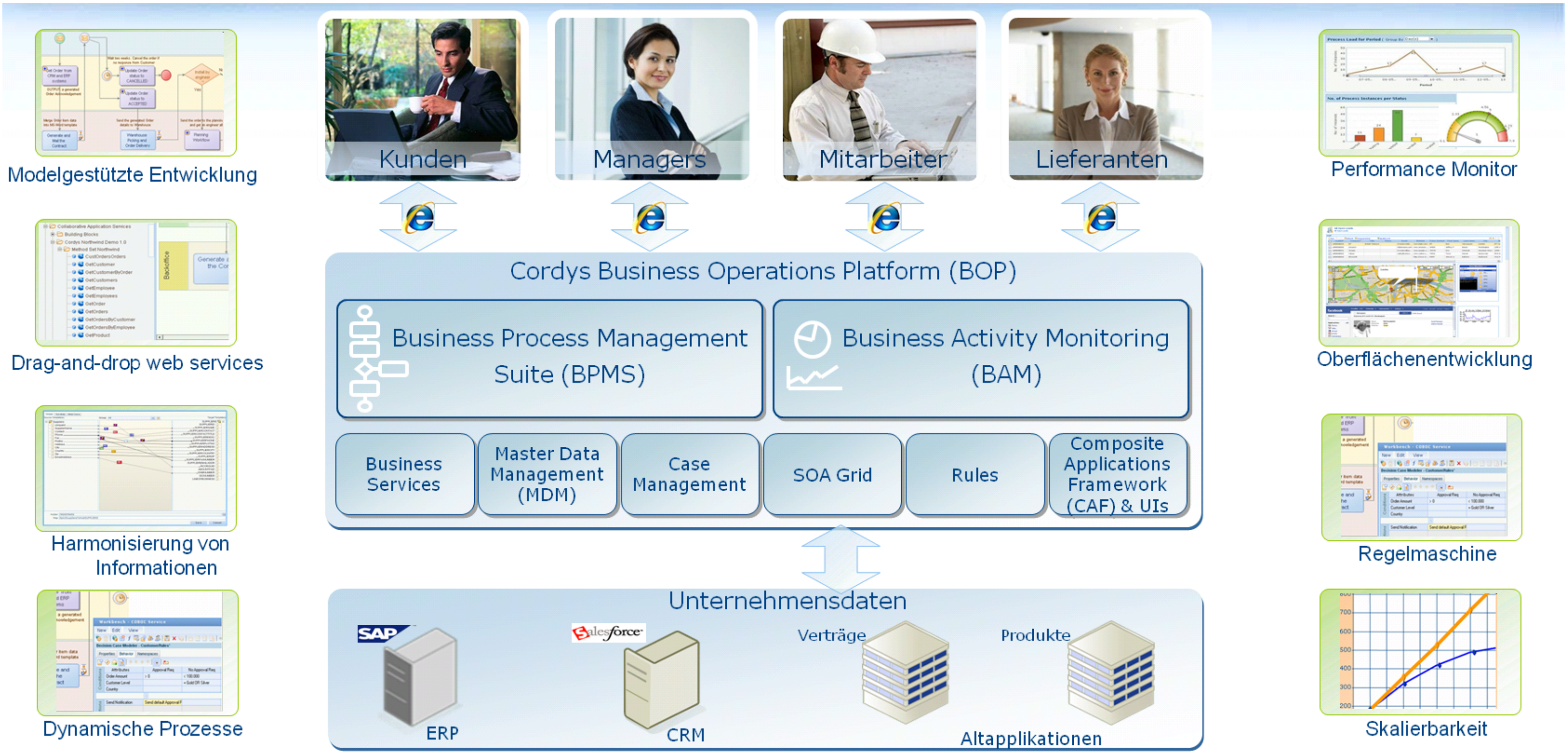


BPM, BRM and SOA

… complement each other in an optimal way

Business Rules can be integrated in Business Processes
- Directly in the busines process model
- As external service calls
- Other ways of integration: Enterprise Application Integration, APIs, …

Integrated Tool Example: Cordys Platform



Modelgestützte Entwicklung

Drag-and-drop web services

Harmonisierung von Informationen

Dynamische Prozesse

Kunden

Managers

Mitarbeiter

Lieferanten

Cordys Business Operations Platform (BOP)

Business Process Management Suite (BPMS)

Business Activity Monitoring (BAM)

Business Services

Master Data Management (MDM)

Case Management

SOA Grid

Rules

Composite Applications Framework (CAF) & UIs

Unternehmensdaten

SAP

Salesforce

Verträge

Produkte

ERP

CRM

Altapplikationen

Performance Monitor

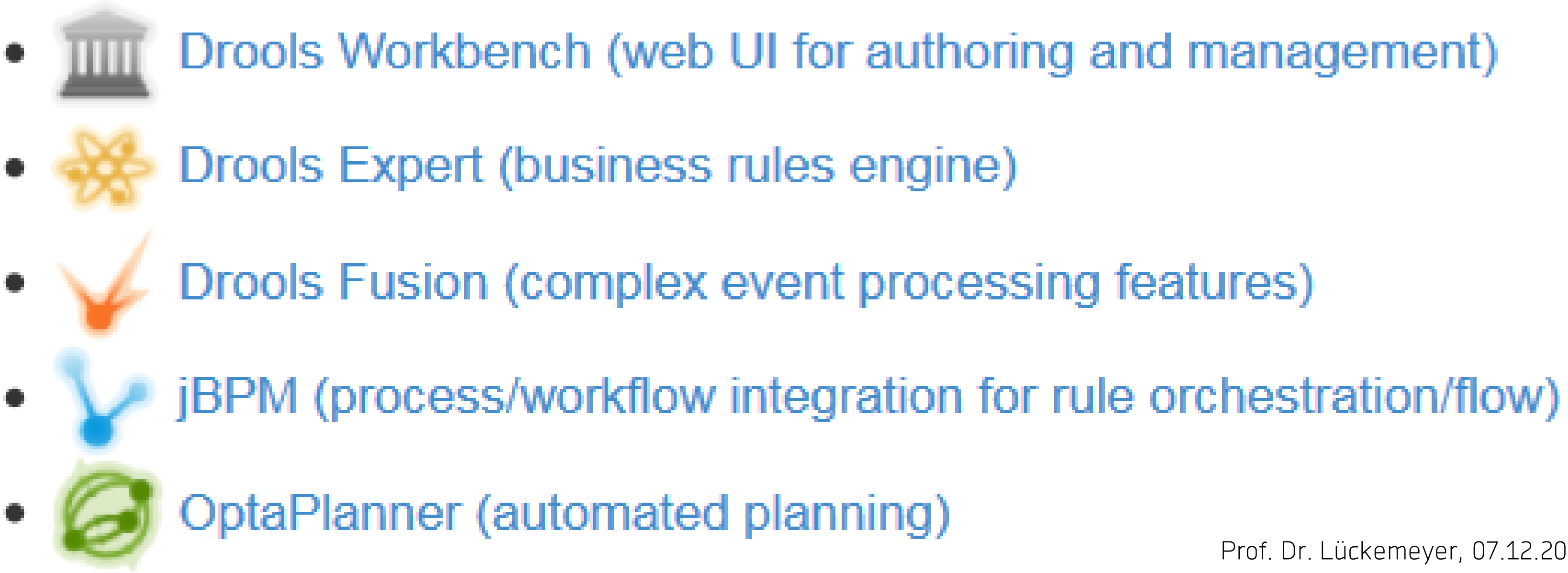
Oberflächenentwicklung

Regelmaschine

Skalierbarkeit

BRM with JBoss Drools

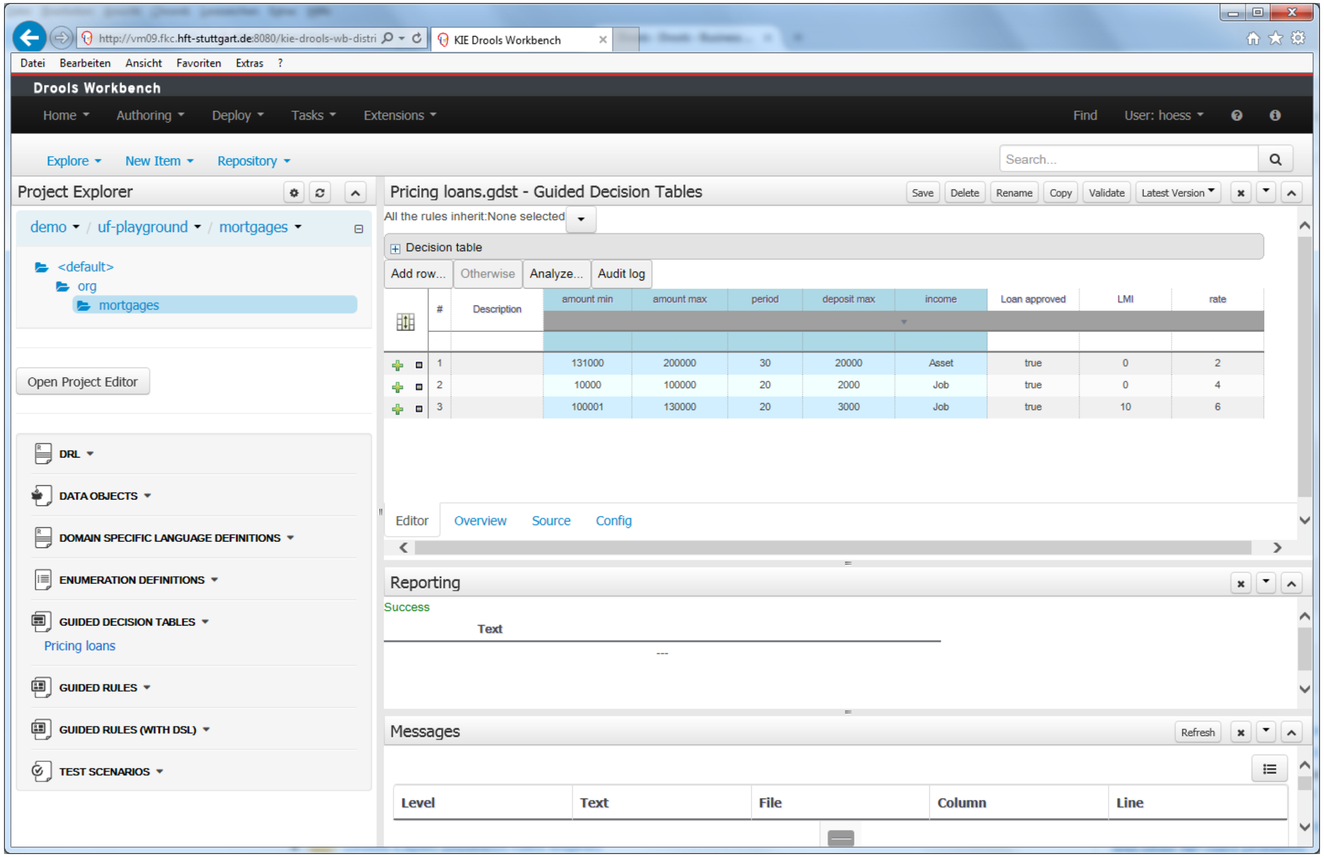
Open Source BRMS (JBoss)
Part of the jBPM Business Logic integration Platform
Tool- and API-based access (also from Java / Eclipse)
5 Sub-projects / Components



- Drools Workbench (web UI for authoring and management)
- Drools Expert (business rules engine)
- Drools Fusion (complex event processing features)
- jBPM (process/workflow integration for rule orchestration/flow)
- OptaPlanner (automated planning)

- Drools Workbench (web UI for authoring and management)
- Drools Expert (business rules engine)
- Drools Fusion (complex event processing features)
- jBPM (process/workflow integration for rule orchestration/flow)
- OptaPlanner (automated planning)

JBoss Drools Workbench: Web-based BRMS



Pricing loans.gdst - Guided Decision Tables

All the rules inherit:None selected

#	Description	amount min	amount max	period	deposit max	income	Loan approved	LMI	rate
1		131000	200000	30	20000	Asset	true	0	2
2		10000	100000	20	2000	Job	true	0	4
3		100001	130000	20	3000	Job	true	10	6

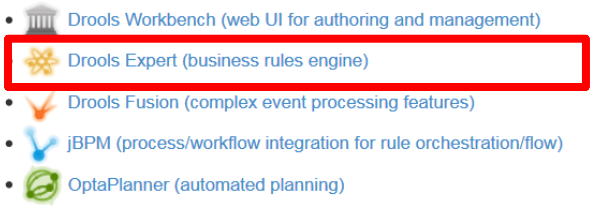
Reporting

Success

Text

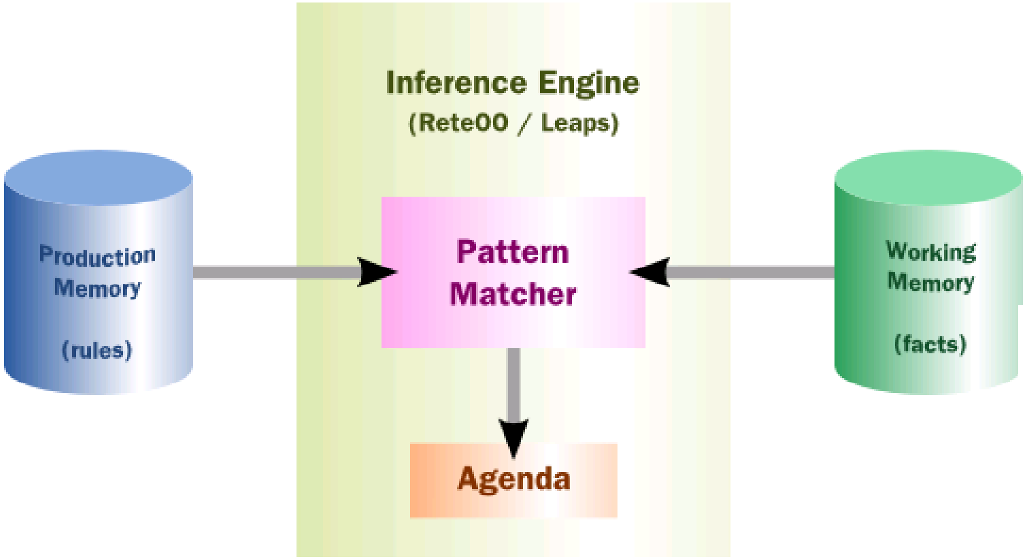
Messages

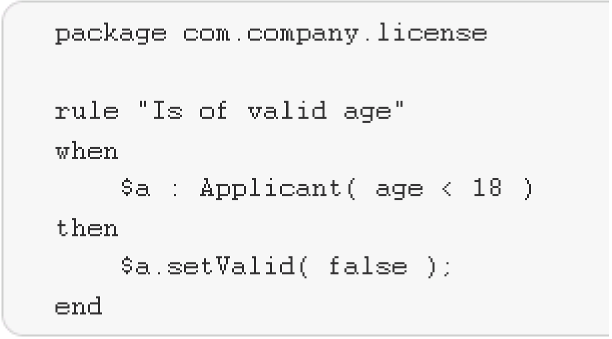
Level	Text	File	Column	Line



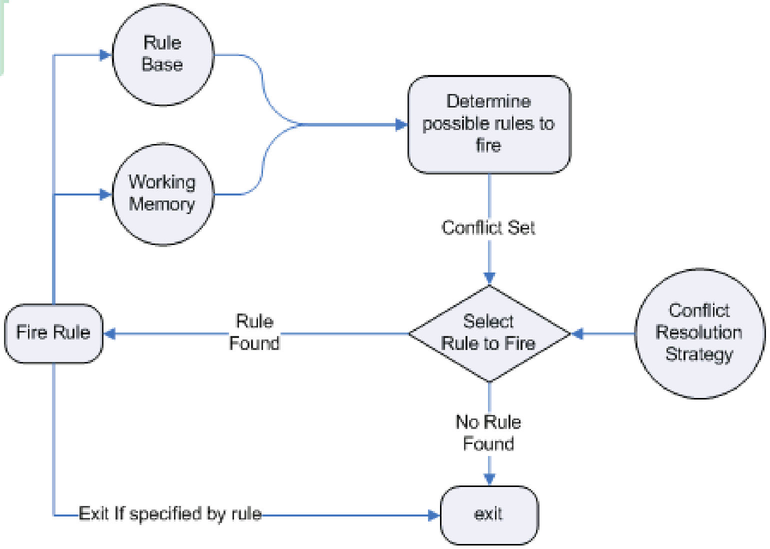
- Drools Workbench (web UI for authoring and management)
- Drools Expert (business rules engine)
- Drools Fusion (complex event processing features)
- jBPM (process/workflow integration for rule orchestration/flow)
- OptaPlanner (automated planning)

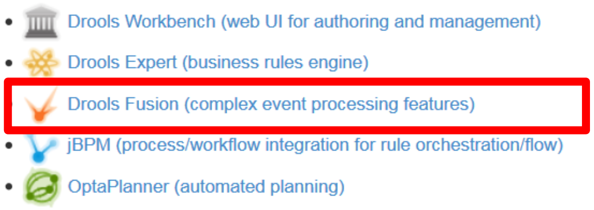
JBoss Drools Expert: Core Rule Engine





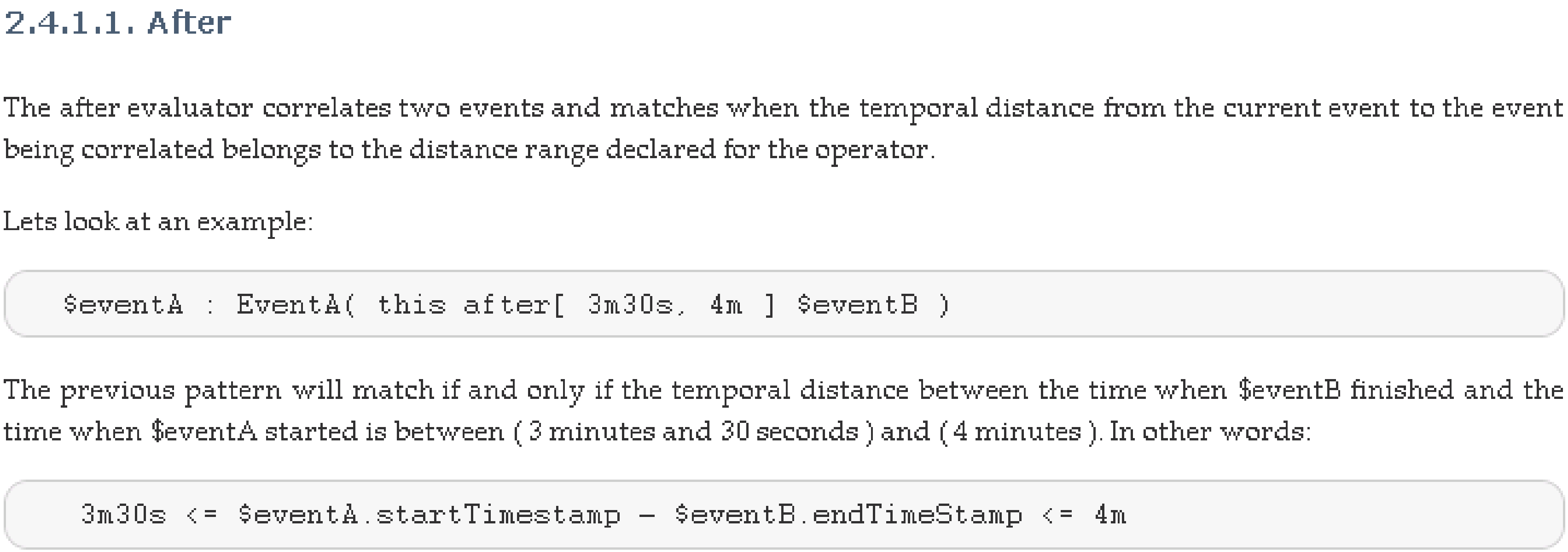
```
package com.company.license

rule "Is of valid age"
when
    $a : Applicant( age < 18 )
then
    $a.setValid( false );
end
```





- Drools Workbench (web UI for authoring and management)
- Drools Expert (business rules engine)
- Drools Fusion (complex event processing features)
- jBPM (process/workflow integration for rule orchestration/flow)
- OptaPlanner (automated planning)

JBoss Drools Fusion: Complex Event Processing (CEP)



2.4.1.1. After

The after evaluator correlates two events and matches when the temporal distance from the current event to the event being correlated belongs to the distance range declared for the operator.

Lets look at an example:

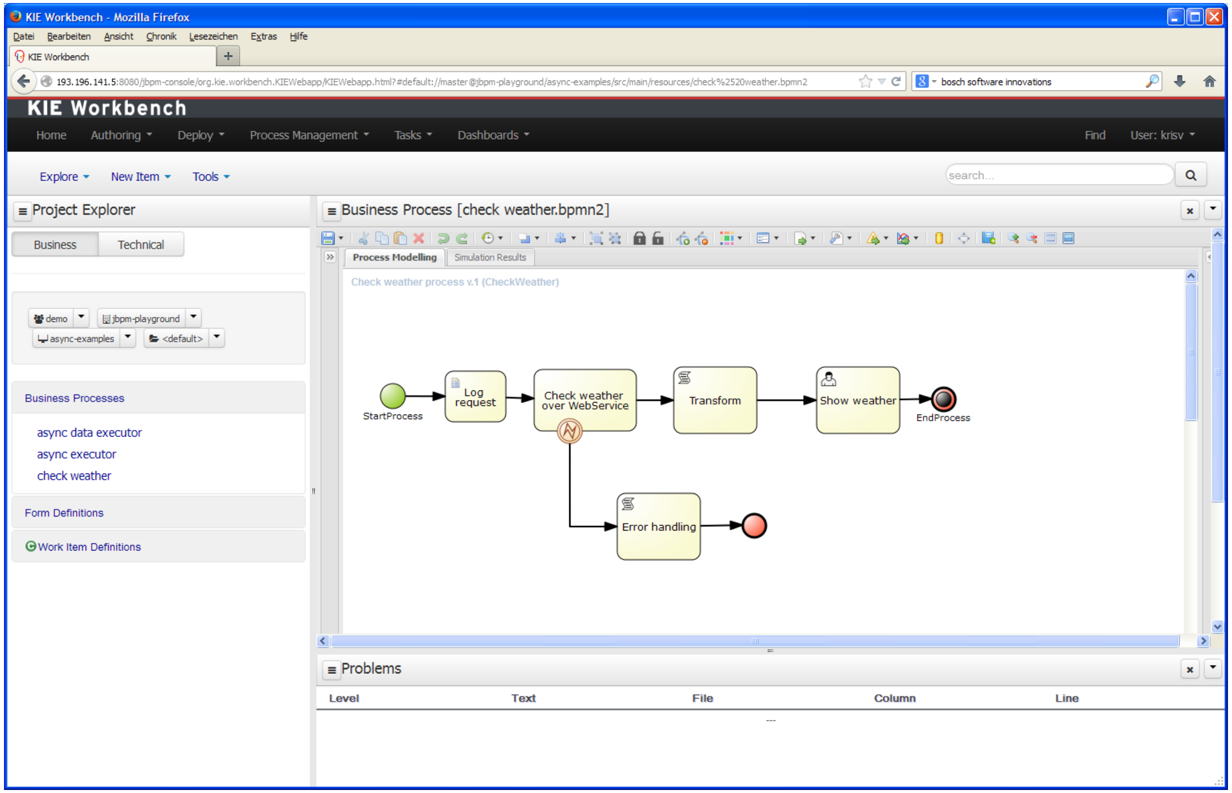
```
$eventA : EventA( this after[ 3m30s, 4m ] $eventB )
```

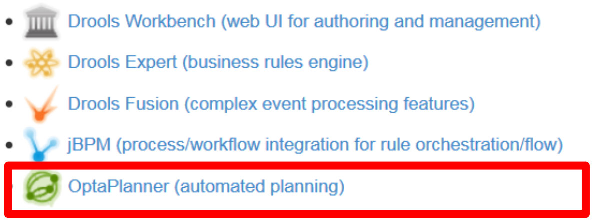
The previous pattern will match if and only if the temporal distance between the time when $eventB finished and the time when $eventA started is between (3 minutes and 30 seconds) and (4 minutes). In other words:

```
 3m30s <= $eventA.startTimestamp - $eventB.endTimeStamp <= 4m
```

- Drools Workbench (web UI for authoring and management)
- Drools Expert (business rules engine)
- Drools Fusion (complex event processing features)
- jBPM (process/workflow integration for rule orchestration/flow)
- OptaPlanner (automated planning)

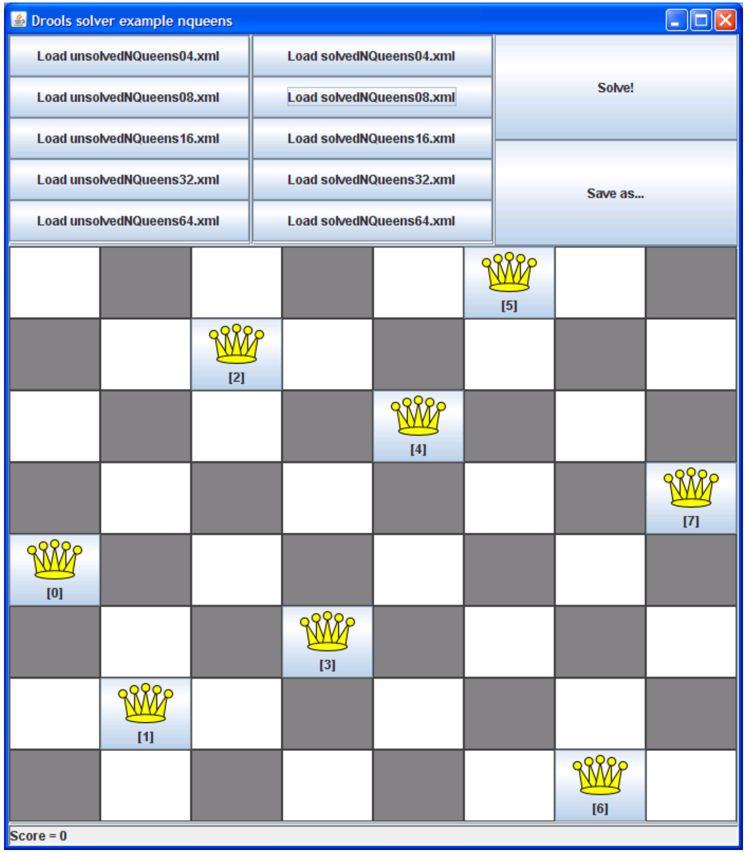
JBoss Drools jBPM: Web-Designer, technical basis like Signavio





- Drools Workbench (web UI for authoring and management)
- Drools Expert (business rules engine)
- Drools Fusion (complex event processing features)
- jBPM (process/workflow integration for rule orchestration/flow)
- OptaPlanner (automated planning)

JBoss Drools Opta Planner: Problem Solver Engine

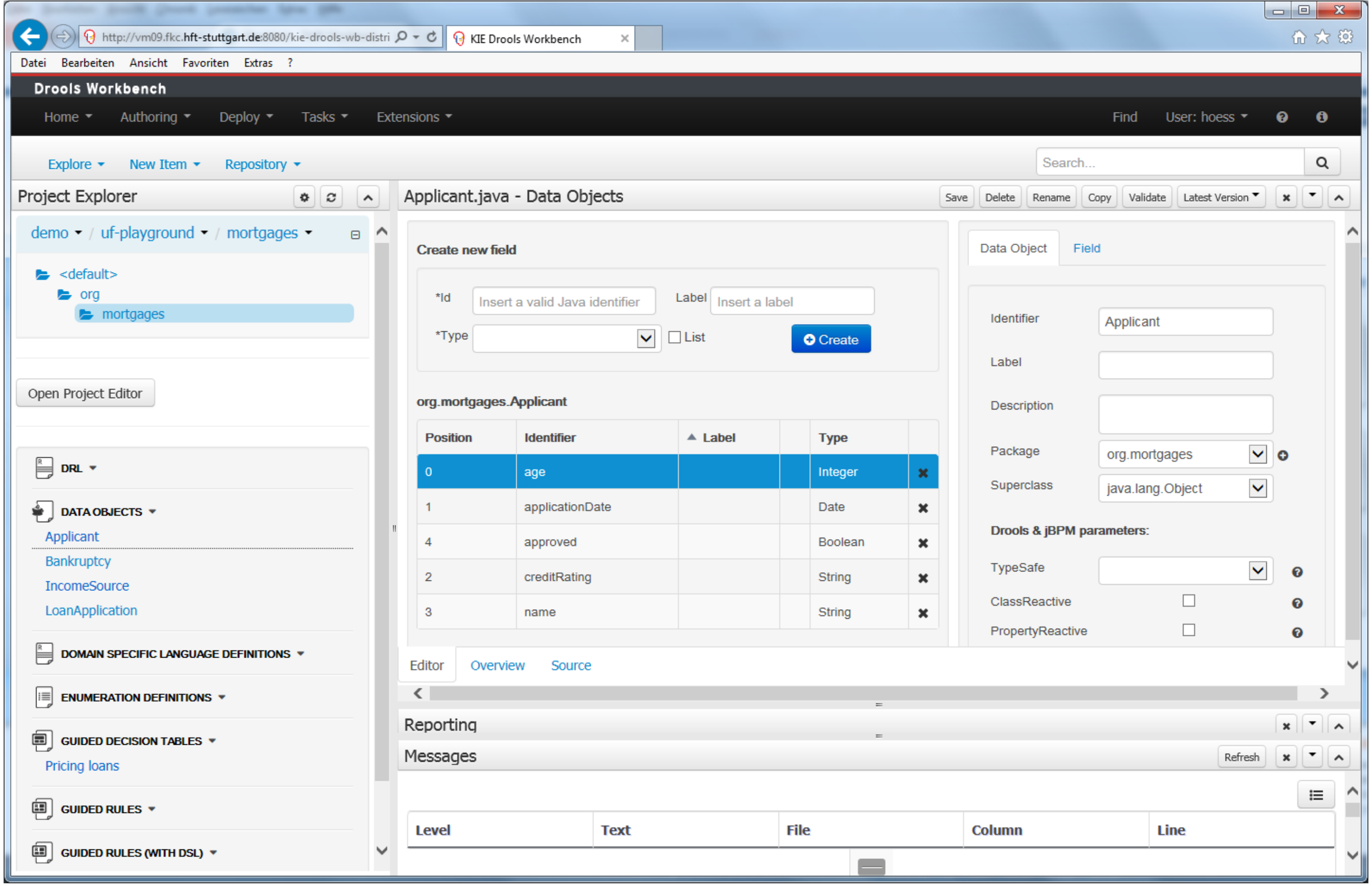


Score = 0

Planning Engine for solving of complex problems

- routing problems (travelling salesman)
- disposition of service technicians
- Schedule planning
- Resource planning

Drools Example - Basic Model



Applicant.java - Data Objects

Create new field

*Id: Insert a valid Java identifier
Label: Insert a label

org.mortgages.Applicant

Position	Identifier	Label	Type
0	age		Integer
1	applicationDate		Date
4	approved		Boolean
2	creditRating		String
3	name		String

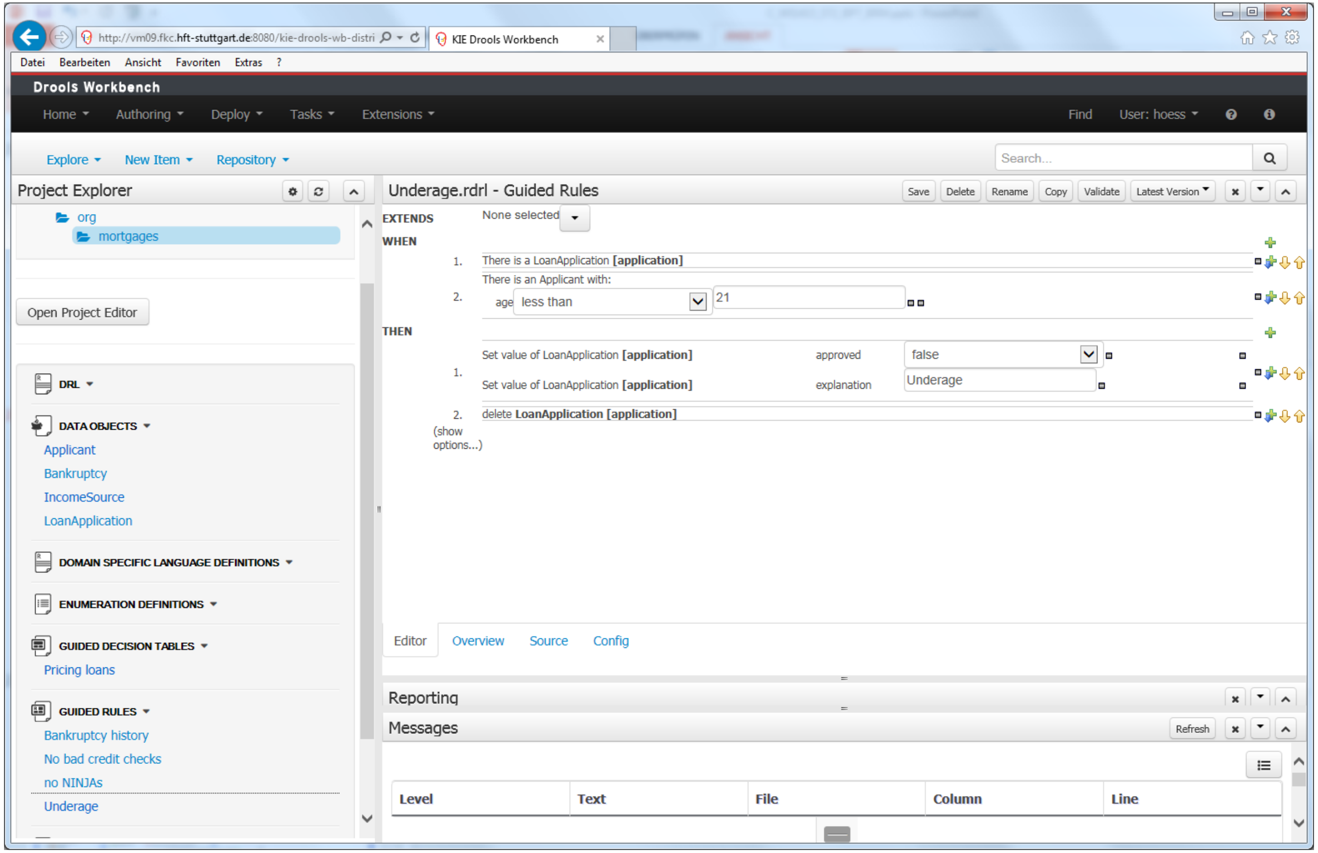
Identifier: Applicant
Label:
Description:
Package: org.mortgages
Superclass: java.lang.Object

Drools & jBPM parameters:

TypeSafe:
ClassReactive □
PropertyReactive □

Level	Text	File	Column	Line

Drools Example - Rule: age checking for loan applications



Underage.rdrl - Guided Rules

EXTENDS None selected

WHEN

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

2. There is an Applicant with:
age less than 21

THEN

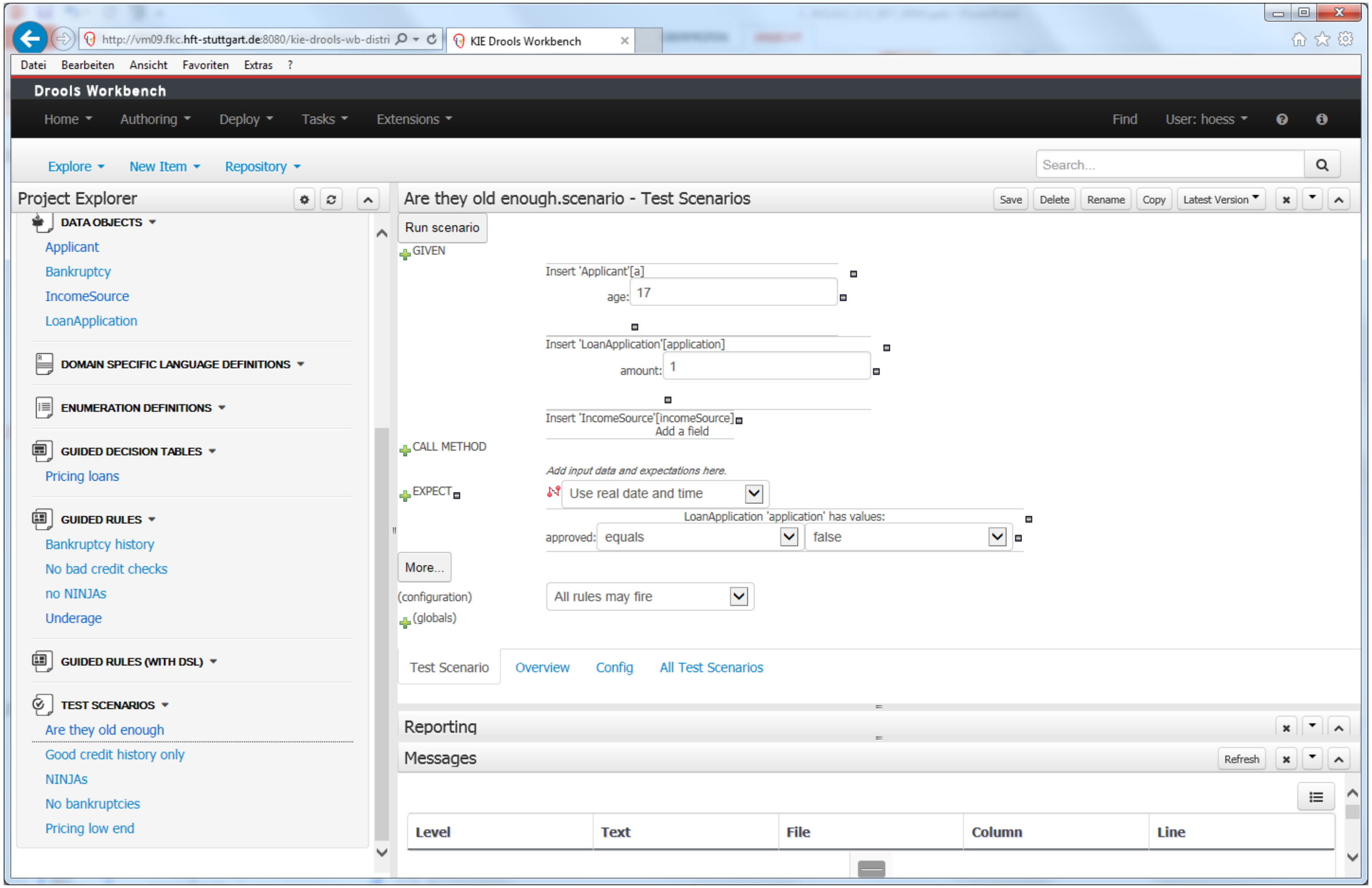
1. Set value of LoanApplication **[application]** approved false
Set value of LoanApplication **[application]** explanation Underage

2. delete **LoanApplication [application]**

(show options...)

Level	Text	File	Column	Line

Drools Example – Test Scenario



Are they old enough.scenario - Test Scenarios

GIVEN

Insert 'Applicant'[a]

age: 17

Insert 'LoanApplication'[application]

amount: 1

Insert 'IncomeSource'[incomeSource]
Add a field

CALL METHOD

Add input data and expectations here.

EXPECT

Use real date and time

LoanApplication 'application' has values:

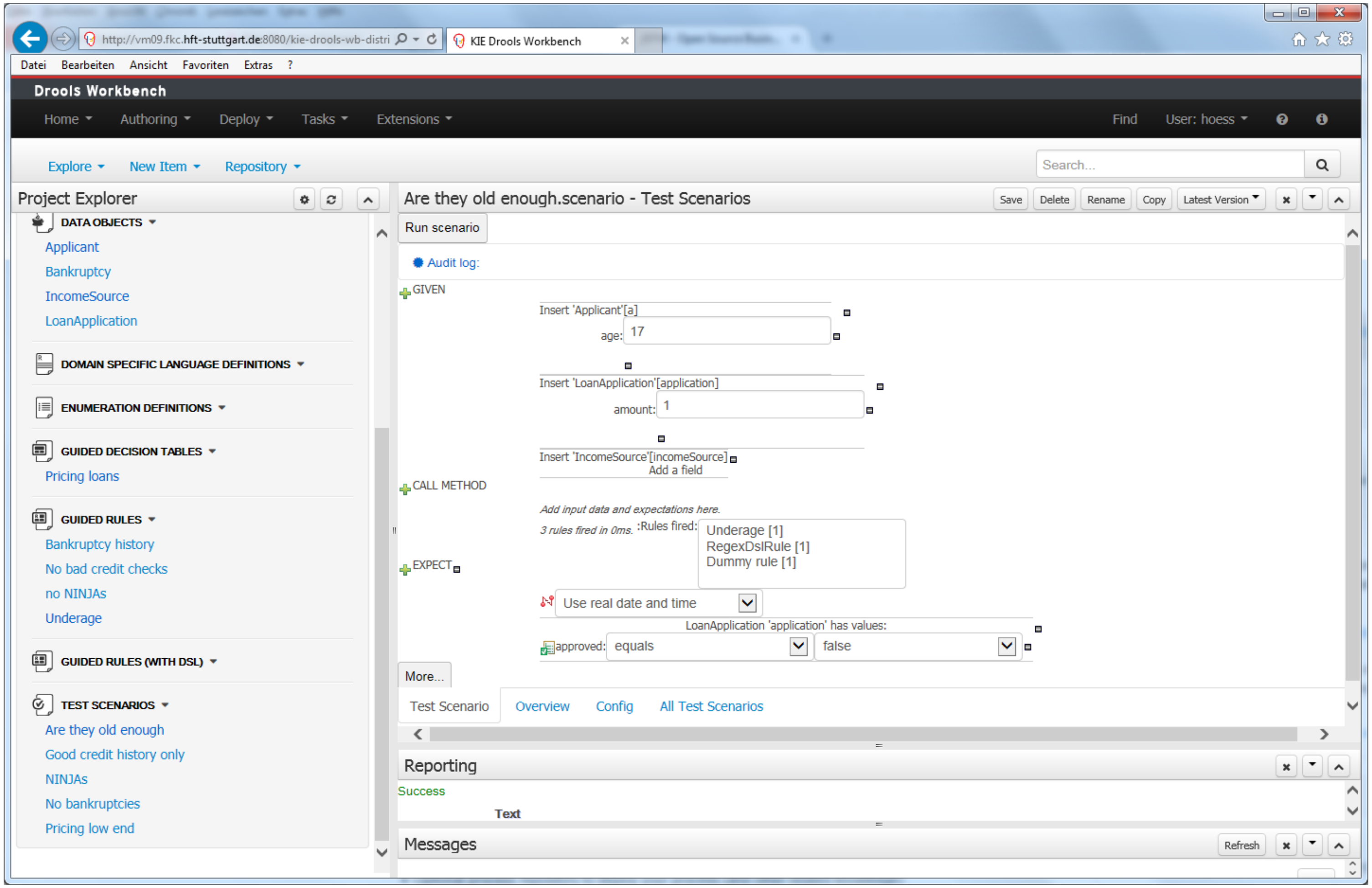
approved: equals false

(configuration) All rules may fire

(globals)

Level	Text	File	Column	Line

Drools Example – Test Result



Insert 'Applicant'[a]

age: 17

Insert 'LoanApplication'[application]

amount: 1

Insert 'IncomeSource'[incomeSource]

Add input data and expectations here.

3 rules fired in 0ms. :Rules fired: Underage [1]
RegexDslRule [1]
Dummy rule [1]

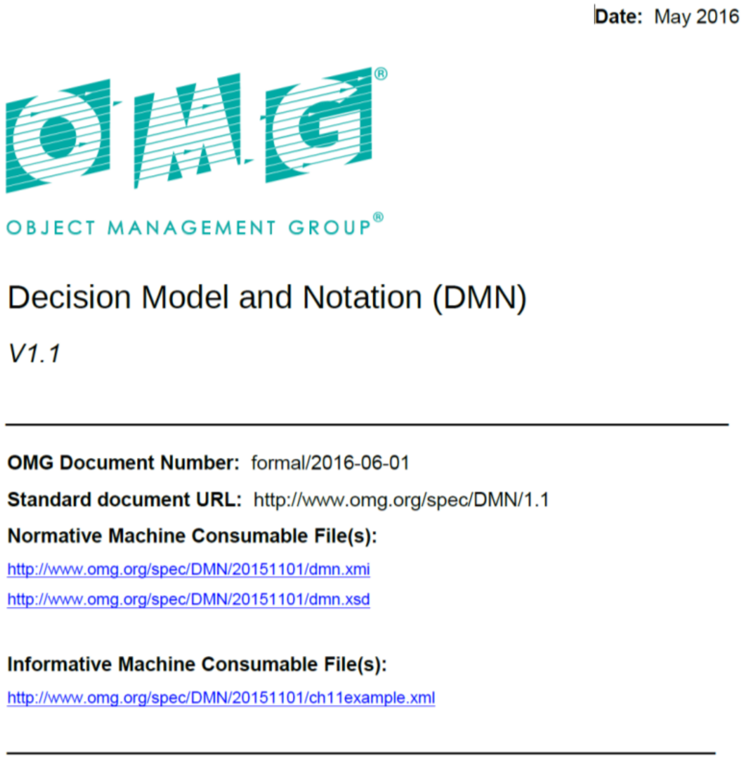
Use real date and time

LoanApplication 'application' has values:

approved: equals false

Success

Yet another OMG-Standard: DMN- Decision Model and Notation



Date: May 2016

Decision Model and Notation (DMN)

V1.1

OMG Document Number: formal/2016-06-01

Standard document URL: http://www.omg.org/spec/DMN/1.1

Normative Machine Consumable File(s):

http://www.omg.org/spec/DMN/20151101/dmn.xmi

http://www.omg.org/spec/DMN/20151101/dmn.xsd

Informative Machine Consumable File(s):

http://www.omg.org/spec/DMN/20151101/ch11example.xml

Contributors

Copyright © 2013, Decision Management Solutions
Copyright © 2013, Escape Velocity LLC
Copyright © 2013, Fair Isaac Corporation
Copyright © 2013, International Business Machines Corporation
Copyright © 2013, Knowledge Partners International
Copyright © 2013, KU Leuven
Copyright © 2013, Model Systems Limited
Copyright © 2013, Oracle Incorporated
Copyright © 2013, TIBCO Software Inc.
Copyright © 2016, Object Management Group, Inc.

DMN 1.1 Symbols: Requirements

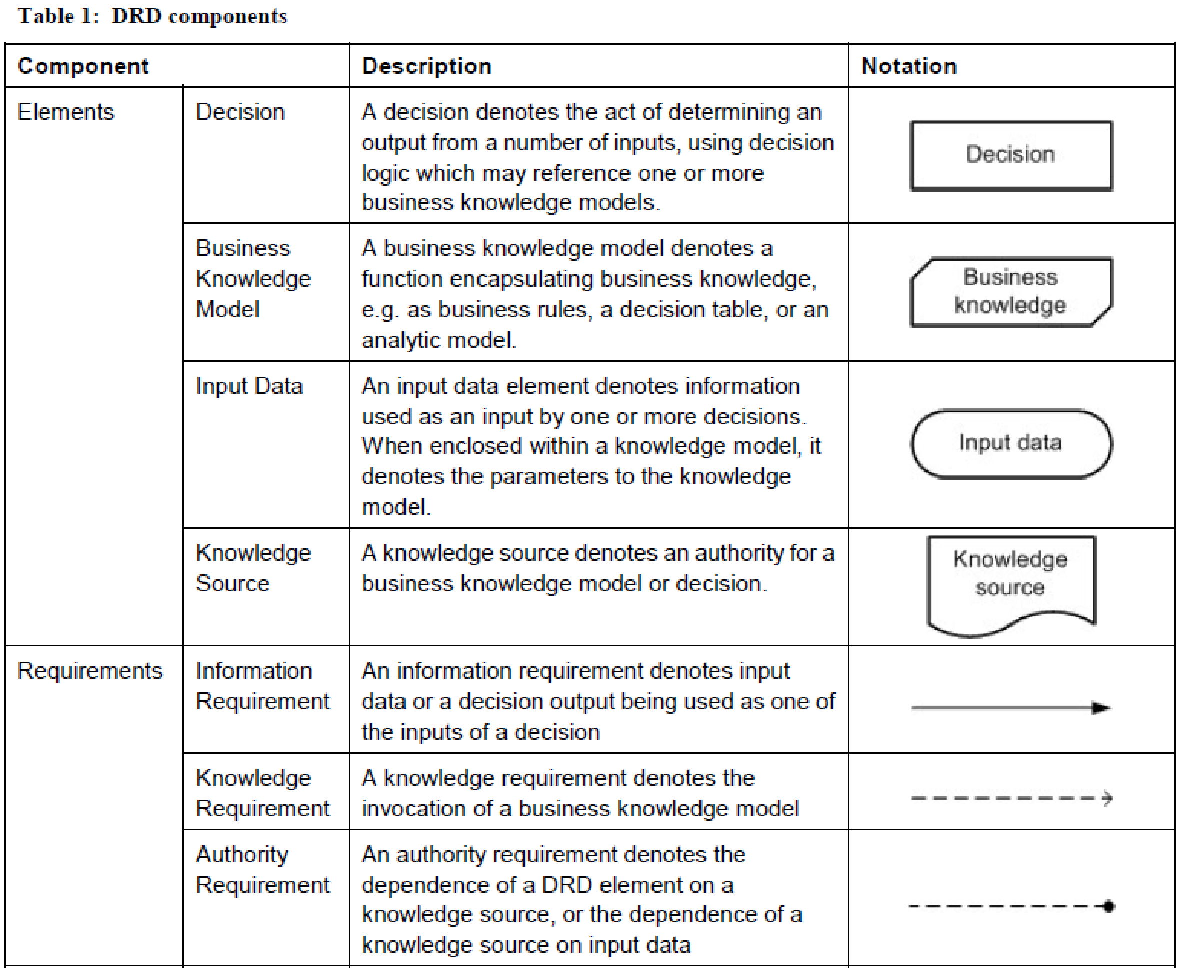


Table 1: DRD components

Component		Description	Notation
Elements	Decision	A decision denotes the act of determining an output from a number of inputs, using decision logic which may reference one or more business knowledge models.	Decision
	Business Knowledge Model	A business knowledge model denotes a function encapsulating business knowledge, e.g. as business rules, a decision table, or an analytic model.	Business knowledge
	Input Data	An input data element denotes information used as an input by one or more decisions. When enclosed within a knowledge model, it denotes the parameters to the knowledge model.	Input data
	Knowledge Source	A knowledge source denotes an authority for a business knowledge model or decision.	Knowledge source
Requirements	Information Requirement	An information requirement denotes input data or a decision output being used as one of the inputs of a decision	───────►
	Knowledge Requirement	A knowledge requirement denotes the invocation of a business knowledge model	- - - - - - - - ->
	Authority Requirement	An authority requirement denotes the dependence of a DRD element on a knowledge source, or the dependence of a knowledge source on input data	- - - - - - - - - -●

DMN: Modeling Decisions and related Data

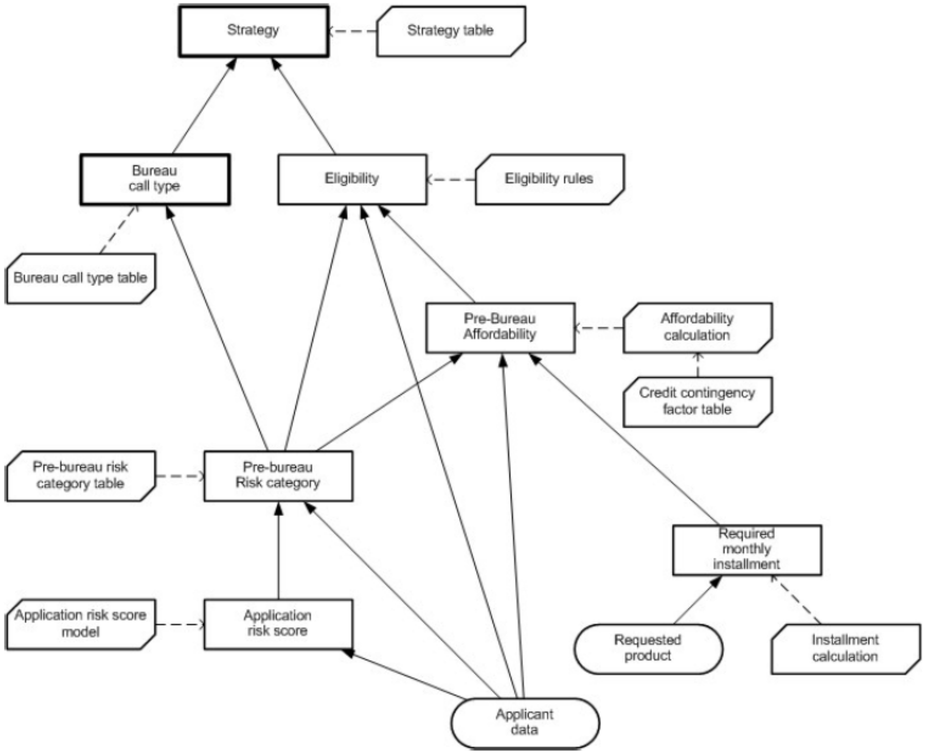
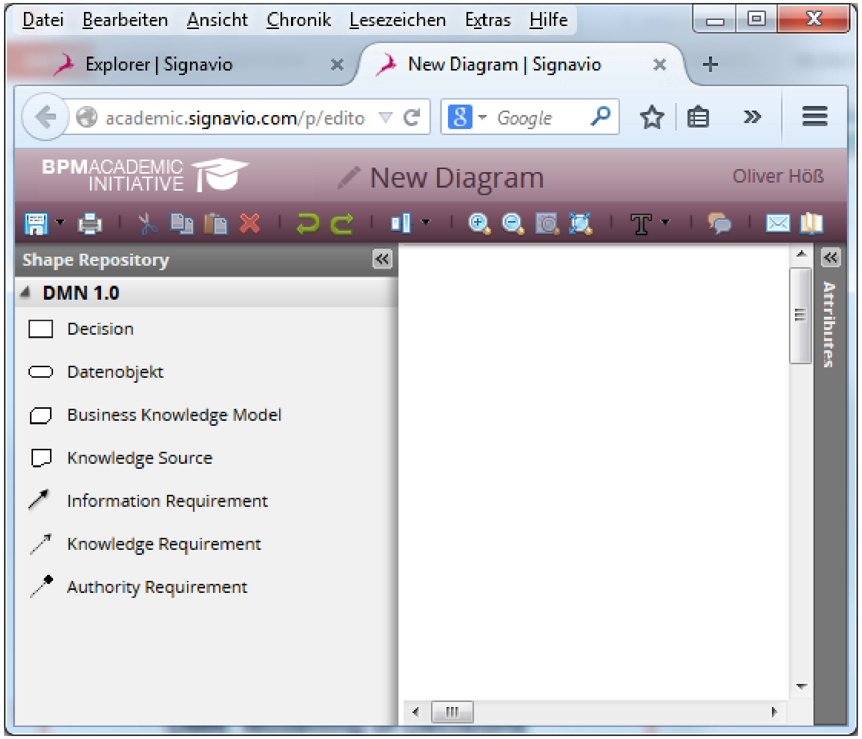
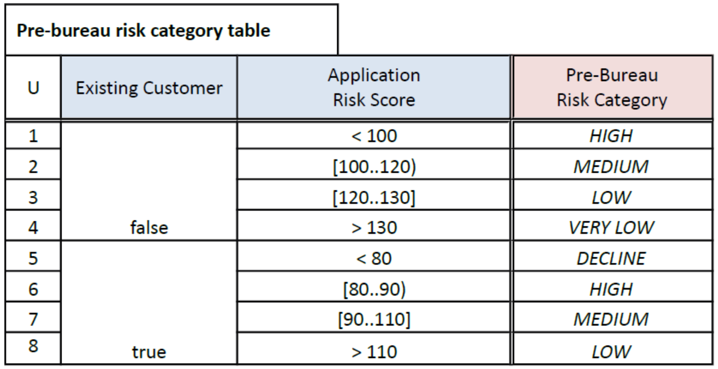


Figure 56: DRD for Bureau Strategy decision point



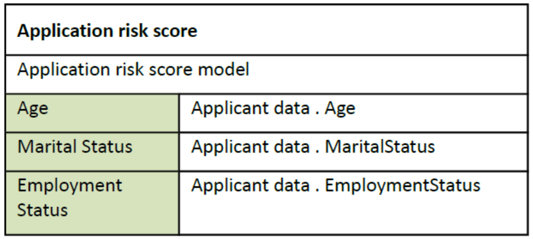
DMN: modeling the decision logic



Pre-bureau risk category table

U	Existing Customer	Application Risk Score	Pre-Bureau Risk Category
1	false	< 100	*HIGH*
2		[100..120)	*MEDIUM*
3		[120..130]	*LOW*
4		> 130	*VERY LOW*
5	true	< 80	*DECLINE*
6		[80..90)	*HIGH*
7		[90..110]	*MEDIUM*
8		> 110	*LOW*

Figure 82: Pre-bureau risk category table decision logic

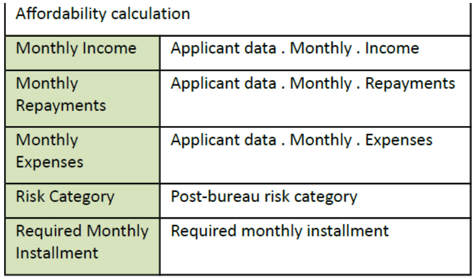


Application risk score

Application risk score model

Age	Applicant data . Age
Marital Status	Applicant data . MaritalStatus
Employment Status	Applicant data . EmploymentStatus

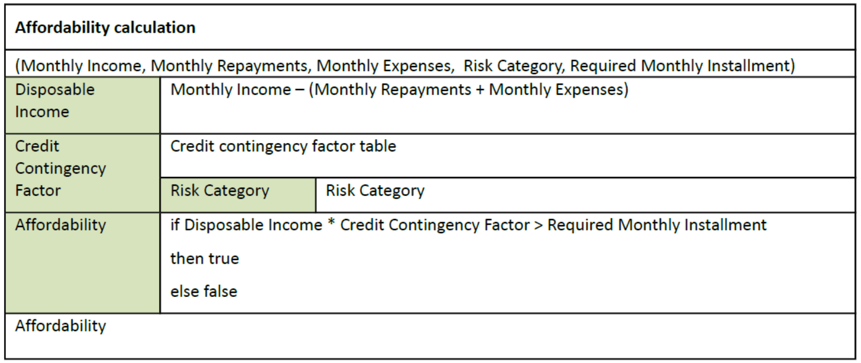
Figure 83: Application Risk Score decision logic



Affordability calculation

Monthly Income	Applicant data . Monthly . Income
Monthly Repayments	Applicant data . Monthly . Repayments
Monthly Expenses	Applicant data . Monthly . Expenses
Risk Category	Post-bureau risk category
Required Monthly Installment	Required monthly installment

Figure 90: Post-Bureau Affordability decision logic



Affordability calculation

(Monthly Income, Monthly Repayments, Monthly Expenses, Risk Category, Required Monthly Installment)

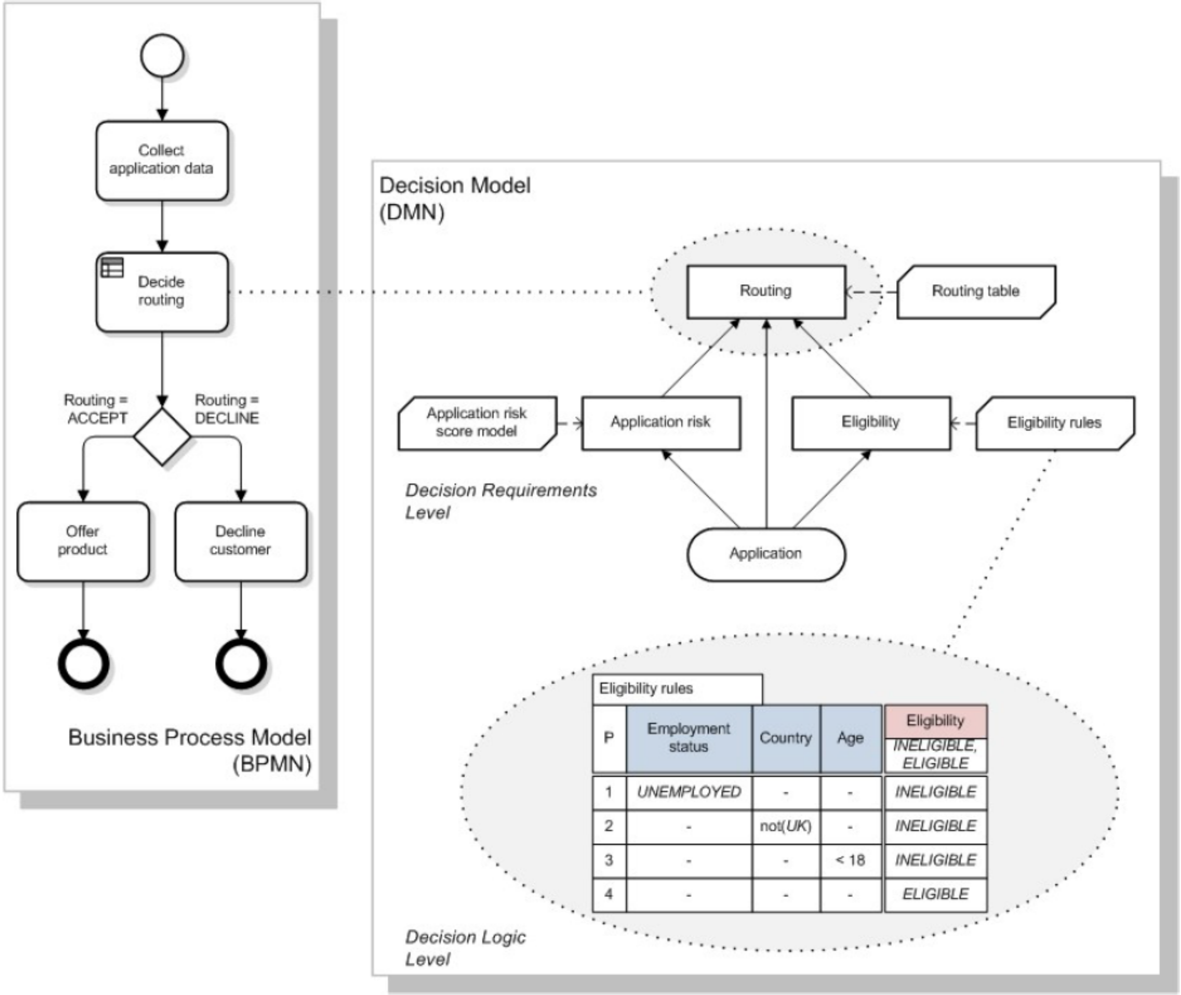
Disposable Income	Monthly Income – (Monthly Repayments + Monthly Expenses)	
Credit Contingency Factor	Credit contingency factor table	
	Risk Category	Risk Category

Affordability	if Disposable Income * Credit Contingency Factor > Required Monthly Installment
	then true
	else false	

Affordability

Figure 91: Affordability calculation decision logic

Relation BPMN <-> DMN

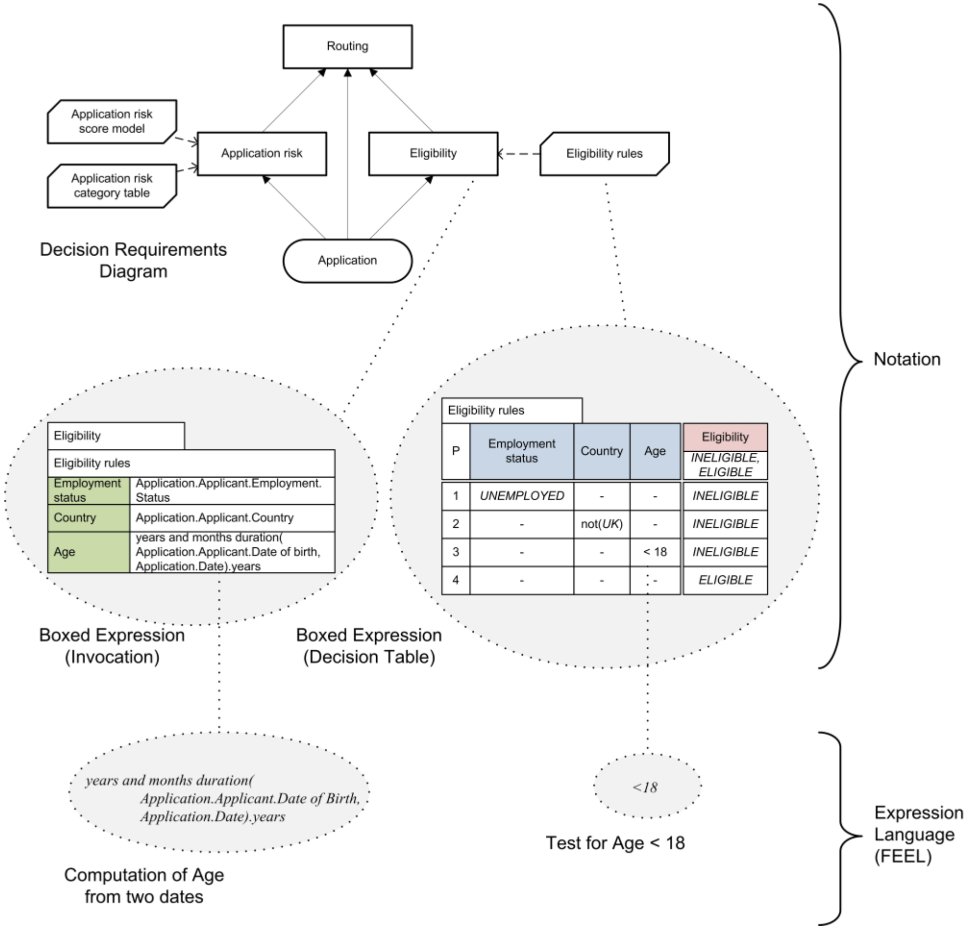


Eligibility rules

P	Employment status	Country	Age	Eligibility
				INELIGIBLE, ELIGIBLE
1	UNEMPLOYED	-	-	INELIGIBLE
2	-	not(UK)	-	INELIGIBLE
3	-	-	< 18	INELIGIBLE
4	-	-	-	ELIGIBLE

Figure 1: Aspects of modeling

DMN: limitations


Routing
Application risk score model
Application risk category table
Application risk
Eligibility
Eligibility rules
Application
Decision Requirements Diagram
Boxed Expression (Invocation)
Boxed Expression (Decision Table)
Notation
years and months duration(Application.Applicant.Date of Birth, Application.Date).years
<18
Computation of Age from two dates
Test for Age < 18
Expression Language (FEEL)

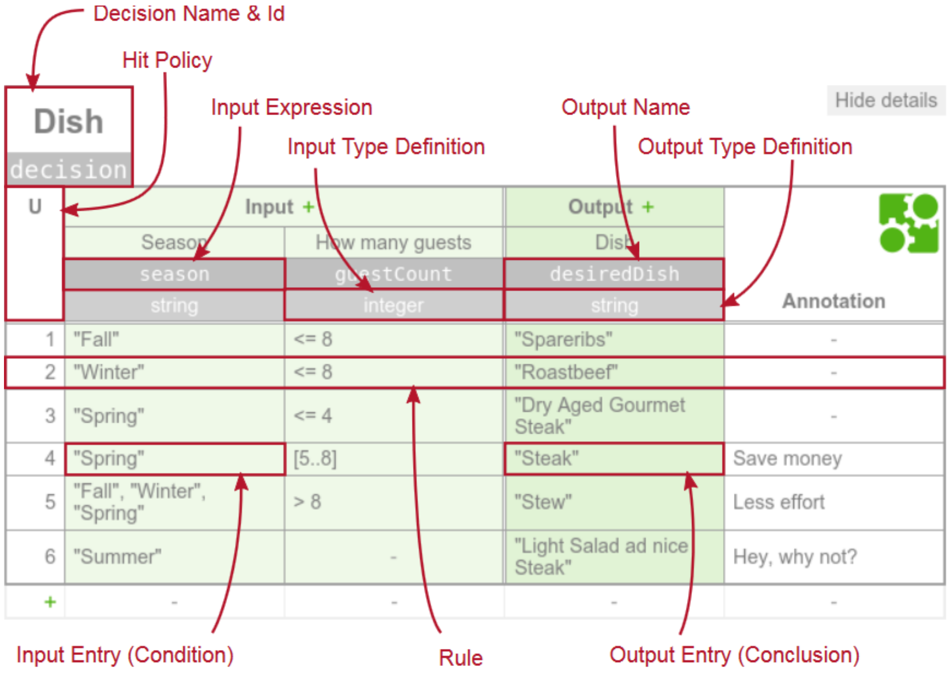
Eligibility	
Eligibility rules	
Employment status	Application.Applicant.Employment. Status
Country	Application.Applicant.Country
Age	years and months duration(Application.Applicant.Date of birth, Application.Date).years

Eligibility rules

P	Employment status	Country	Age	Eligibility *INELIGIBLE, ELIGIBLE*
1	*UNEMPLOYED*	-	-	*INELIGIBLE*
2	-	not(*UK*)	-	*INELIGIBLE*
3	-	-	< 18	*INELIGIBLE*
4	-	-	-	*ELIGIBLE*

Grey ellipses and dotted lines only to indicate correspondences between concepts in different levels. They do *not* form part of the notation of **DMN**. Future implementations may provide facilities for moving between levels of modeling, such as "opening," "drilling down," or "zooming in" but **DMN** does not specify how. At the decision logic level, every decision in a DRG is defined using a **value expression** which specifies how the decision's output is determined from its inputs. At that level, the decision is considered to *be* the evaluation of the expression. The value expression may be notated using a **boxed expression**.

Modeling DMN in Camunda: decision tables

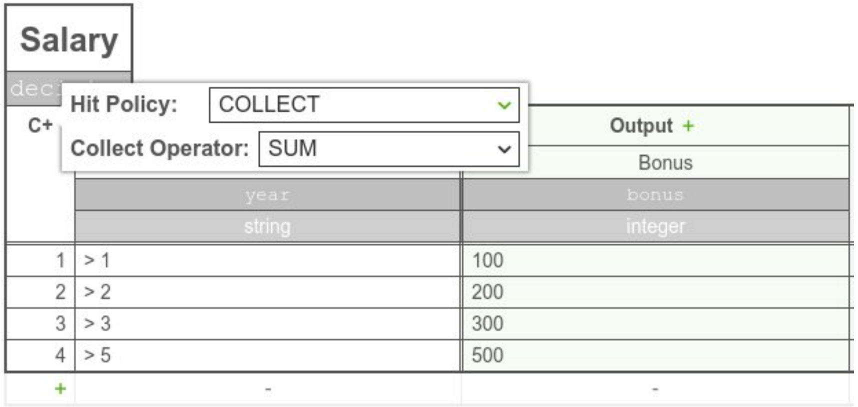


Dish decision	Input		Output	
U	Season season string	How many guests guestCount integer	Dish desiredDish string	Annotation
1	"Fall"	<= 8	"Spareribs"	-
2	"Winter"	<= 8	"Roastbeef"	-
3	"Spring"	<= 4	"Dry Aged Gourmet Steak"	-
4	"Spring"	[5..8]	"Steak"	Save money
5	"Fall", "Winter", "Spring"	> 8	"Stew"	Less effort
6	"Summer"	-	"Light Salad ad nice Steak"	Hey, why not?
+	-	-	-	-

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions
xmlns="http://www.omg.org/spec/DMN/20151101/dmn11.xsd" id="definitions" name="definitions"
namespace="http://camunda.org/schema/1.0/dmn">
  <decision id="decision" name="Dish">
    <decisionTable id="decisionTable">
      <!-- ... -->
      <rule id="rule2-950612891-2">
        <inputEntry id="inputEntry21">
          <text>"Winter"</text>
        </inputEntry>
        <inputEntry id="inputEntry22">
          <text><![CDATA[<= 8]]></text>
        </inputEntry>
        <outputEntry id="outputEntry2">
          <text>"Roastbeef"</text>
        </outputEntry>
      </rule>
      <!-- ... -->
    </decisionTable>
  </decision>
</definitions>
```

Modeling DMN in Camunda: Decision Table Hit policies

- Unique Hit Policy
- Any Hit Policy
- First Hit Policy
- Rule Order Hit Policy
- Collect Hit Policy
    - Aggregators for Collect Hit Policy
        - SUM aggregator
        - MIN aggregator
        - MAX aggregator
        - COUNT aggregator



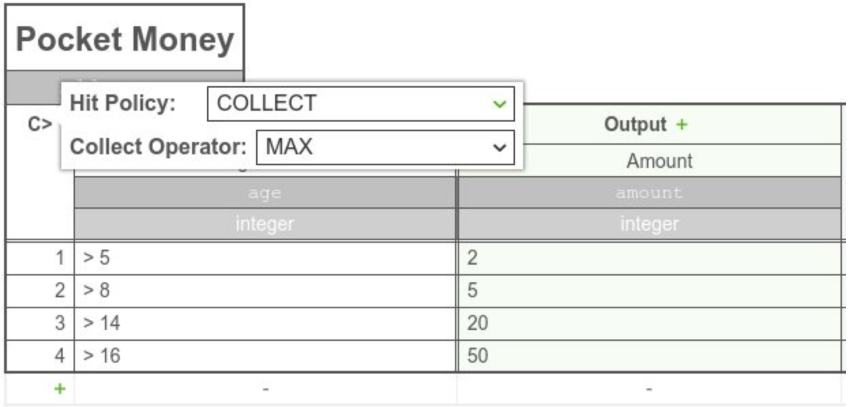
Salary

deci

Hit Policy: COLLECT

Collect Operator: SUM

C+		Output +
		Bonus
	year	bonus
	string	integer
1	> 1	100
2	> 2	200
3	> 3	300
4	> 5	500
+	-	-



Pocket Money

Hit Policy: COLLECT

Collect Operator: MAX

C>		Output +
		Amount
	age	amount
	integer	integer
1	> 5	2
2	> 8	5
3	> 14	20
4	> 16	50
+	-	-

Friendly Enough Expression Language (FEEL)

Language Elements

- Comparison
- Negation
- Date Functions
- Range

Start	End	Example
include	include	[1..10]
exclude	include	]1..10] or (1..10]
include	exclude	[1..10[or [1..10)
exclude	exclude	]1..10[or (1..10)

- Qualified Names
  - >= x: Test if the input is greater than or equal the variable x
  - < customer.age: Test if the input is less then the age property of the variable customer
- Disjunction
  - "Spareribs","Steak","Stew": Test if the input is any of the strings
  - <2,>10: Test if the input is either less than 2 or greater than 10
  - 10,[20..30]: Test if the input is either 10 or between 20 and 30
  - Disjunction of ranges
  - >customer.age,>21: Test if the input is greater than the age property of the variable customer or 21

Summary

- ✓ Business Rules Management term, motivation and relation to other BPM areas explained
- ✓ Key tasks in Business Rules Management elaborated
- ✓ Business rules evaluation and execution using Camunda outlined

Questions? Questions!

THANK YOU VERY MUCH FOR YOUR ATTENTION!