bbc 2017 BUILDING BUSINESS CAPABILITY

Orlando, Nov 6-10, 2017

bbc BUILDING BUSINESS CAPABILITY

Technology Theater

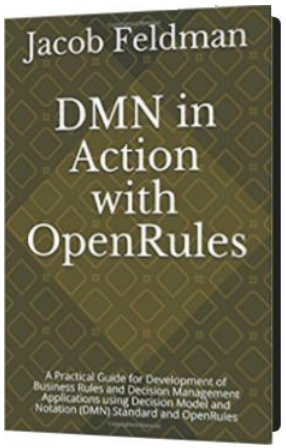
Decision Modeling in Action with DMN and OpenRules

Presenter: Dr. Jacob Feldman

OpenRules Inc., CTO

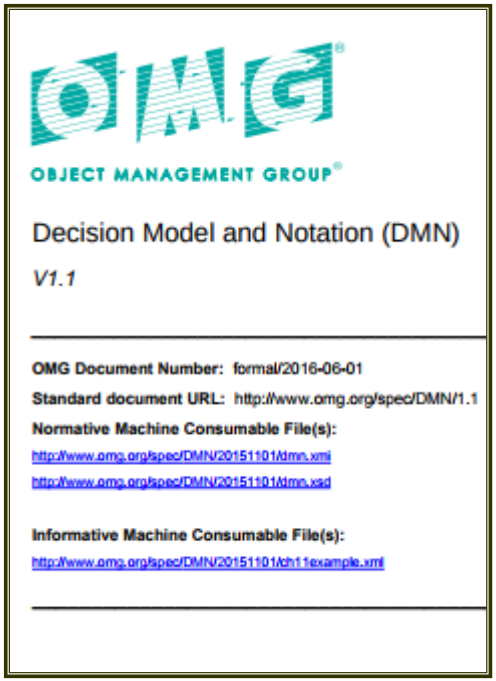
jacobfeldman@openrules.com

www.OpenRules.com



DMN - Decision Model and Notation

- OMG standard since 2014
- Specifies key concepts and constructs for Operational Business Decision Modeling
- Current release 1.1 supports DMN XML interchange format
- Next Release 1.2 – Q1 2018
- 17 vendors announced DMN support


OBJECT MANAGEMENT GROUP®

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

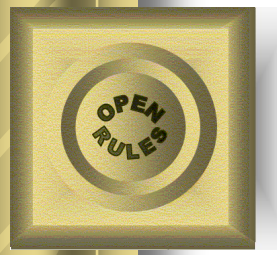
Normative Machine Consumable File(s):

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

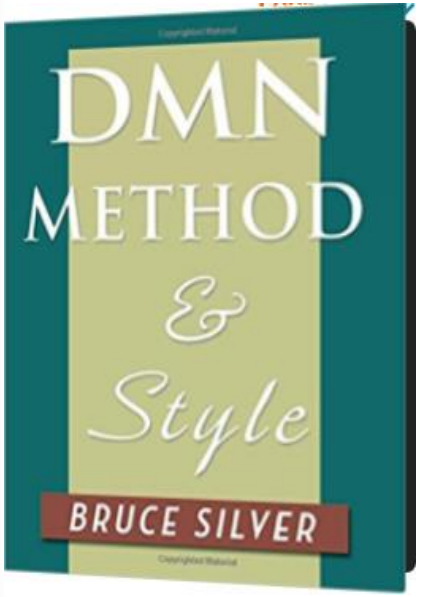


Decision Model and Notation (DMN) Supporting Tools

#	Product	Select
1	AlfrescoActiviti	☐
2	Avola	☐
3	BiZZDesign	☐
4	Blueriq	☐
5	Camunda	☐
6	DecisionsFirstModeler	☐
7	Drools	☐
8	FICO	☐
9	FlexRule	☐
10	IDIOM	☐
11	OneDecision	☐
12	OpenRules	☐
13	RapidGen	☐
14	Sapiens	☐
15	Signavio	☐
16	Sparkling Logic	☐
17	Trisotech	☐

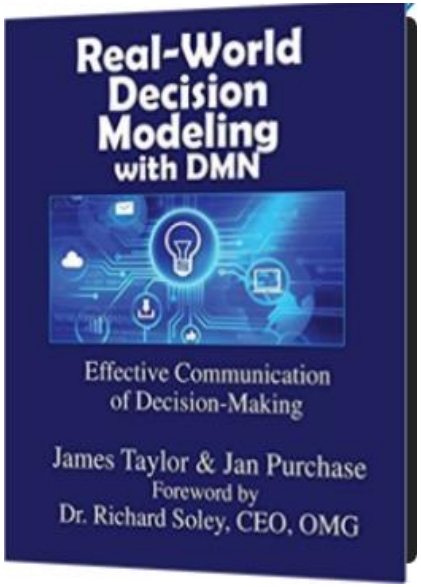
Recent DMN Books

Bruce Silver



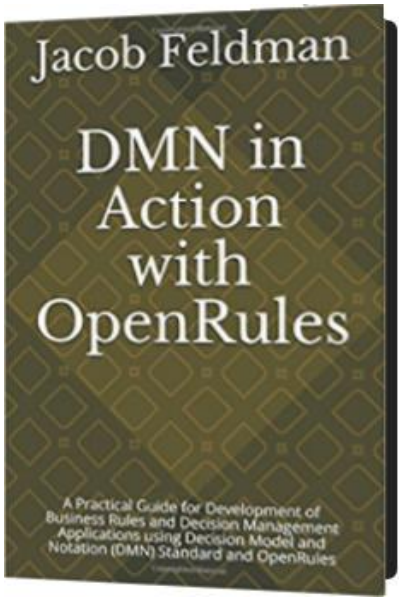
2016

James Taylor
Jan Purchase



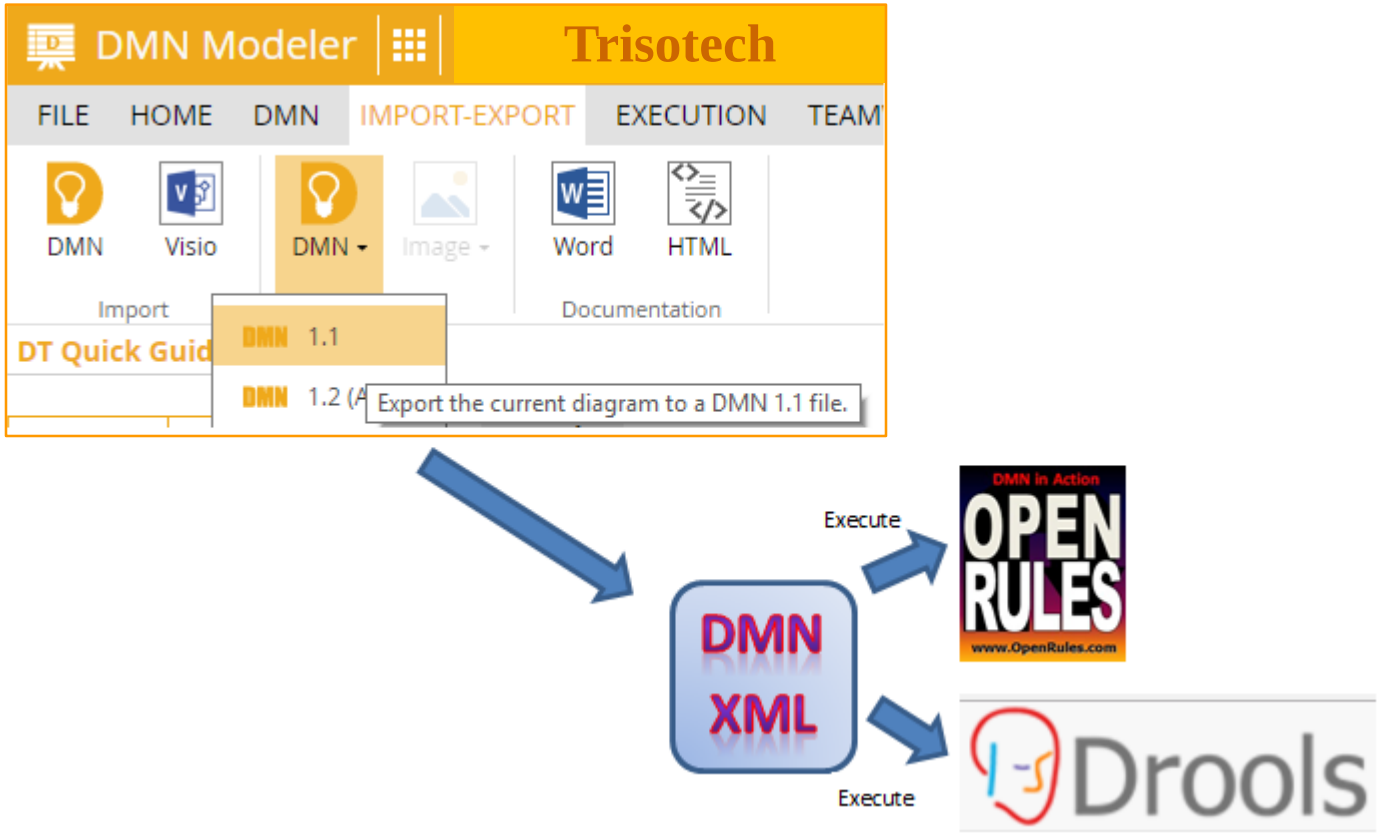
2016

Jacob Feldman



2017

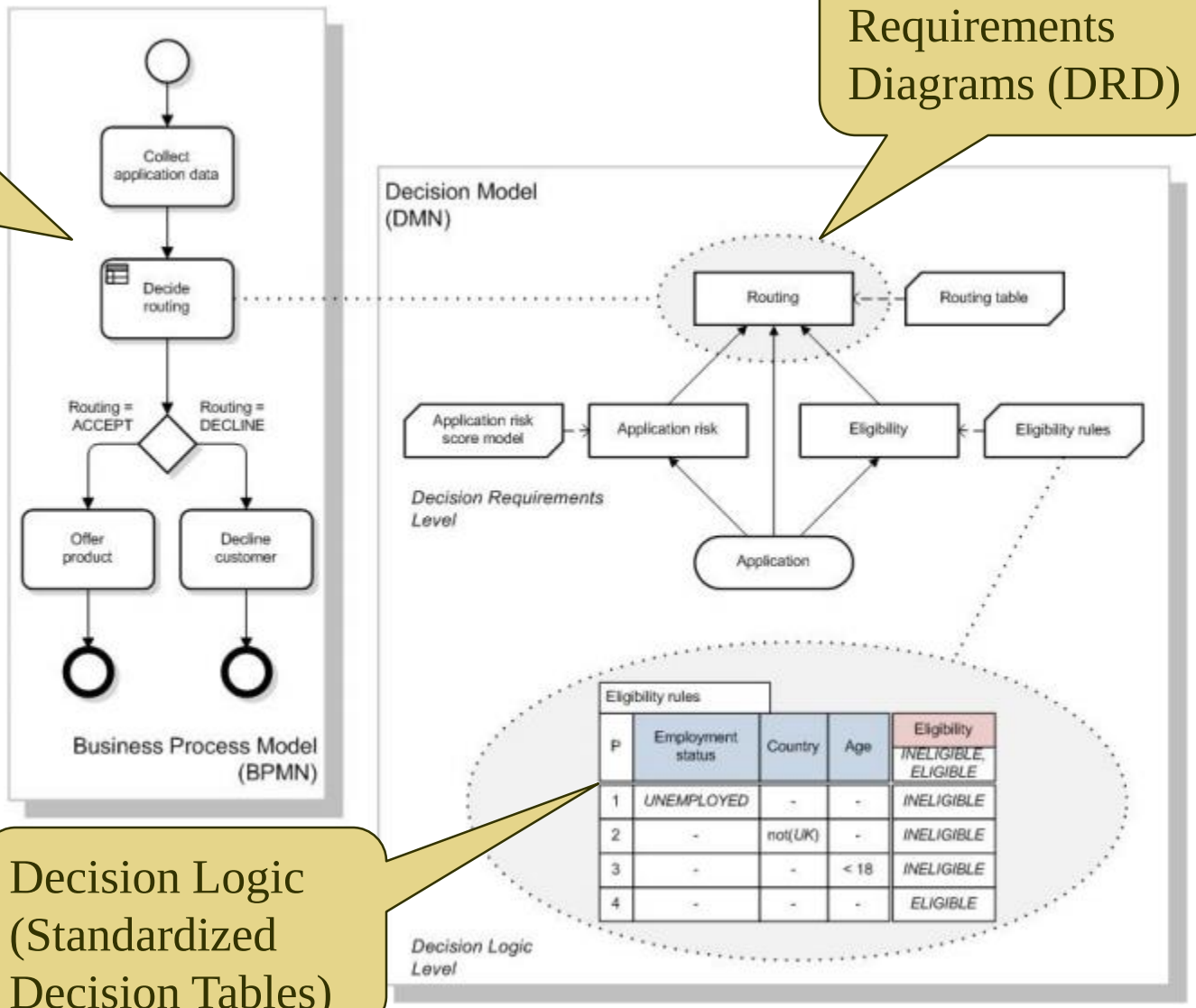
DMN Interchange



DMN Modeling Concepts

Integration with Business Processes (BPMN)

Decision Requirements Diagrams (DRD)



Decision Logic (Standardized Decision Tables)

Decision Modeling Constructs

- Core Constructs
  - Diagrams with Logical Connections (information requirements)
  - Decision Tables
  - Basic Expression Language (S-FEEL)
  - Conformance Level 2
- Advanced Constructs
  - Boxed Expressions (FEEL functions with parameters, contexts, if-then-else, for..return loops, filters, sorting, recursion, …)
  - Conformance Level 3

Hands-On Decision Modeling

- The best way to understand DMN is to build and test real Decision Models
- Let's build and execute for a quite popular business problems known as "**Vacation Days**"
- See 20 different solutions at www.DMCommunity.org

Decision Model "Vacation Days"

The number of vacation days depends on age and years of service.

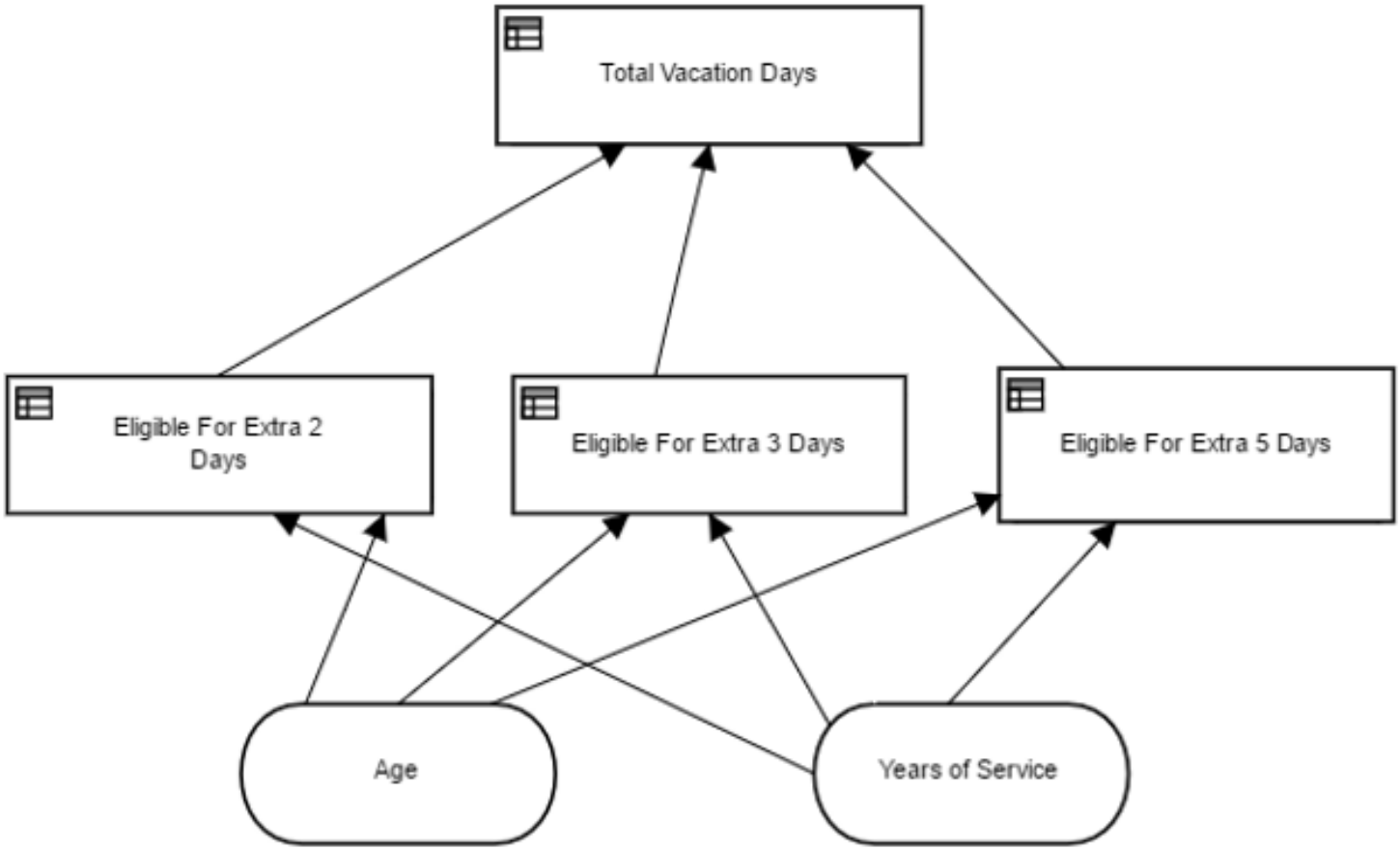
Every employee receives at least 22 days.
Additional days are provided according to the following criteria:

1) Only employees younger than 18 or at least 60 years, or employees with at least 30 years of service will receive 5 extra days.

2) Employees with at least 30 years of service and also employees of age 60 or more, receive 3 extra days, on top of possible additional days already given.

3) If an employee has at least 15 but less than 30 years of service, 2 extra days are given. These 2 days are also provided for employees of age 45 or more. These 2 extra days can not be combined with the 5 extra days.

DRD- Decision Requirement Diagram



Total Vacation Days

DecisionTableMultiHit CalculateVacationDays				
If	If	If	Conclusion	
Eligible for Extra 5 Days	**Eligible for Extra 3 Days**	**Eligible for Extra 2 Days**	**Vacation Days**	
			=	22
TRUE			+=	5
	TRUE		+=	3
FALSE		TRUE	+=	2

Eligible For 2 Extra Days

DecisionTable SetEligibleForExtra2Days		
If	If	Then
Age in Years	**Years of Service**	**Eligible for Extra 2 Days**
	[15..30)	TRUE
>= 45		TRUE
		FALSE

Eligible For 3 Extra Days

DecisionTable SetEligibleForExtra3Days		
If	If	Then
Age in Years	**Years of Service**	**Eligible for Extra 3 Days**
	>= 30	TRUE
>= 60		TRUE
		FALSE

Eligible For 5 Extra Days

DecisionTable SetEligibleForExtra5Days		
If	If	Then
Age in Years	**Years of Service**	**Eligible for Extra 5 Days**
< 18		TRUE
>= 60		TRUE
	>= 30	TRUE
		FALSE

Glossary

Glossary glossary		
Variable Name	**Object**	**Attribute**
Age in Years	Employee	age
Years of Service		service
Eligible for Extra 5 Days		eligibleForExtra5Days
Eligible for Extra 3 Days		eligibleForExtra3Days
Eligible for Extra 2 Days		eligibleForExtra2Days
Vacation Days		vacationDays

Test Data

Datatype Employee	
String	id
int	age
int	service
boolean	eligibleForExtra5Days
boolean	eligibleForExtra3Days
boolean	eligibleForExtra2Days
int	vacationDays

Data Employee employees

ID	Age in Years	Years of Service	Eligible for Extra 5 Days	Eligible for Extra 3 Days	Eligible for Extra 2 Days	Vacation Days
A	17	1	FALSE	FALSE	FALSE	0
B	25	5	FALSE	FALSE	FALSE	0
C	49	30	FALSE	FALSE	FALSE	0
D	49	29	FALSE	FALSE	FALSE	0
E	57	32	FALSE	FALSE	FALSE	0
F	64	42	FALSE	FALSE	FALSE	0

Test Cases with Expected Results

DecisionTableTest testCases		
#	ActionUseObject	ActionExpect
Test ID	**Employee**	**Vacation Days**
Test A	:= employees[0]	27
Test B	:= employees[1]	22
Test C	:= employees[2]	30
Test D	:= employees[3]	24
Test E	:= employees[4]	30
Test F	:= employees[5]	30

Executing Decision Model against Test Case A

```
RUN TEST: Test A 2017-11-06 14:30:53.758
Create report directory: report
Decision DetermineVacationDays: Define Vacation Days
Decision DefineVacationDays: SetEligibleForExtra 5 Days
  Assign: Eligible for Extra 5 Days  = true [true]
Decision DefineVacationDays: SetEligibleForExtra 3 Days
  Assign: Eligible for Extra 3 Days  = false [false]
Decision DefineVacationDays: SetEligibleForExtra 2 Days
  Assign: Eligible for Extra 2 Days  = false [false]
Decision DefineVacationDays: Define Vacation Days
  Conclusion: Vacation Days  = 22 [22]
  Conclusion: Vacation Days  += 5 [27]
Decision DetermineVacationDays: Show Results
Employee(id=0) {
    id=A
    age=17
    eligibleForExtra2Days=false
    eligibleForExtra3Days=false
    eligibleForExtra5Days=true
    service=1
    vacationDays=27
    }
Validating results for the test <Test A>
Test A was successful
Executed test Test A in 27 ms
```

Execution Report with Explanations (Test A)

Decision "DetermineVacationDays" (report/Test A.html)

Executed Decision Tables and Rules (Mon Nov 06 14:30:53 EST 2017)

Decision Table : Rule#	Executed Rule	Variables and Values
SetEligibleForExtra5Days:1	**IF** Age in Years < 18 **THEN** Eligible for Extra 5 Days = true	Age in Years=17 Eligible for Extra 5 Days=true
SetEligibleForExtra3Days:3	Eligible for Extra 3 Days = false	Eligible for Extra 3 Days=false
SetEligibleForExtra2Days:3	Eligible for Extra 2 Days = false	Eligible for Extra 2 Days=false
CalculateVacationDays:1	Vacation Days = 22	Vacation Days=22
CalculateVacationDays:2	**IF** Eligible for Extra 5 Days = true **THEN** Vacation Days += 5	Eligible for Extra 5 Days=true Vacation Days=27

Execution Report with Explanations (Test C)

Decision "DetermineVacationDays" (report/Test C.html)

Executed Decision Tables and Rules (Mon Nov 06 14:30:53 EST 2017)

Decision Table : Rule#	Executed Rule	Variables and Values
SetEligibleForExtra5Days:3	**IF** Years of Service >= 30 **THEN** Eligible for Extra 5 Days = true	Years of Service=30 Eligible for Extra 5 Days=true
SetEligibleForExtra3Days:1	**IF** Years of Service >= 30 **THEN** Eligible for Extra 3 Days = true	Years of Service=30 Eligible for Extra 3 Days=true
SetEligibleForExtra2Days:2	**IF** Age in Years >= 45 **THEN** Eligible for Extra 2 Days = true	Age in Years=49 Eligible for Extra 2 Days=true
CalculateVacationDays:1	Vacation Days = 22	Vacation Days=22
CalculateVacationDays:2	**IF** Eligible for Extra 5 Days = true **THEN** Vacation Days += 5	Eligible for Extra 5 Days=true Vacation Days=27
CalculateVacationDays:3	**IF** Eligible for Extra 3 Days = true **THEN** Vacation Days += 3	Eligible for Extra 3 Days=true Vacation Days=30

Alternative DMN DecisionTable

DecisionTable DefineVacationDays		
If	If	Then
Age in Years	**Years of Service**	**Vacation Days**
<18		22 + 5
[18..45)	<15	22
[18..45)	[15..30)	22 + 2
[18..45)	>=30	22 + 5 + 3
[45..60)	<15	22 + 2
[45..60)	[15..30)	22 + 2
[45..60)	>=30	22 + 5 + 3
60+		22 + 5 +3

It may look more compact but:

- It's hard to recognize the plain English logic
- Difficult to change or add more rules

In Decision Modeling Compactness is not always your friend!

My Presentation "DMN without Programming"

- Title
  - How Business Analysts Build Executable Decision Models with DMN but without Programming
- Main idea
  - Keep DMN Simple, Oriented to Business People
- When:
  - Fri Nov 10: 9:00 – 10:00 am
- Where:
  - Room: Timor 1 & 2