DMN Community Challenge January 2016

Decision Table for Vacation Days Calculation

By:

Mélanie Gauthier

Trisotech

Email: mgauthier@Trisotech.com

Website: www.Trisotech.com

January 18, 2016

Table of Contents

Introduction

The Decision Management Community [1] provides resources for DM practitioners, listing tools, news, providing examples and also giving a monthly challenge.

As a vendor that always seek improvement, we took the January challenge using Trisotech DMN Modeler [2].

The Challenge

The January challenge is to create the best possible decision tables to represent the following statements, provided by Prof. Jan Vanthienen who is one of the best known experts in decision tables and a member of the DMN Task Force.

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.

Problem Analysis

The provided statements can be a little confusing: if an employee is 60 years old and has 30 years of experience, can he get the 3 extra days twice? We took the assumption that each extra days can only be awarded once.

At first, we started drawing three decision tables to represent the details of the three rules and then combined the result of those in another decision table that would decide if rule #3 can be applied or not (depending on rule #1). That table would calculate the vacation days based on all the data. This approach provided the appropriate results, but it was really complex and you had to go through four decision tables to actually view the rules.

We then took a step back and analysed more closely the statements to determine properly what affects the number of vacation days.

This allows us to split the two inputs (age and years of service) into the range that matter.

Four age ranges are important:

- Younger 18
- 18 to younger than 45
- 45 to younger than 60
- 60 or older

Three years of service ranges are important:

- Less than 15 years
- 15 years to less than 30 years
- 30 years or more

Solution 1: Unique Hit Policy Decision Table

To make the decision table easy to read, we decided the first column would represent the various age possibilities, in increasing order. The second input will be the years of service that affect employees in that particular age range, in increasing order too. We left the calculation mainly to make it more obvious from where the number comes.

Decision Table

	inputs		outputs	
U	Age	Years of Service	Vacation Days	Description
	Number [14,)	*Number [0,)*	*Number [22,)*	
1	<18	-	22+5	Basic days + rule #1 People younger than 18 cannot have 30 years of experience and this is the only case that would have changed their holidays count so this information is irrelevant
2	[18..45)	<15	22	Basic days
3		[15..30)	22+2	Basic days + rule #3
4		>=30	22+5+3	Basic days + rule #1 + rule #2
5	[45..60)	<30	22+2	Basic days + rule #3
6		>=30	22+5+3	Basic days + rule #1 + rule #2
7	>=60	-	22+5+3	Basic days + rule #1 + rule #2

Advantages

- Unique hit policy decision tables are easier to validate for completeness (no rule overlapped)
- This table is easy to read since values are ordered
- Every case is depicted

Disadvantage

- Complex to edit if the rules change

A spin on the statement

Doing that table helps to view the requirement more clearly. They can be rewritten as follow:

> Every employee gets 22 vacation days.
>
> Depending on age and years of service, the employee gets the following extra days (use only the first rule that applies; extra days can only be awarded once)
>
> - Under 18, employee gets 5 extra vacation days
> - 60 or over or having at least 30 years of service, employee gets 8 extra vacations days
> - 45 or over or having at least 15 years of service, employee gets 2 extra vacation days
> - Otherwise employee gets no extra days

Solution 2: First Hit Policy Decision Table

With the above statements, it is easy to draw a First Hit Policy decision table.

Decision Table

	inputs		outputs	
F	Age	Years of Service	Vacation Days	Description
	Number [14,)	*Number [0,)*	*Number [22,)*	
1	<18	-	22+5	Basic days + rule #1 People younger than 18 cannot have 30 years of experience and this is the only case that would have changed their holidays count so this information is irrelevant
2	>=60	-	22+8	Basic days + rule #1 + rule #2
3	-	>=30		
4	>=45	-	22+2	Basic days + rule #3
5	-	>=15		
6	-	-	22	Basic days only

Advantage:

- Easier to change parameter: if you want to change 45 to 40, you just need to change rule #4; in the previous table, many rows are affected
- Easier to see which conditions gives you 27/30/24/22 vacations days since they are grouped together

Disadvantage

- First hit policy may be more confusing to user
- Harder to validate manually, although in this particular cases, the rules are simple enough to be visualized.

Additional details

In both tables, we included some constraints for the inputs and the output.

Age: 14 or older (this is actually depending on your country/stage, this is the legal age to work in Quebec, Canada)

Years of experience: 0 or greater

Vacation days: 22 or more, that way if a new rule is introduced that gives less than 22 days, a rule execution would trigger an error.

Which solution is the best?

The best solution really depends on the modeler preference and what needs to be emphased.

Respected Prof. Jan Vanthienen prefers to avoid first hit policy for the reason mentioned previously. He favors complete unique decision table that gives a more complete picture of the problem.

About Trisotech

Trisotech is a global leader in digital enterprise solutions, offering innovative and easy-to-use software tools that allow customers to discover, model, analyze and find insights into their digital enterprise. One of the tool provided is the DMN Modeler [2] which allows to easily create decision model and decision tables.

References

[1] DMN Community: https://dmcommunity.wordpress.com

[2] Trisotech DMN Modeler: http://www.trisotech.com/dmn-modeler