

# DM Community Challenge Feb 2018

---

*Submitted by Jan Purchase, Decision Architect, [Lux Magi Ltd](#).*

## Introduction

The challenge was to represent the logic, provided by a tax subject matter expert, which determines the next tax filing action for a taxpayer in the most effective format. A rules-as-columns decision table representing the logic was provided as a specification. The [original challenge can be found here](#).

I interpreted 'the most effective format' to mean:

- One that represents the requirement transparently to other stakeholders
- One that is concise and accurate
- One that is easy to change
- One that reproduces the behaviour of the original decision as far as possible

I selected the decision model and notation standard (DMN) as a vehicle to represent my model. The use of a standard makes my model available to the widest possible audience and avoids vendor specific features. I chose to capture the model using [Trisotech](#) DMN Modeler (part of the Trisotech Digital Enterprise Suite) because of its ease of use, power and its complete support for DMN 1.1 compliance level 3. Because I am using a standard means that there is nothing specific to Trisotech's product in my solution.

The example is a simple decision, consisting only of a single decision table. Although this isn't very representative of the scale of real decision models and cannot be used to show how well DMN handles complexity at scale, or [its other benefits](#), it does show the advantages of the DMN representation of logic.

I implemented and tested the decision model using [RapidGen](#), a high-performance DMN execution engine. I was interested to see the level of performance I could achieve with a simple decision model.

## Preparation of the Model

One of the big advantages of decision modeling is that the rigour of the approach helps one to detect and remove inconsistencies in the logic very early. It also gives us the rigour to simplify the representation of the logic without changing it. Whilst translating the logic provided in the decision table specification into DMN I spotted some mistakes in its content and some opportunities for simplification. The flaws in the original specification are covered later.

The logic specified by the subject matter expert is shown in Figure 1. Note the original rule numbers at the top; these will be pivotal in discussing the logic of the specification.

This Year's Filing	1	2	3	4	5	6	7	8	9	10	11	12
Doc Code in 7, 8, 28	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Taxable Income < 50000	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Interest Income < 400	-	N	N	N	N	-	-	-	-	Y	Y	Y
Taxable Pension = 0	-	-	-	-	-	N	N	Y	Y	Y	Y	Y
Age <55	-	Y	N	N	Y	N	N	N	N	Y	Y	Y
AGI < 25000	-	-	N	Y	-	N	Y	N	Y	-	-	-
Doc Code = 28	-	-	-	-	-	-	-	-	-	Y	N	N
Zip Code not in SC, APO, LL	-	-	-	-	-	-	-	-	-	-	Y	-
<b>THEN</b>												
1040EZ Telefile (MFR15)										Y	Y	
1040EZ (MFR 13)												Y
1040A plus Sch 1,2 (MFR 02)	Y	Y	Y		Y	Y		Y				
1040A with Sch R (MFR 14)				Y			Y		Y			

Figure 1 Specified Tax Logic

### Opportunities for Simplification

This logic has a number of opportunities for simplification. Specifically:

1. Rule 5 is a duplicate of 2 (so I removed 5)
2. Rules 6 and 8 can be combined into one rule independent of taxable pension amount (so I simplified 6 and removed 8)
3. Rules 7 and 9 can be combined into one rule independent of taxable pension amount (so I simplified 7 and removed 9)
4. Rules {6,7} now eclipse (subsume) {3,4} (so I removed 3 and 4)

Therefore this logic can be simplified to that of Figure 2.

This Year's Filing	1	2	6	7	10	11	12
Doc Code in 7, 8, 28	Y	Y	Y	Y	Y	Y	Y
Taxable Income < 50000	N	Y	Y	Y	Y	Y	Y
Interest Income < 400	-	N	-	-	Y	Y	Y
Taxable Pension = 0	-	-	-	-	Y	Y	Y
Age <55	-	Y	N	N	Y	Y	Y
AGI < 25000	-	-	N	Y	-	-	-
Doc Code = 28	-	-	-	-	Y	N	N
Zip Code not in SC, APO, LL	-	-	-	-	-	Y	-
<b>THEN</b>							
1040EZ Telefile (MFR15)					Y	Y	
1040EZ (MFR 13)							Y
1040A plus Sch 1,2 (MFR 02)	Y	Y	Y				
1040A with Sch R (MFR 14)				Y			

Figure 2 Simplified Tax Logic

### Resulting Model in DMN

This can be reformatted into the DMN decision model DRD shown in Figure 3 and the decision table in Figure 4.

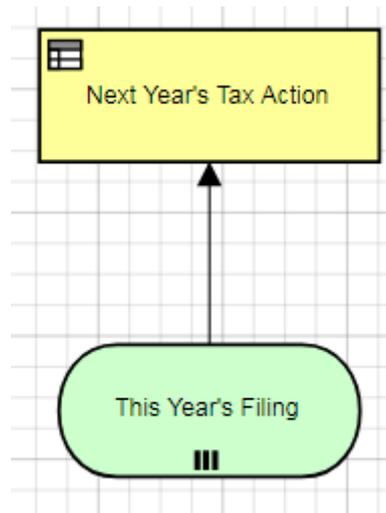


Figure 3 DMN Decision Requirement Diagram

**Next Year's Tax Action**  
*tNextYearAction*  
 "1040EZ (MFR 15)", "1040EZ (MFR 13)", "1040A (MFR 02)", "1040A (MFR14)"

inputs								outputs	
U	This Year's Filing Document Code	This Year's Filing Taxable Income	This Year's Filing Interest Income	This Year's Filing Age	This Year's Filing AGI	This Year's Filing Taxable Pension	This Year's Filing Zip Code Status	Next Year's Tax Action	Description
	<i>tDocCode</i> [1..30]	<i>tAmount</i> ≥0	<i>tAmount</i> ≥0	<i>tAge</i> [16..120]	<i>tAmount</i> ≥0	<i>tAmount</i> ≥0	<i>tZipCodeStatus</i> "OK", "SC", "APO", "LL"	<i>tNextYearAction</i> "1040EZ (MFR 15)", "1040EZ (MFR 13)", "1040A (MFR 02)", "1040A (MFR14)"	
1	7, 8, 28	≥50000	-	-	-	-	-	"1040A (MFR 02)"	line 1
2		<50000	≥400	<55	-	-	-	"1040A (MFR 02)"	line 2, 5
3			-	≥25000	-	-	-	"1040A (MFR 02)"	line 3, 6, 8
4		-	<25000	-	-	-	-	"1040A (MFR14)"	line 4, 7, 9
5	28	<50000	<400	<55	-	0	-	"1040EZ (MFR 15)"	line 10
6	7, 8	<50000	<400	<55	-	0	not("SC", "APO", "LL")	"1040EZ (MFR 15)"	line 11
7							"SC", "APO", "LL"	"1040EZ (MFR 13)"	line 12

Figure 4 DMN Decision Table

This representation of decision logic – ‘rules as rows’ is, in my experience, more accessible by business subject matter experts than the original format and more scalable. Specifically it copes with cases where the number of rules exceeds the number of conditions which is by far the most common case in practice.

I’ve tested the resulting table, using Trisotech’s excellent test case facility, and it behaves the same way as the original specification for all the boundary test cases.

## Choice of Hit Policy

I was asked by some of my colleagues about why I chose the Unique hit policy as opposed to First, which considers the candidacy of each rule in order and hits the first one that is satisfied, yielding its outcome. The First hit policy is an intuitive format that is favoured by some SMEs and can yield more concise decision tables for simple, exception based logic.

Our DMN training course addresses this point in more detail, but my caution with the use of the First hit policy arises from my years of experience with it in the context of large decision models. Specifically, because the behaviour of rules in a First hit policy table depends on all the rules above them:

- The tables can be hard to understand and verify when they become large.
- It can be difficult to understand the behaviour of rules that occur late in such large tables
- It can be difficult to work out how to edit such tables to achieve required changes in the logic – there are more dependencies to consider (notably a dependency between each rule and the rules that appear above it in the table) and more opportunities for ‘collateral damage’ as a result
- One cannot use order to sort rules to make them more legible by bringing together related rules (see section 15.3 of our book [‘Real World Decision Modeling with DMN’](#)) because rule order determines semantics.

In short, use of the First hit policy can yield concise intuitive logic, ideal for simple business logic that will not change. But it can be a liability for larger tables or those destined to change frequently. As a result, I’d be cautious about using it as a default option in this case.

## Flaws with the Specification

Further analysis detected three logical gaps in the original specification and one overlap.

### Logical Gaps

The specification does not cover specific combinations of circumstances. Specifically these are cases that satisfy rules 10-12 for all circumstances except that the taxable pension is greater than zero. The original specification does not provide rules that cover these scenarios. Specifically no outcome is defined for the table in this combined case:

- The *Document Code* is one of 7, 8 or 28
- The *Taxable Income* is less than 50000
- The *Interest Income* is less than 400
- The *Age* is less than 55
- The *Taxable Pension* is greater than 0

In addition, the specification does not define what happens if the Document Code has any other value than 7, 8 or 28.

### Overlap

In the original specification rules 11 and 12 overlap, that is, their conditions can both be satisfied at the same time, specifically by the case:

- The *Document Code* is 7 or 8
- The *Taxable Income* is less than 50000
- The *Interest Income* is less than 400
- The *Age* is less than 55
- The *Taxable Pension* is 0
- The *Zip Code* is not one of APO, LL, SC

This is made more onerous by the fact that these two rules have different outcomes. The intended meaning of these rules is unclear because the specification does not indicate what should happen when multiple rules are satisfied. Should the first rule to be satisfied be hit (First semantics) or should every rule to be satisfied be hit (Collect semantics)? I have assumed the former in my solution as it makes more sense.

### **Resolution**

At this juncture I would ask the subject matter expert what should happen in these gap and overlap circumstances and whether they constitute realistic business cases. I am not a tax expert and therefore would not attempt to resolve the logical gaps myself. I have resolved the overlap in my solution (in order to satisfy the Unique constraint), but again I would seek to confirm my interpretation with a subject matter expert at the earliest opportunity.

### **Execution**

I deployed the decision model into a RapidGen Genius DMN execution engine running on a HP Omen 1500 laptop, with mobile Core i7 3.2GHz, 16Gb and SSD, to check the veracity of the rules and test performance. Against bulk generated data, and with transaction logging disabled, I measured a throughput of 500k decisions per second – very impressive for such modest hardware! This certainly outperforms many of the Business Rules Management Systems I have used over the years. The rules executed as expected given the caveats noted above.

### **Included with the Submission**

This submission includes a working decision model with 15 test cases (12 corresponding to the original rules and 3 corresponding to the missing case).

Any feedback, thoughts, questions? [jap@luxmagi.com](mailto:jap@luxmagi.com)

---

Jan Purchase is chief decision architect at Lux Magi Ltd. He runs a decision management practice and both consults and trains in decision management and DMN. He also blogs on the subject at [dmblog.luxmagi.com](http://dmblog.luxmagi.com).