Close is Not Close Enough

DMN Technology Compatibility Kit (TCK)
Year Three (and a half)

Keith D Swenson
Fujitsu America
Sept 19, 2019

Decision Camp 2019
Bolzano, Italy
Decision Model and Notation

Success: $\text{public perception} = \text{delivered capability}$
### Experience of Employee

<table>
<thead>
<tr>
<th>UC</th>
<th>Approval Authority</th>
<th>Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>&lt; 1000</td>
<td>low</td>
</tr>
<tr>
<td>2</td>
<td>[1000..10000]</td>
<td>medium</td>
</tr>
<tr>
<td>3</td>
<td>&gt; 10000</td>
<td>high</td>
</tr>
</tbody>
</table>

- **Completeness Indicator** (here Complete)
- **Hit Policy** (here Unique)
- **Name of Decision Table**
- **Input**
- **Output**

*One Row = One Rule*
### Experience of Employee

<table>
<thead>
<tr>
<th>UC</th>
<th>Approval Authority</th>
<th>Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X**2 &amp; %34</td>
<td>²m</td>
</tr>
<tr>
<td>2</td>
<td>55 ∈ az △ 1</td>
<td>²n(*2)</td>
</tr>
<tr>
<td>3</td>
<td>Δ</td>
<td>²m</td>
</tr>
</tbody>
</table>

**Completeness Indicator** (here Complete)

**Hit Policy** (here Unique)

**Rule Number**

**Name of Decision Table**

**Input**

**Output**

**One Row = One Rule**
What BPM Guide has to say…

Why should I care about DMN?

- **Standard**: DMN is not owned by a certain enterprise but by an institution (OMG), which is already established through other world-wide standards, e.g., BPMN or UML. The standard is supported by many software products; you are less dependent on any particular vendor’s products.

- **Business-IT-Alignment**: Decisions can be modeled and executed using the same notation. This allows to use business analysis results as “code” making changes to the rules behind the decision really easy. Given appropriate tooling it might even be realistic that business people can directly edit concrete rules within a given structure (more on this later). As rules (leading to decisions) change much more often than processes or entities, being agile in changing rules is absolutely necessary for being future ready!

- **Experience**: DMN was developed by people having decades of experience with Decision Management. Even though, the standard does not force any special implementation choices for rule engines, allowing for modern and lightweight implementations (like in camunda BPM :-)).

1 Scope

The primary goal of DMN is to provide a common notation that is readily understandable by all business users, from the business analysts needing to create initial decision requirements and then more detailed decision models, to the technical developers responsible for automating the decisions in processes, and finally, to the business people who will manage and monitor those decisions. DMN creates a standardized bridge for the gap between the business decision design and decision implementation. DMN notation is designed to be usable alongside the standard BPMN business process notation.

Another goal is to ensure that decision models are interchangeable across organizations via an XML representation. The authors have brought forth expertise and experience from the existing decision modeling community and have sought to consolidate the common ideas from these divergent notations into a single standard notation.

WITH or WITHOUT fidelity?
DMN TCK is

A way for vendors to
➔ **demonstrate** their compliance to standard

Provide files to help vendors
➔ test for errors and become **compliant**

Customers to assess
➔ how **compliant** a vendor is.
The Goals of the TCK

- Define a set of Test Cases
- Carefully assure conformance to spec.
- Provide tools to run the tests
- Recognize the vendor success
DMN Implementation Engine being tested

Runner

DMN Model

Vendor’s domain

Examples available

What the TCK provides
1. Input data
2. Input data
3. Input data
<table>
<thead>
<tr>
<th>Company</th>
<th>Technology/Platform</th>
<th>Submitted</th>
<th>Last Submission</th>
</tr>
</thead>
<tbody>
<tr>
<td>Camunda</td>
<td>Camunda BPM 7.9.0</td>
<td>0/1598</td>
<td>2018-11-16</td>
</tr>
<tr>
<td>Fujitsu America</td>
<td>Digital Transformation Platform (DXP) 2.4</td>
<td>1597/1598</td>
<td>2019-07-22</td>
</tr>
<tr>
<td>Goldman Sachs, L.</td>
<td>jDMN 3.0.0</td>
<td>1442/1598</td>
<td>2019-05-23</td>
</tr>
<tr>
<td>OpenRules, Inc.</td>
<td>OpenRules 7.0.0</td>
<td>0/1598</td>
<td>2017-07-05</td>
</tr>
<tr>
<td>Oracle</td>
<td>Oracle Process Cloud 17.3.3</td>
<td>0/1598</td>
<td>2017-07-21</td>
</tr>
<tr>
<td>Red Hat</td>
<td>Drools 7.25.0.Final</td>
<td>1598/1598</td>
<td>2019-08-13</td>
</tr>
<tr>
<td>Trisotech</td>
<td>Trisotech DMN Modeler 6.2.7</td>
<td>1598/1598</td>
<td>2019-08-26</td>
</tr>
</tbody>
</table>

The DMN TCK is supported by:
Drill Down to find out which tests pass and fail
Down to the individual test level

<table>
<thead>
<tr>
<th>Compliance</th>
<th>Test Suite</th>
<th>Doc</th>
<th>Source</th>
<th>Test</th>
<th>Digital Transformation Platform (DXP) 2.4</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>compliance-level-2</td>
<td>00009:invocation arithmetic-test 01</td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
</tr>
<tr>
<td></td>
<td>0003:iteration-test-01</td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
</tr>
<tr>
<td></td>
<td>0004:landing-test-01</td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
</tr>
<tr>
<td></td>
<td>0005:literal-invocation-test-01</td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
</tr>
<tr>
<td></td>
<td>0014:loan-comparison-test-01</td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
</tr>
<tr>
<td></td>
<td>0016:sum-every-test-01</td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
</tr>
<tr>
<td></td>
<td>0022:for-loops-test-01</td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
<td><img src="https://example.com" alt="image" /></td>
</tr>
</tbody>
</table>

The DMN TCK is supported by:

- actico
- camunda
- Fujitsu
- Method & Style
- Open Rules
- Oracle
- Red Hat
- Trisotech
<table>
<thead>
<tr>
<th>Metric</th>
<th>2016</th>
<th>2017</th>
<th>2018</th>
<th>2019</th>
<th>Sept</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor Results</td>
<td>0</td>
<td>1</td>
<td>7</td>
<td>8</td>
<td>8*</td>
</tr>
<tr>
<td>Models</td>
<td>0</td>
<td>39</td>
<td>65</td>
<td>112</td>
<td>122</td>
</tr>
<tr>
<td>Decision Objects</td>
<td>0</td>
<td>183</td>
<td>677</td>
<td>1135</td>
<td>1797</td>
</tr>
<tr>
<td>Test Files</td>
<td>0</td>
<td>34</td>
<td>62</td>
<td>105</td>
<td>110</td>
</tr>
<tr>
<td>Test Cases</td>
<td>0</td>
<td>61</td>
<td>588</td>
<td>1098</td>
<td>1679</td>
</tr>
<tr>
<td>Run Enabled</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1598</td>
</tr>
</tbody>
</table>
Moved Fully to DMN 1.2

- Decided not to support both 1.1 and 1.2 at the same time
  - 1.2 included many bug fixes
  - No desire to duplicate all the tests

- Archived the 1.1 website
- Converted all the models to the new syntax
- Adopted new tests enabled by 1.2 spec
Many many new tests

- Thanks to: Doug McCreath – Australia

- Approximately 1000 new tests by examining the specs

- Ran into a dozen issues that needed to go to the RTF
  - Type conversions
  - Type support and null
  - Nested lists
Top Issues

- Most issues were “details”
  - A couple of examples in the document that did not match the rules
  - Lists of lists, mixed types, new “Any” type
- Referring to (loading) another model
  - TCK expects models to be found in the same folder
  - Value and parameter passing when model schemas conflict
- Unicode support in string literals:
  - \uXXXX defined, but no explanation as to what XXXX means
- Type coercions
- Error handling (see next page)
Error Handling

- FEEL takes the approach an error causes a null output

- Processing does not stop!
  - Contrary to the “fail fast” design goal

- Spec says that some sort of error message *might* be produced
  - But does not clearly define when they appear
  - There is no requirement that they ever appear

- TCK Introduces a way to indicate an error test, but we can’t use it

- While the spec is ambiguous, the TCK should take a bolder stance and clarify exactly when an error is required, and when it is not needed.
  - The RTF clearly won’t get around to doing this.
All the tests are freely available to anyone

Anyone can join and participate in the TCK

A significant benefit for those implementing DMN

- 120 test models
- 1600 test cases
## Vendors Claiming DMN Support

<table>
<thead>
<tr>
<th>#</th>
<th>Product</th>
<th>Select</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Actico</td>
<td>☐</td>
</tr>
<tr>
<td>2</td>
<td>AlfrescoActiviti</td>
<td>☐</td>
</tr>
<tr>
<td>3</td>
<td>Avola</td>
<td>☐</td>
</tr>
<tr>
<td>4</td>
<td>BIZZDesign</td>
<td>☐</td>
</tr>
<tr>
<td>5</td>
<td>Bluerg</td>
<td>☐</td>
</tr>
<tr>
<td>6</td>
<td>Camunda</td>
<td>☐</td>
</tr>
<tr>
<td>7</td>
<td>DecisionsFirstModeler</td>
<td>☐</td>
</tr>
<tr>
<td>8</td>
<td>Drools</td>
<td>☐</td>
</tr>
<tr>
<td>9</td>
<td>FICO</td>
<td>☐</td>
</tr>
<tr>
<td>10</td>
<td>FlexRule</td>
<td>☐</td>
</tr>
<tr>
<td>11</td>
<td>Fujitsu</td>
<td>☐</td>
</tr>
<tr>
<td>12</td>
<td>IDIOM</td>
<td>☐</td>
</tr>
<tr>
<td>13</td>
<td>OneDecision</td>
<td>☐</td>
</tr>
<tr>
<td>14</td>
<td>OpenRules</td>
<td>☐</td>
</tr>
<tr>
<td>15</td>
<td>RapidGen</td>
<td>☐</td>
</tr>
<tr>
<td>16</td>
<td>Sapens</td>
<td>☐</td>
</tr>
<tr>
<td>17</td>
<td>Signavio</td>
<td>☐</td>
</tr>
<tr>
<td>18</td>
<td>Sparkling Logic</td>
<td>☐</td>
</tr>
<tr>
<td>19</td>
<td>Trisotech</td>
<td>☐</td>
</tr>
</tbody>
</table>

*Has Results*
Its really simple…

If they don’t have results on the DMN TCK Site,
They don’t have DMN.
Participants
If you are creating DMN models
• which are valid and well formed, and
  • you have defined test cases, and
    • you would like them to execute on
      • all vendors DMN engines
        • including all future versions…

Then… please submit them to DMN TCK!

It is a kind of guarantee for future compatibility
Future Directions

- Address 1.3 Changes
- Address the things that are too detailed for the RTF to handle
- Consider tests for modeling tools

(?️) Put forward a standard API for Decision as a Service

- DMN models already define exact inputs needed for a given decision output.
- Why not take one more small step to define how to submit these inputs in JSON and get JSON results back?
- Then you can code a call from browser UI or from BPM model to the decision, without having to change it for every DMN vendor
### Experience of Employee

<table>
<thead>
<tr>
<th>Rule Number</th>
<th>Approval Authority</th>
<th>Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>&lt; 1000</td>
<td>low</td>
</tr>
<tr>
<td>2</td>
<td>[1000..10000]</td>
<td>medium</td>
</tr>
<tr>
<td>3</td>
<td>&gt; 10000</td>
<td>high</td>
</tr>
</tbody>
</table>

#### Address

http://myservice/decisions/.../ExperienceOfEmployee

#### Input

```json
{
   "Approval Authority": 987
}
```

#### Output

```json
{
   "Experience": "low"
}
```
An API Enables Integration

- **Browser**
  - Web Page
  - HTML & JS
  - Code To call Rule

- **BPM Engine or Application**
  - Business Process or Application Code
  - Code To call Rule

- **Engine**
  - **DMN Model**
  - API
An API Enables Integration

Browser

Web Page
HTML & JS

Code To call Rule(s)

DMN Model
Engine
API

DMN Model
Engine
API

DMN Model
Engine
API

DMN Model
Engine
API

DMN Model
Engine
API
DMN TCK is

A way for vendors to

→ demonstrate their compliance to standard

Provide files to help vendors

→ test for errors and become compliant

Customers to assess

→ how compliant a vendor is.
Now Separating Process from Decision

(1) Independently callable through the REST API so that this integration can happen in the UI.

(2) Can call from process to decision, but this is less important..

(Taken from Fujitsu DXP Presentation)
Caller can combine decision models in a single call

Input Values

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
</tr>
</thead>
<tbody>
<tr>
<td>(in)</td>
<td>(in)</td>
<td>(out)</td>
<td>(out)</td>
<td></td>
</tr>
</tbody>
</table>

Input values and output values of the first DT are passed to the second DT

Output Values

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
</tr>
</thead>
<tbody>
<tr>
<td>(in)</td>
<td>(in)</td>
<td>(in)</td>
<td>(in)</td>
<td>(out)</td>
<td></td>
</tr>
</tbody>
</table>

Decision Table A

Decision Table B