Decision Modelling

Barbara Re
Decision Logic and Decision Task

(Ross 2011, p. 152f)
There are two well-known modeling notations for Decision Logic

• The Decision Model
  • Based on a book from Barbara von Halle and Larry Goldberg

• Decision Model and Notation DMN
  • A new standard from OMG
Decision Model and Notation (DMN)
Two Different Perspectives

• Decision-making is addressed from two different perspectives by existing modeling standards:
  
  • Business process models (e.g., BPMN) can describe the coordination of decision-making within business processes by defining specific tasks or activities within which the decision-making takes place
  
  • Decision logic (e.g., PRR, PMML) can define the specific logic used to make individual decisions, for example as business rules, decision tables, or executable analytic models.
DMN will provide a third perspective – the Decision Requirements Diagram – forming a bridge between business process models and decision logic models:

• **Business process models** will define tasks within business processes where decision-making is required to occur

• **Decision Requirements Diagrams** will define the decisions to be made in those tasks, their interrelationships, and their requirements for decision logic

• **Decision logic** will define the required decisions in sufficient detail to allow validation and/or automation
Aspects of Modelling
Decision Model and Notation (DMN)

- The Decision Model and Notation is a new standard from the OMG
- It is currently published in its version 1.0
- Purpose of DMN: provide the constructs that are needed to model decision, so that organizational decision-making can be
  - readily depicted in diagrams
  - accurately defined by business analysts
  - (optionally) automated
It defines the concept of a Decision Requirements Graph (DRG) comprising a set of elements and their connection rules, and a corresponding notation: the Decision Requirements Diagram (DRD)

It defines Decision Logic via Value expression and/or decision table

It provides a language called FEEL for defining and assembling decision tables, calculations, if/then/else logic, simple data structures, and externally defined logic from Java and PMML into executable expressions with formally defined semantic
Scope and Uses of DMN

• Decision modeling is carried out by business analysts in order to understand and define the decisions used in a business or organization.

• Such decisions are typically operational decisions made in day-to-day business processes, rather than the strategic decision-making for which fewer rules and representations exist.

• Three uses of DMN can be discerned in this context:
  • For modeling human decision-making
  • For modeling the requirements for automated decision-making
  • For implementing automated decision-making
Main Concepts of DMN

- Business Process
- Decision Requirements
- Decision Logic

(Coenen 2013)
Basic Concepts – Decision Requirements Level

- A **decision** is the act of determining an **output** value (the chosen option), from a number of **input** values, using logic defining how the output is determined from the inputs.

- **Decision logic** may include one or more **business knowledge models** which encapsulate business know-how.

- A decision may require multiple business knowledge models, and a business knowledge model may require multiple other business knowledge models.
Basic Concepts – Decision Requirements Level

- Decisions can be decomposed into sub-decisions. Top level decisions can be thought of as selecting an answer from a range of possible answers. Lower level decisions often will simply provide input to other decisions.

- Decisions may therefore be connected in a network called a Decision Requirements Graph (DRG), which may be drawn as a Decision Requirements Diagram (DRD)
Basic Concepts – Decision Requirements Level

- A decision may require multiple business knowledge models, and a business knowledge model may require multiple other business knowledge models.
Basic Concepts – Decision Logic Level

• Using decision logic, the same components described at decision requirements level may be specified in greater detail, to capture:
  • a complete set of business rules and calculations
  • (if desired) to allow the decision-making to be fully automated
Decision and corresponding value expression

• 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.

• The decision is considered to be the evaluation of the expression.

• The value expression may be notated using a **boxed expression**.
Business knowledge model and corresponding value expression

- At the decision logic level, a business knowledge model is defined using a value expression that specifies how an output is determined from a set of inputs.
- Value expressions may be encapsulated as functions, which may be invoked from decisions’ value expressions.
• A business knowledge model may contain any decision logic which is capable of being represented as a function
• This will allow the import of many existing decision logic modeling standards (e.g., for business rules and analytic models) into DMN
• An important format of business knowledge, specifically supported in DMN, is the Decision Table
• Such a business knowledge model may be notated using a Decision Table
Main concepts – Decision Requirements Level

- Business concepts only
- Business decisions
- Areas of business knowledge
- Sources of business knowledge

(Doenen 2013)
DRD Elements

- **Decision**: The determination of an output from a number of inputs, using decision logic, which may reference one or more Business Knowledge Models.

- **Business Knowledge Model**: A function encapsulating knowledge, such as a decision table.

- **Input Data**: Information used as an input to one or more decisions. Provides the parameters for a Business Knowledge Model.

- **Knowledge Source**: Has authority for a Business Knowledge Model or Decision.
## Constructs of a Decision Requirements Model

<table>
<thead>
<tr>
<th>Construct</th>
<th>DMN Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ELEMENTS</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Decision</td>
<td>[Decision]</td>
<td>The act of determining an output from a number of inputs, using decision logic which may reference one or more business knowledge models.</td>
</tr>
<tr>
<td>Business Knowledge Model</td>
<td>[Business knowledge]</td>
<td>A function encapsulating business knowledge, in the form of business rules, decision table or analytic model. Some of the tool may not support this element. In such case the decision logic is directly linked to the Decision rather than the business knowledge model.</td>
</tr>
<tr>
<td>Knowledge Source</td>
<td>[Knowledge source]</td>
<td>The authority for a business knowledge model or decision.</td>
</tr>
<tr>
<td>Input Data</td>
<td>[Input data]</td>
<td>Information used as an input by one or more decisions. It also denotes the parameters of a Business Knowledge Model.</td>
</tr>
<tr>
<td><strong>REQUIREMENTS</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Information Requirement</td>
<td></td>
<td>Information - input data or decision output - required for a decision.</td>
</tr>
<tr>
<td>Knowledge Requirement</td>
<td></td>
<td>The invocation of a business knowledge model.</td>
</tr>
<tr>
<td>Authority Requirement</td>
<td></td>
<td>Showing the knowledge source of an element or the dependency of a knowledge source on input data.</td>
</tr>
</tbody>
</table>
Decision

• A decision is the act of determining an output value (the chosen option), from a number of input values, using logic defining how the output is determined from the inputs.

• Two properties should be captured for every decision:
  • Question: A natural language statement that represents the decision in the form of a question. This should be specific and detailed.
  • Allowed Answers: A natural language description of the possible answers to this question.

• For action-oriented decisions, the allowed answers represent the responses that the process must handle when the decision model is invoked by a business rule task.
• Business knowledge models represent functions that encapsulate reusable decision making logic. The logic they encapsulate might be a set of business rules, a decision tree, a decision table, or an analytic model.

• The specifics of knowledge representation involved need not be displayed on the diagram but could be.

• The decision logic that can be specified in a business knowledge model can also be linked directly to a decision, but encapsulating it in a business knowledge model allows it to be reused, parameterized and displayed on a Decision Requirements Diagram.
Input Data

• Decisions require inputs, and many of these are input data, which is data that is input to the decision making from outside the decision context.

• Input data elements typically represent business entities that are being used in the decision making, such as Policy or Customer. However, sometimes they can represent any information element at any level of detail.

• Each input data element can be described in terms of a hierarchical information model that specifies exactly what information elements comprise the input data.
Decision with the Listed Input Data option
Knowledge Source

- Knowledge sources represent the source of know-how for making a decision
- This could be:
  - *Regulations or policies* about how a decision must be made, best practices or expertise on how it should be made
  - *Analytic knowledge* on how it might be made more accurate
Knowledge Sources representing authorities

- Knowledge sources are the authorities for a decision and typically refer to some external document or source that contains detailed guidance.
Knowledge source representing predictive analytics

• They may be drawn from **Input Data** and **Decisions to Knowledge Sources**, where, in conjunction with use (a), they represent the derivation of Business Knowledge Models from instances of Input Data and Decision results, using analytics.
DRD Requirements

- Shows that Input Data or Decision output is required as an input by another Decision
- Shows that a Business Knowledge Model is invoked by a Decision or another Business Knowledge Model
- Shows the dependency of a DRD Element on a Knowledge Source
Elements and Allowed Relationships of the Requirements Graph

- Decision 1 is used as input for decision 2
- Decision depends on Knowledge Source
- Business Knowledge invokes a Decision
- Business Knowledge 1 invokes Business Knowledge 2
- Input data is used as input for decision
- Input data depends on Knowledge Source
- Knowledge Source depends on Decision
- Knowledge Source depends on Business Knowledge
- Knowledge Source 1 depends on Knowledge Source 2

(Coenen 2013)
Main concepts – Decision Logic

- Greater detail
- Business rules
- Calculations
- Automated
- Display

(Coenen 2013)
This will allow the import of many existing decision logic modeling standards (e.g. for business rules and analytic models) into DMN.

(Coenen 2013)
"I'm here because my boss said we should use more decisions tables for our project. What types of decision tables do you sell?"
Boxed Expression

The notation for decision logic is **boxed expressions** which decompose the decision logic model into small pieces that are associated with elements of Decision Requirements Diagram.
Structure of a Decision Table in DMN

<table>
<thead>
<tr>
<th>Decision name</th>
<th>HC</th>
<th>Input expression 1</th>
<th>Input expression 2</th>
<th>Output name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>value 1a, value 1b</td>
<td>value 2a, value 2b</td>
<td>value 3a</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>value 1b</td>
<td>value 2b</td>
<td>value 3b</td>
</tr>
</tbody>
</table>

- **Hit policy**
- **Completeness indicator**
- **Name of Decision Table**
- **Set of Inputs**
- **Output(s)**
  - Multiple output columns allowed

*Lists of expected values (optional)*

**Rule number**

**Double line between inputs section and outputs section, and between inputs/outputs headers and the rule entry cells.**

Decision Modeling
Decision Tables

<table>
<thead>
<tr>
<th>Region of employee = Region of customer</th>
<th>Claims Expenditure (estimated)</th>
<th>Experience of employee</th>
<th>Number of open claims of employee</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>yes/no</td>
<td>Number</td>
<td>low/medium/high</td>
<td>Number</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>yes</td>
<td>[100..10000]</td>
<td>low</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>&gt; 10000</td>
<td>low</td>
<td>-100</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>&gt; 10000</td>
<td>medium</td>
<td>-100</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>[10..20]</td>
<td></td>
<td>-100</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>[20..30]</td>
<td></td>
<td>-500</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>&gt; 30</td>
<td></td>
<td>-1000</td>
</tr>
</tbody>
</table>

**Experience of Employee**

<table>
<thead>
<tr>
<th>Approval Authority</th>
<th>Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>low</td>
</tr>
<tr>
<td>2</td>
<td>medium</td>
</tr>
<tr>
<td>3</td>
<td>high</td>
</tr>
</tbody>
</table>

**Exceptional Case**

<table>
<thead>
<tr>
<th>Agent Id</th>
<th>Customer Frame Contract Id</th>
<th>Required Role</th>
<th>Special Employee</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4711</td>
<td>Special Customer Task Force Berlin</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>GB15</td>
<td>Special Customer Task Force Berlin</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>camunda</td>
<td>Mr. Important</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Required Skill / Role**

<table>
<thead>
<tr>
<th>Type of Claim</th>
<th>Private/Business Customer?</th>
<th>Required Role</th>
<th>Required Skill</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Third Party Liability</td>
<td>Private</td>
<td>Service Center</td>
</tr>
<tr>
<td>2</td>
<td>Third Party Liability</td>
<td>Business</td>
<td>Service Center</td>
</tr>
<tr>
<td>3</td>
<td>Accident</td>
<td>Private</td>
<td>Service Center</td>
</tr>
<tr>
<td>4</td>
<td>Accident</td>
<td>Business</td>
<td>Service Center</td>
</tr>
<tr>
<td>5</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Hit Policies (1)

• The hit policy specifies what the result of the decision table is, if there are multiple matches for a given set of inputs.

• The hit policy indication is mandatory and is summarized using a single character in a particular decision table cell.

Single Hit Policies:

<table>
<thead>
<tr>
<th>Hit Policy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unique</td>
<td>This is the default policy. All rules are exclusive and only a single rule is matched.</td>
</tr>
<tr>
<td>Any</td>
<td>Multiple matching rules, all matching rules with the same output. Any of these outputs can be used.</td>
</tr>
<tr>
<td>Priority</td>
<td>Multiple matching rules with different outputs. Returns the matching rule with the highest output priority which is specified in an ordered list of values, e.g. the list of expected output values.</td>
</tr>
<tr>
<td>First</td>
<td>Multiple matching rules with different outputs. First hit by rule order is returned. Once there is a hit, the evaluation stops (and ignore the rest of the rules). The matching has a dependency on the order of the rules. The last rule is often the <em>catch-remainder</em> rule. This type of policy is hard to validate manually and must be used with care.</td>
</tr>
</tbody>
</table>
Hit Policies (2)

Multiple Hits Policies for Single Output

<table>
<thead>
<tr>
<th>Hit Policy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No order</td>
<td>Returns all hits in a unique list in arbitrary order.</td>
</tr>
<tr>
<td>Output order</td>
<td>Returns all hits in decreasing priority order. Output priorities are specified in an ordered list of values.</td>
</tr>
<tr>
<td>Rule order</td>
<td>Returns all hits in rule order, i.e. dependency on the order of the rules.</td>
</tr>
</tbody>
</table>

Aggregation for Multiple Hits Policy

<table>
<thead>
<tr>
<th>Aggregation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collect</td>
<td>The result of the decision table is the list of all the outputs, ordered or unordered per the hit policy.</td>
</tr>
<tr>
<td>Sum</td>
<td>The result of the decision table is the sum of all the outputs.</td>
</tr>
<tr>
<td>Min</td>
<td>The result of the decision table is the smallest value of all the outputs.</td>
</tr>
<tr>
<td>Max</td>
<td>The result of the decision table is the largest value of all the outputs.</td>
</tr>
<tr>
<td>Count</td>
<td>The result of the decision table is the number of outputs.</td>
</tr>
<tr>
<td>Average</td>
<td>The result of the decision table is the average value of all the outputs, defined as the sum divided by the count.</td>
</tr>
</tbody>
</table>
FEEL = Friendly Enough Expression Language

FEEL is a script language for decision tables

Decision Modeling
Orientation of Rules in a DMN Decision Table

Rules as Rows:

<table>
<thead>
<tr>
<th>table name</th>
<th>input expression 1</th>
<th>input expression 2</th>
<th>Output name</th>
</tr>
</thead>
<tbody>
<tr>
<td>HC</td>
<td>value 1a, value 1b</td>
<td>value 2a, value 2b</td>
<td>value 1a, value 1b</td>
</tr>
<tr>
<td>1</td>
<td>input entry 1a</td>
<td>input entry 2a</td>
<td>output entry 1a</td>
</tr>
<tr>
<td>2</td>
<td>input entry 1b</td>
<td>input entry 2b</td>
<td>output entry 1b</td>
</tr>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
<td>output entry 1a</td>
</tr>
</tbody>
</table>

Rules as Columns:

<table>
<thead>
<tr>
<th>table name</th>
<th>input expression 1</th>
<th>input expression 2</th>
<th>Output name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>value 1a, value 1b</td>
<td>value 2a, value 2b</td>
<td></td>
</tr>
<tr>
<td>input entry 1a</td>
<td></td>
<td>input entry 2a</td>
<td></td>
</tr>
<tr>
<td>input entry 1b</td>
<td></td>
<td>input entry 2b</td>
<td></td>
</tr>
<tr>
<td>Output name</td>
<td>value 1a, value 1b</td>
<td>output entry 1a</td>
<td></td>
</tr>
</tbody>
</table>

Rules as Crosstabs:

<table>
<thead>
<tr>
<th>table name</th>
<th>input expression 1</th>
<th>input expression 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>HC</td>
<td>input entry 1a</td>
<td>input entry 2a</td>
</tr>
<tr>
<td></td>
<td>output entry 1a</td>
<td>output entry 1b</td>
</tr>
</tbody>
</table>

Decision Modeling
The Decision Model

The Decision Model

• The Decision Model was developed by Barbara von Halle and Larry Goldberg (2010)

• Objective:
  • a rigorous, repeatable, and technology-independent model of business logic that is simple to create, interpret, modify, and automate

• The Decision Model is a template for perceiving, organizing, and managing the business logic behind a business decision.

• It is a declarative representation of decision logic
  • specifies the conditions on which a decision is made
  • does not specify how the conditions are tested, in particular it does not specify the order in which conditions are tested
A Decision Model has two different kinds of diagrams:

**Decision Model Diagram**

**Rule Family Table**

<table>
<thead>
<tr>
<th>Conditions</th>
<th>Person Student Loans</th>
<th>Person Business Loans</th>
<th>Person Customer Status</th>
<th>Person Miscellaneous Loans Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is</td>
<td>Yes</td>
<td>Is not</td>
<td>Current customer</td>
<td>Is</td>
</tr>
<tr>
<td>Is</td>
<td>Yes</td>
<td>Is not</td>
<td>Current customer</td>
<td>Is</td>
</tr>
<tr>
<td>Is</td>
<td>Yes</td>
<td>Is</td>
<td>Current customer</td>
<td>Is</td>
</tr>
<tr>
<td>Is</td>
<td>Yes</td>
<td>Is</td>
<td>Current customer</td>
<td>Is</td>
</tr>
</tbody>
</table>
Decision Model Diagrams

• The root of a Decision Model diagram (its start) is an octagonal shape that represents the entire business decision
  • It is this shape that relates to tasks within business process models.

• The other nodes in the Decision Model diagram represent Rule Families

(von Halle & Goldberg 2010, p. 26f)
Decision Model Diagram

The Decision Model Diagram represents Rule Family Tables.

(von Halle & Goldberg 2010, p. 29)
Rule Family: Basic Element of the Decision Model

- Rule Family is a two-dimensional table relating conditions to one—and only one—corresponding conclusion.

<table>
<thead>
<tr>
<th>Conditions</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person Employment History</td>
<td>Person Likelihood of Defaulting on a Loan</td>
</tr>
<tr>
<td>Person Mortgage Situation</td>
<td></td>
</tr>
<tr>
<td>Person Miscellaneous Loans Assessment</td>
<td></td>
</tr>
<tr>
<td>Is</td>
<td>Poor</td>
</tr>
</tbody>
</table>

Column headings: names of facts being tested
name of conclusion being reached
operator
value of column heading

(von Halle & Goldberg 2010, p. 18f)
Rule Family Tables are Decision Tables

• A Rule Family table is a kind of decision table
  • In a Rule Family Table each row represents a rule
  • In a decision table each column represents a rule

• A Decision Model is a structured collection of decision tables

• There are some specialties:
  • A Rule Family must only have one conclusion column
  • Inferential relationships between Rule Family are made explicit in a Decision Model diagram

(von Halle & Goldberg 2010, p. 25)
Rule Family: Basic Element of the Decision Model

- A Rule Family node has three parts:
  - The name is the conclusion of the Rule Family
  - Inferred conditions: There are Rule Families with these names
  - Basic conditions: There are no Rules Families with these names

(von Halle & Goldberg 2010, p. 18f)
This Condition column is part of a logical expression interpreted as “If/when the Policy Renewal Override is Yes”

The conclusion column is part of a logical expression interpreted as “then the Policy Renewal Method is Automatic”

A discrete business logic instance is a single row in the Rule Family table

This condition column is also part of a logical expression interpreted as “If/when the Policy Tier Within Bounds is Yes”

---

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Manual Policy Override</th>
<th>Policy Tier Within Bounds</th>
<th>Policy Renewal Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>is</td>
<td>Yes</td>
<td>is</td>
</tr>
<tr>
<td>2</td>
<td>is</td>
<td>No</td>
<td>is</td>
</tr>
<tr>
<td>3</td>
<td>is</td>
<td>No</td>
<td>is</td>
</tr>
</tbody>
</table>
Translating a Rule Family into Natural Language

It is possible to convert each row in a Rule Family into a sentence that sounds natural to a business audience.

### Possible Conversions

- If/when Person Employment History is Poor and Person Mortgage Situation is Poor and Person Miscellaneous Loans Assessment is High, then the Person Likelihood of Defaulting on a Loan is High.

- A Person with Poor Employment History and Poor Mortgage Situation and High Miscellaneous Loans Assessment has a High Likelihood of Defaulting on a Loan.

- *It is obligatory that the Person Likelihood of Defaulting on a Loan is High if the Person Employment History is Poor and the Person Mortgage Situation is Poor and the Person Miscellaneous Loans Assessment is High.*

\begin{tabular}{|c|c|c|c|c|}
\hline
\textbf{Rule Pattern} & \textbf{Person Employment History} & \textbf{Person Mortgage Situation} & \textbf{Person Miscellaneous Loans Assessment} & \textbf{Person Likelihood of Defaulting on a Loan} \\
\hline
1 & Is & Poor & Is & Poor & Is & High & Is & High \\
\hline
\end{tabular}

(von Halle & Goldberg 2010, p. 20)
A Rule Family represents all Rules for one Conclusion

The Decision Model has only one Rule Family for each type of conclusion column, i.e. all rules for a conclusion are in one table.

(von Halle & Goldberg 2010, p. 29)
Rule Pattern

• A set of Rule Family rows with a common set of populated condition cells is called a Rule Pattern.

• The following Rule Family represents two rule patterns

<table>
<thead>
<tr>
<th>Rule Pattern</th>
<th>Conditions</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Person Student Loans</td>
<td>Person Business Loans</td>
</tr>
<tr>
<td>1</td>
<td>Is</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Is</td>
<td>Yes</td>
</tr>
<tr>
<td>1</td>
<td>Is</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Is</td>
<td>Yes</td>
</tr>
</tbody>
</table>

1. The 1st and 3rd rule have conditions for the fact types “Person Student Loans” and “Person Customer Status”.

2. The 2nd and 4th rule have conditions for the fact types “Person Business Loans” and “Person Customer Status”.

(von Halle & Goldberg 2010, p. 24)
Two dependend Rule Families

- Conditions of one rule family can depend on another rule family
- Example: Person Employment History in the first rule family depends on
  - Person Years at Current employer &
  - Person Number of Jobs in Past Five Years

<table>
<thead>
<tr>
<th>Rule Pattern</th>
<th>Conditions</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person Employment History</td>
<td>Person Mortgage Situation</td>
<td>Person Likelihood of Defaulting on a Loan</td>
</tr>
<tr>
<td>1</td>
<td>Is Poor</td>
<td>Is High</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rule Pattern</th>
<th>Conditions</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person Employment History</td>
<td>Person Years at Current Employer</td>
<td>Person Employment History</td>
</tr>
</tbody>
</table>

(von Halle & Goldberg 2010, p. 23)
• The Decision Model diagram shows the inferential relationships between Rule Families.
  - Solid lines between Rule Family nodes represent inferential relationships.
  - The name of the node at the end with the dot occurs as condition in the other node.
  - Leave nodes in a Decision Model diagram to not have inferred conditions.

(von Halle & Goldberg 2010, p. 26f)
The Rule family directly connected to the business decision shape is called the “Decision Rule Family”, its conclusion is the conclusion sought by the entire Decision Model.

A Decision Model diagram begins with an octagonal shape that represents the entire business decision. The other shapes in the Decision Model diagram represent Rule Families. This diagram has 6 Rule Families.

The name of each Rule Family is its conclusion column heading.

von Halle & Goldberg 2010, p. 28
The solid line terminated by the dot connects Rule Families that have an inferential relationship: The conclusion of one Rule Family is used as a condition in another.

The dotted line:
The labels below the dotted line denote condition column headings that do not serve as a conclusion column heading in another Rule Family.

Inferred Conditions

Inferred Conditions

Conditions based on facts

(von Halle & Goldberg 2010, p. 28)
(von Halle & Goldberg 2010, p. 28)
The Decision Model vs. DMN: Diagrams

- On the graphical level, the Decision Model Diagram is a subset of DMN's Decision Requirements Diagram
  - Decision is in both diagrams
  - Rule Family corresponds to Business Knowledge Model
- DMN is more expressive; compared to the Decision Model Diagram contains Input data and Knowledge Sources
The Decision Model vs. DMN: Decision Tables

• Decision Model and DMN use decision tables to represent the decision logic.
• The main structural differences are down to the split cell versus single cell convention for the operator and operand.
• The semantics of decision tables in DMN is more expressive: It can return multiple values and can specify, how multiple values are aggregated.

Decision Tables in DMN and TDM

DMN Decision Table

<table>
<thead>
<tr>
<th>Student Course Eligibility Table</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A: Student Total Courses Enrolled Count</td>
<td>Course Total Students Currently Subscribed</td>
<td>Student Course Eligibility</td>
</tr>
<tr>
<td>1 &lt;= 4</td>
<td>Course Subscription Threshold</td>
<td>&quot;Eligible&quot;</td>
</tr>
<tr>
<td>2 &gt; 4</td>
<td>-</td>
<td>&quot;Ineligible&quot;</td>
</tr>
<tr>
<td>3 -</td>
<td>&gt;= Course Subscription Threshold</td>
<td>&quot;Ineligible&quot;</td>
</tr>
</tbody>
</table>

TDM Rule Family View

<table>
<thead>
<tr>
<th>Student Course Eligibility Table</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Row ID</td>
<td>Rule Pattern</td>
<td>Student Total Courses Enrolled Count</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>is less than or equal to</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>is greater than</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>is greater than or equal to</td>
</tr>
</tbody>
</table>

Key Features
- Single-hit decision table (returns a single value)
- Hit-Policy: Any (signified by the 'A' in the top-left corner)
- Possible domain values are provided optionally underneath the name of the expression
- No specific order of reading

Conclusion Fact Type: Regular (returns a single value)
- A RFV has no concept of a hit-policy per se; any row can always be hit
- The Rule Pattern denotes the pattern of cells populated. There are three different patterns in this RFV
- No specific order of reading
- The operator is spelled out in words - this is not mandatory, but is designed to be more business-friendly
Literatur


• Broom, N., 2014. *The Decision Model and Notation (DMN) standard - A worked example*. A web version is on http://visionalysis.co.uk/2014/01/08/going-dmn-tal/