* 50 ways to boost your business & decision analysis

Jan Vanthienen
KU Leuven (Belgium)
Leuven Institute for Research in Information Systems
jan.vanthienen@kuleuven.be

Research and teaching:
Business rules, processes and information
decision models and tables
Data & Process analytics
Smart Business
To visualize/improve the Business

* **Document agreement** on what the Business wants to accomplish
* **Remove ambiguity**
* **Enable the “Big Picture” view of the Business**

To automate the Business

* **Driving software development with models**
* **Creating software directly from models**
Overview

• Business process concerns
• Decisions and processes
• Decision model & notation
• Decision tables
• Decision processes
Business process concerns
What should be in the process model?

Exceptions?  
Timers?  
Happy path?  
Decisions?  
Decision logic?  
Roles?  
Messages?  
Notifications?  
Triggers?  
Conditions?  
...
Decisions are important for business, not only processes. Why would we only model the processes?

* Where is the decision?  
  How is the decision logic modeled?

* Model the Decision activity: Decide acceptance
Strategic decisions
• Do we enter the insurance market?
• Should we sell travel insurance?

Tactical decisions
• Which products will we promote?
• How to measure and manage performance?

Operational decisions
• How to handle routine cases?
• Follow known rules

Decision execution
• Handle the cases
• Execute the model
Decision trees should not be process paths

* Do not hardcode decision rules into the process model
* Separating (decision) rules from the process simplifies the process
* Simplify nested decision paths: Decide applicant type

Applicant type depends on:
- Age
- (and in some cases also Medical Record)
Nested decision paths

(Source: Decision Management Solutions)
Separate the rules from the process.
Decision models are not lower level details of one process

* **Decisions models can span over multiple activities, and even multiple processes**

* **Separation of concerns**
**Observation 4: model decisions**

Decision(s) (rules) need to be modeled

* A standard for processes (BPMN) is not enough

* Upcoming Decision Modeling & Notation standard (DMN)
Good decision table models are a proven technique to represent decision rules

*Consistency, completeness and correctness by design*

**Applicant Risk Rating**

<table>
<thead>
<tr>
<th>U</th>
<th>Applicant Age</th>
<th>Medical History</th>
<th>Applicant Risk Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>&gt; 60</td>
<td>good</td>
<td>Medium</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>bad</td>
<td>High</td>
</tr>
<tr>
<td>3</td>
<td>[25..60]</td>
<td>-</td>
<td>Medium</td>
</tr>
<tr>
<td>4</td>
<td>&lt; 25</td>
<td>good</td>
<td>Low</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>bad</td>
<td>Medium</td>
</tr>
</tbody>
</table>
* Observation 6: the decision process

Sometimes the entire process is about a decision

* Model the decision first, and then think about how to execute it
* The same decision can be processed in many ways
* The process of making a decision depends on the desired criteria

(throughput, efficiency, customer comfort, ...)

(c) Jan Vanthienen, 2015
There are many more business rules than decision rules

*Behavioral rules & constraints, timing rules, task allocation rules, ...*

*If all you have is a hammer ...*

*Separate rules and processes*
Business rules constrain and guide the process(es)

- *The Process - Rule continuum*
- *Declarative process modeling, smart BPM, smart decisions, smart business*
Multiple models

*The blind men and the elephant.* Poem by John Godfrey Saxe
(Cartoon G. Renee Guzlas, artist).
Decisions and processes
Decisions need to be modeled
Process and Decision Model
* Why separate decisions-processes?

* Simpler processes
* Different responsibilities, different stakeholders
* Different timing of changes
* Improved agility (change decision and keep process)

* Simpler decision modeling and discovery
* Reuse decisions across processes
* Improved visibility and focus
* Automate manual decisions
Decision model & notation
DMN components

Business Process Model (BPMN)

Decision Model (DMN)

Eligibility rules

Decision Logic Level

(c) Jan Vanthienen, 2015
Decision Requirements Graph

- Eligibility
- Policy rules
- Application form
- Risk
- Score model
- Customer behaviour

- Policy group
- Decision
- Business knowledge
- Knowledge source
- Input data

Information requirement
Knowledge requirement
Authority requirement
What is to be decided? Possible outcomes?

Decisions require:
- Input data
  - Transactions
  - Master data
  - External data
- Decision logic
  - Rules, knowledge
  - Policies
  - Analytics
- Outcome of other decisions
  - Reusability

The process of decision making?
Example: decision model
* Natural language
  * Unclear, ambiguous

* Logic
  * Powerful, unambiguous, but not for business people

* Structured English Rules
  * Subset of natural English
  * Trade-off between:
    * easy of use (but not very powerful)
    * powerful (but difficult to use)

* Decision trees, tables, graphs, diagrams
  * Different representations for different purposes:
    acquisition, V&V, decision making, dependencies, impact analysis

* Object Constraint Language
  * Part of UML
  * Useful for pre- and postconditions
To Quit or not to Quit

If I don't get a raise of at least 10 percent, I will find a job somewhere else. But if I get promoted, then I will expect my own office or I'll quit, unless the work is going to be more interesting; in which case, I'll stay with just a 10 percent raise.

<table>
<thead>
<tr>
<th></th>
<th>Y</th>
<th>N</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Promotion ?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Own office ?</td>
<td>Y</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>Raise of $\geq 10%$ ?</td>
<td></td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>More interesting work ?</td>
<td></td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>Quit</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Stay</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>
* **Decision tables (DT)**
  * Decision rules in a tabular format

* **Decision table methodology (DTM)**
  * How to use a constrained form of decision tables in order to model decisions
  * Goal-oriented decision modeling network
  * Good decision table design
  * Single hit tables (complete, consistent and correct), relations between tables, table notation, contraction, optimization, normalization.

* **Decision Modeling & Notation (DMN) standard**
  * Standard syntax and notation for exchange
  * Recognize other forms of tables
  * Combine tables with other concepts in decision modeling
  * Standard expression language
The global model

Decision table hierarchy
A condition subtable returns the outcome of a decision and uses it in another table

The detailed model

Decision table construction
Kinds of tables
* Decision tables

* Single hit (returns 1 rule with outcome(s))

  * Default:
    If rules are non-overlapping: unique hit

  * Recognize others:
    If rules are overlapping, the 1 rule has to be selected: any hit, first hit, priority hit

* Multiple hit (returns a list of rules)
DMN identifies different table types, indicated by the first letter:

* **unique hit** tables: every input case is included in one rule only. There is no overlap between rules.  
  Good

* **any hit** tables: every input case may be included in more than one rule, but the outcomes are equal. Rules are allowed to overlap.  
  Ugly

* **priority hit** tables: multiple rules can match, with different outcome values. This policy returns the matching rule with the highest output value priority (e.g. highest discount).  

* **first hit** tables: multiple (overlapping) rules can match, with different outcome values. The first hit by rule order is returned (and evaluation can halt). This is a common usage, because it resolves inconsistencies by forcing the first hit. It is important to distinguish this type of table from others because the meaning depends on the sequence of the rules. Because of this sequence, the table is hard to validate manually and therefore has to be used with care.  
  Bad
Tables with redundancy (ugly)

Overlapping rows (but with the same conclusion)

<table>
<thead>
<tr>
<th>TypeOfOrder</th>
<th>CustomerLocation</th>
<th>TypeofCustomer</th>
<th>SpecialDiscount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Web</td>
<td>US</td>
<td>Wholesaler</td>
</tr>
<tr>
<td>2</td>
<td>Phone</td>
<td>non-US</td>
<td>Retailer</td>
</tr>
<tr>
<td>3</td>
<td>non-US</td>
<td>non-US</td>
<td>Retailer</td>
</tr>
<tr>
<td>4</td>
<td>Phone</td>
<td>retail</td>
<td>Retailer</td>
</tr>
</tbody>
</table>

-> multiple rows can apply: what if one is changed? **Contradiction**
This is a list of rules, not a good decision table (where is Phone, non-US?)

The better version:

<table>
<thead>
<tr>
<th>TypeOfOrder</th>
<th>CustomerLocation</th>
<th>TypeOfCustomer</th>
<th>SpecialDiscount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Web</td>
<td>US</td>
<td>Wholesaler</td>
</tr>
<tr>
<td>2</td>
<td>Phone</td>
<td>non-US</td>
<td>Retailer</td>
</tr>
<tr>
<td>3</td>
<td>non-US</td>
<td>non-US</td>
<td>Retailer</td>
</tr>
<tr>
<td>4</td>
<td>Phone</td>
<td>retail</td>
<td>Retailer</td>
</tr>
</tbody>
</table>

(c) Jan Vanthienen, 2015
First hit table (overlapping rows but with different conclusions)

-> multiple rows can apply: take the first hit (requires sequence!)
Hard to validate. This is a list of rules, not a good decision table

The better version:

<table>
<thead>
<tr>
<th>TypeOfOrder</th>
<th>CustomerLocation</th>
<th>TypeOfCustomer</th>
<th>SpecialDiscount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web</td>
<td>US</td>
<td>Wholesaler</td>
<td>10%</td>
</tr>
<tr>
<td>Phone</td>
<td>Not Applicable</td>
<td>Retailer</td>
<td>5%</td>
</tr>
<tr>
<td>non-US</td>
<td>-</td>
<td>-</td>
<td>Not Applicable</td>
</tr>
<tr>
<td>Phone</td>
<td>-</td>
<td>-</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>
**The decision table as relation**

- **Exclusivity**

  - **Completeness**
### Applicant Risk Rating

<table>
<thead>
<tr>
<th>Applicant Age</th>
<th>Medical History</th>
<th>Applicant Risk Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt; 60</td>
<td>good</td>
<td>Medium</td>
</tr>
<tr>
<td>&gt; 60</td>
<td>bad</td>
<td>High</td>
</tr>
<tr>
<td>[25..60]</td>
<td>-</td>
<td>Medium</td>
</tr>
<tr>
<td>&lt; 25</td>
<td>good</td>
<td>Low</td>
</tr>
<tr>
<td>&lt; 25</td>
<td>bad</td>
<td>Medium</td>
</tr>
</tbody>
</table>

### Applicant Risk Rating

<table>
<thead>
<tr>
<th>Applicant Age</th>
<th>Medical History</th>
<th>Applicant Risk Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 25</td>
<td>good</td>
<td>Low</td>
</tr>
<tr>
<td>[25..60]</td>
<td>bad</td>
<td>Medium</td>
</tr>
<tr>
<td>&gt; 60</td>
<td>-</td>
<td>High</td>
</tr>
</tbody>
</table>

### Applicant Risk Rating

<table>
<thead>
<tr>
<th>Applicant Age</th>
<th>Medical History</th>
<th>Applicant Risk Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 25</td>
<td>good</td>
<td>X</td>
</tr>
<tr>
<td>[25..60]</td>
<td>X X X</td>
<td>X</td>
</tr>
<tr>
<td>&gt; 60</td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

(c) Jan Vanthienen, 2015
16 Criteria for good decision table models

* Structure and content
  * Basic structure and the importance of completeness and consistency
  * Extended entry tables (multi-valued conditions)
  * Exclusivity and completeness of the condition entries (domain partitioning)
  * **Exclusivity** and completeness of the columns (types of tables)
  * Types of actions

* Form, conciseness and readability
  * Horizontal or vertical format
  * The order of the rules is irrelevant, but not for humans
  * Group-oriented contraction
  * The order of the conditions is irrelevant, except for display size
  * Row order optimization
  * Tree notation
  * Block-oriented notation
  * Representing relations between conditions (indicating impossible values)

* Factoring and normalization
  * Subtables
  * Intertabular verification

* Purpose and pragmatics
  * Decision logic and structure
2011: Gwen (from insurance):

“Jan, I attended your presentation last year, and we applied the ideas you told us. I must thank you. What used to take 5 people for 5 months is now done by 1 person in 2 weeks or less. And …”

2012: Gwen & Kate:

Actually, it saves the company 2.9 million $ … a year

2013: Hey Jan, good news, we got promoted this year.
**Issues DMN solves**

*Separating decisions and processes*
*Using a standard modeling notation.*
Separating decision structure and decision logic

Allows to model decision relations, even if not all logic is expressed in tables.
Issues DMN solves

* **Decision modeling methodology**
  * Keep the insights of the past and avoid confusion.
  * **Good** decision table models are a proven technique to represent decision rules

  *Consistency, completeness and correctness by design*
Issues DMN solves

* **Decision table types**
  * Recognize, and unambiguously exchange.
  * DMN allows multiple table types (good for exchange)
    * Unique hit, Any hit (like in some methodologies), First hit (like in some tools), Multiple hit, …
    * Good decision table models are much stricter
  * Decision table methodology is a methodology, DMN is not
    * 16 principles

* There is nothing new in DMN about decision tables, but DMN allows to standardize and recognize other decision table formats and avoids ambiguity about the concept.
**Standard notation for exchange and implementation**

- Strict notation and simple expression language ((S-)FEEL).
- FEEL (“Friendly Enough Expression Language) implements the required mechanisms
  - Built-in types, functions and operators
  - Every decision in a DRD can be defined using a formal expression that specifies how the decision’s output is determined from its inputs
  - Complete decision models can be defined
  - Formal expressions may also be encapsulated as functions
  - Supports abstraction, composition, and scalability

- S-FEEL (“Simple FEEL”) is a basic subset of FEEL designed to cover the essential requirements of Decision Table-based DMN models
The Decision Process
Start each individual decision activity as soon as all its preconditions are fulfilled.

Avoid superfluous decision activities (unnecessary work).

Group customer contacts.
Classic References


