Business Logic and Process Logic

Knut Hinkelmann
Combining Structured Processes and Knowledge Work

We deal with two special kinds of modeling knowledge work

- **Business Logic**
  - Modeling knowledge work as a set of rules, e.g. decision rules
  - Corresponds to the Transaction and Expert Model of Davenport (2010)

- **Case Management**
  - No predefined process flow
  - Can occur as a (ad-hoc) BPMN or CMMN subprocess
  - Corresponds to the Collaboration Model of Davenport (2010)
Distinguishing Process Logic and Business Logic

- The process model contains the process logic
- Business logic can be assigned to tasks in the process model (in particular to knowledge-intensive tasks)

The business logic can occur in different forms:
- implicit in head of people
- as text (e.g. guidelines)
- as decision model
- coded in an application
Business Processes and Business Rules

- A process model represents the flow of a process (= process logic)
- Business Logic should be separated from the process logic.
- Business Logic can be represented as Business Rules.
- In BPMN there is the task type "Business Rules". This type can be used to label knowledge-intensive tasks.
Business Processes and Business Rules

Source: Rainer Endl (2004): Regelbasierte Entwicklung betrieblicher Informationssysteme, EUL-Verlag, S. 16
siehe auch:
Business Rules guiding a Business Process

- Business Rules represent Business Logic.
- Typically, business rules can guide a business process via:
  - gateways
  - activities
Business Rules guiding Gateways

- **Example:** Consider the business process Seat Party. It contains three gateways, which could be guided by business rules.

**Reservation Seating:** It is obligatory that a party is seated if the party has a reservation and a table is available and the table has appropriate size for the party.

**Regular Seating:** It is obligatory that a party is seated if the party includes a regular and a table is available and the table has appropriate size for the party.
Business Rules guiding Activities

- An activity is guided by a business rule when the business rule helps the person performing the activity do his or her work.

- Example: The business rule **Large Party Gratuity** is used by a server when she performs the activity **Produce Bill**.
  - If it is a manual activity, the server might include a line item on the bill for the mandatory gratuity
  - If a software application is used to produce the final bill, the application must implement the **Large Party Gratuity** rule.

**Large party gratuity:** It is obligatory that the gratuity is at least 15% if the gratuity is applied to a bill and the bill is incurred by a party and the party is greater than 7 people.
Separating Business Rules from Process Models and Software Systems

(von Halle & Goldberg 2010, p. 71)
## Representations of Process, Business and Data Logic

<table>
<thead>
<tr>
<th>Process Logic</th>
<th>Presentation Logic</th>
<th>Business Logic</th>
<th>Data Logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>low</td>
<td>low</td>
<td>low</td>
<td>low</td>
</tr>
<tr>
<td>mechanisms to formalize process logic:</td>
<td>mechanisms to formalize presentation logic:</td>
<td>mechanisms to represent business logic:</td>
<td>mechanisms to formalize data logic:</td>
</tr>
<tr>
<td>• Process model (e.g. BPMN)</td>
<td>• page flow</td>
<td>• documents</td>
<td>• logical data model (e.g. via UML)</td>
</tr>
<tr>
<td>• executable process model – workflows (e.g. via XPDL)</td>
<td>• sequence diagrams</td>
<td>• checklists/guidelines</td>
<td>• physical data model (e.g. via SQL)</td>
</tr>
<tr>
<td>• Java Code</td>
<td>• Java Code</td>
<td>• use cases</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• decision models</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• decision tables/decision trees</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• production rules</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Java Code</td>
<td></td>
</tr>
</tbody>
</table>

Adapted from (Küng 2010)
**Decision Tasks in Business Processes**

- A **decision task** is a knowledge-intensive task in which some decision is made.
- Decision tasks are based on business logic.
  - The business logic that is used for decision making is called **decision logic**.
- Decision tasks can provide data for gateways.
  - At the gateway only the result of the decision should be tested (for the selection of the path) not the criteria for the decision.

![Decision Logic Diagram](attachment:image.png)

**Decision: Is the applicant eligible?**

- **Check Eligibility**
- **Applicant eligible?**
  - **yes**: **pay reimbursement**
  - **no**: **reject application**

**Decision: Which reimbursement scheme is appropriate in this specific case?**
Examples for a Business Decisions

- Process: Handling auto insurance applications
- Decision Task: Check Eligibility of Applicant
- Potential outcomes: "yes" and "no" (i.e. eligible/non-eligible)
- Decision Rules: Conditions to select the outcome for a case

Case: John Smith applies for an auto insurance

Decision Task: Check Eligibility

Outcome: John Smith is eligible for auto insurance

(Ross 2011, p. 152f)
Levels of Business Rules Expression

- For expressing rules there is a trade-off between accessibility of business meaning and desirable automation.

- Rules can be expressed on various levels:
  
  **Informal and semi-formal:** natural language statements within a limited range of patterns, e.g.
  
  It is obligatory that a credit account customer is at least 18 years old.

  **Formal:** statements conforming to a more closely defined syntax with particular mathematical/logical properties, e.g. decision tables or logical formulas.

  \[
  \{X, Y, (\text{customer } X) \ (\text{creditAccount } Y) \ (\text{holder } X,Y) \quad \Rightarrow \quad (\text{ge } (\text{age } X) \ 18) \\
  \forall x \ \forall y \ \text{Customer}(x) \land \text{CreditAccount}(y) \land \text{Holder}(x,y) \rightarrow \text{age}(x)>18
  \]

  **Technical:** Combining structured data and operators, e.g.

  CreditAccount
  
  self.customer.age >= 18

(Morgan 2002, p. 63)
Notations for Decision Rules

Decision Rules can be represented, for example, in

- SBVR Structured English/Rule Speak
- Decision Tables
- Decision Trees

Decision table:

<table>
<thead>
<tr>
<th>Conditions</th>
<th>Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Printer does not print</td>
<td>Y Y Y Y N N N</td>
</tr>
<tr>
<td>A red light is flashing</td>
<td>Y Y N N Y N N</td>
</tr>
<tr>
<td>Printer is unrecognised</td>
<td>Y N Y N Y N N</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Actions</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Check the power cable</td>
<td>X</td>
</tr>
<tr>
<td>Check the printer-computer cable</td>
<td>X X</td>
</tr>
<tr>
<td>Ensure printer software is installed</td>
<td>X X X X</td>
</tr>
<tr>
<td>Check/replace ink</td>
<td>X X</td>
</tr>
<tr>
<td>Check for paper jam</td>
<td>X X</td>
</tr>
</tbody>
</table>

Decision tree: