Goal-oriented modeling and evaluation of models

John Krogstie

Professor, IDI, NTNU

Overview of lectures

Based on

- **B3.3.5 Rule modeling**
- **A9 Krogstie, J. Using EEML for Combined Goal and Process Oriented Modeling: A Case Study**

- **Look both on an exampel of goal-modelling in enterprise modelling (today) and on evaluation of models and modeling languages (SEQUAL) (Friday)**

From week 3 - 8 perspectives to modeling

- Behavioral
- Functional
- Structural
- Goal and rule-oriented
- Object-oriented
- Social communication
- Actor/role-oriented
- Topological

Rule perspective (from week 3)

- **Description of goals/means connections**
- **Main concept: Rule (goal, constraint)**
- **A rule is something that influences the actions of a set of actors.**
- **Standard form: IF condition THEN action**
- **Examples:**
  - Rule hierarchies (goal-oriented modeling) see e.g. i* and EEML
  - Tempora
    - Rules
    - Rule-hierarchies

Advantages of declarative rule-based representations

- **Problem-orientation.**
- **Maintenance.**
- **Knowledge enhancement: ``Every organization, in as far as it is organized, acts as though its members were confronting to a set of rules only a *few of which may be explicit* . This has inspired certain researchers to look upon specification of information systems as a process of rule reconstruction**

Problems with the simple rule-format

- **Every statement must be either true or false, there is nothing in between.**
- **It is usually not possible to distinguish between rules of necessity (alethic rules) and deontic rules**
- **In many goal and rule modeling languages it is not possible to specify *who* the rules apply to.**
- **The source of the rule is often not explicit**
- **Formal rule languages have the advantage of eliminating ambiguity, but still there can be problem with comprehension.**
  - Single rules
  - Missing rule structure

Using EEML for Combined Goal and Process Oriented Modeling: A Case Study

John Krogstie

Professor, IDI, NTNU

Overview of presentation

- Case: Trial-application for the State Education Loan Fund
- Use of EEML – Extended Enterprise Modeling Language (combined with formal rule system) in the case.
- EEML supported in METIS
- SEQUAL for evaluation of model and modeling languages
- High-level evaluation of trial results – focus on evaluation of models
- Summary

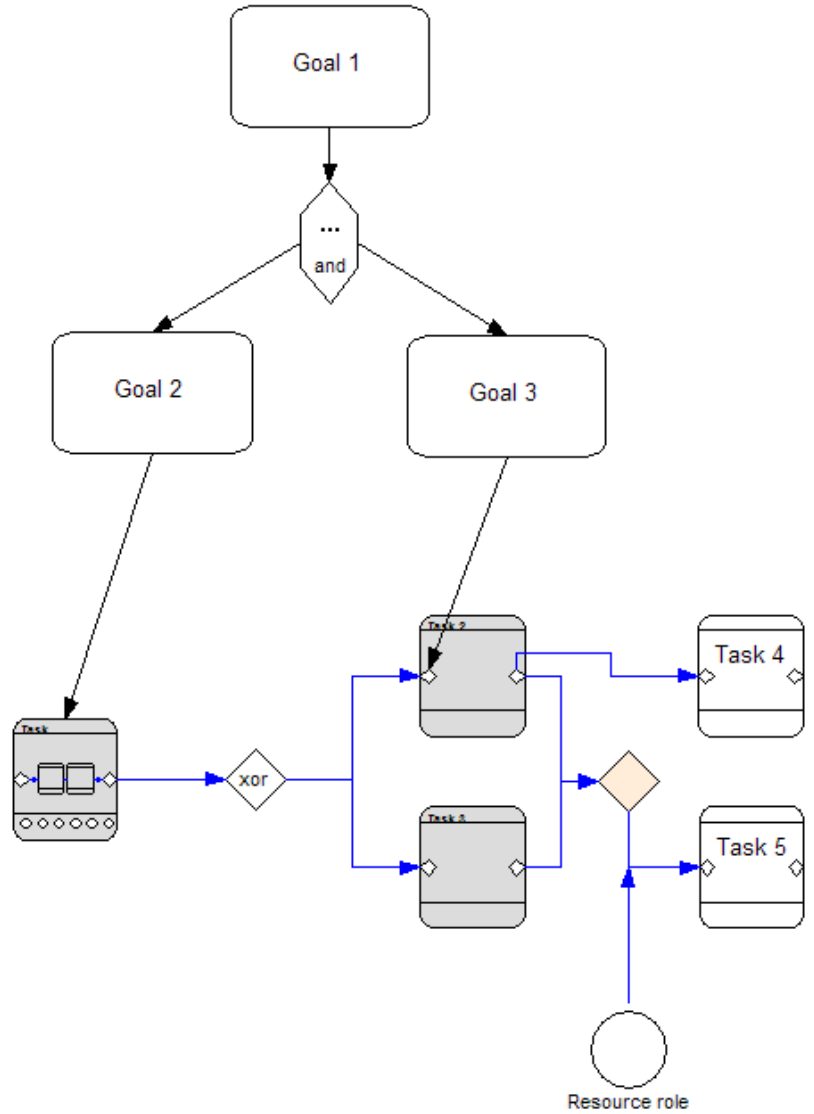
Case: Loan administration system

- **State Education Loan Fund**
- **This case was part of deciding a new overall architecture for their main application (PoC – Proof of concept)**
- **Want to improve automation of application handling. Must provide reasoning for the decisions being made**
- **Goals for rule-modeling in the loan fund**
  a) Support quick implementation of new rules, as these are changed regularly through the political process.
  b) Be able to analyze the consequences of proposed changes in the laws and regulations (with the politicians and department officials)
  c) Make it easier to maintain and evolve the rule base, including the more detailed internal rules.
  d) Support the education and training of the employees at the Loan Fund.
- **A and C -> Blaze Advisor**
- **B and D ?**

Architecture in State loan fund - solution

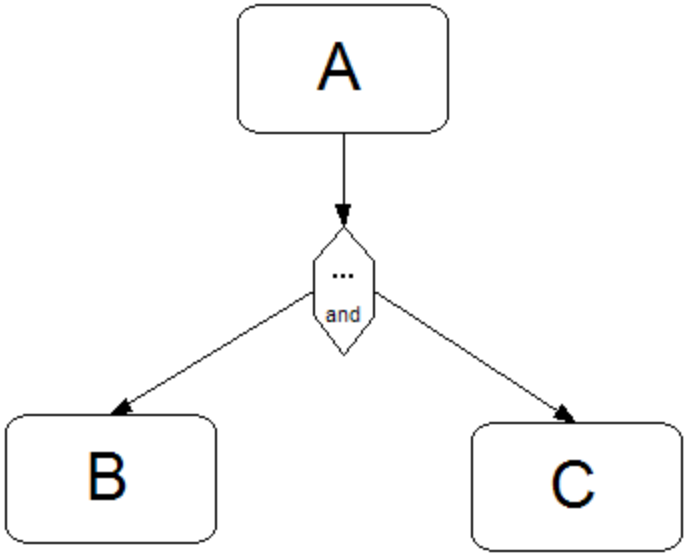
- Case-processing system
- Web-front end to enter applications
- Cases handled automatically or manually depending on if enough information is available
- Data from other sources (other organizations e.g. from tax authorities, prison authorities, universities, military etc.)
- Rule engine, automatic treatment of applications, giving rules and data are available

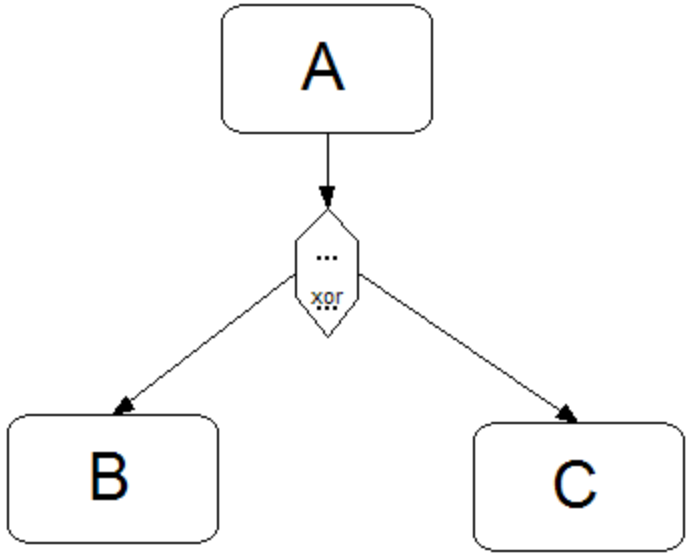
Adressing rule analysis in addition to rule execution – EEML – Extended Enterprise Modelling Language

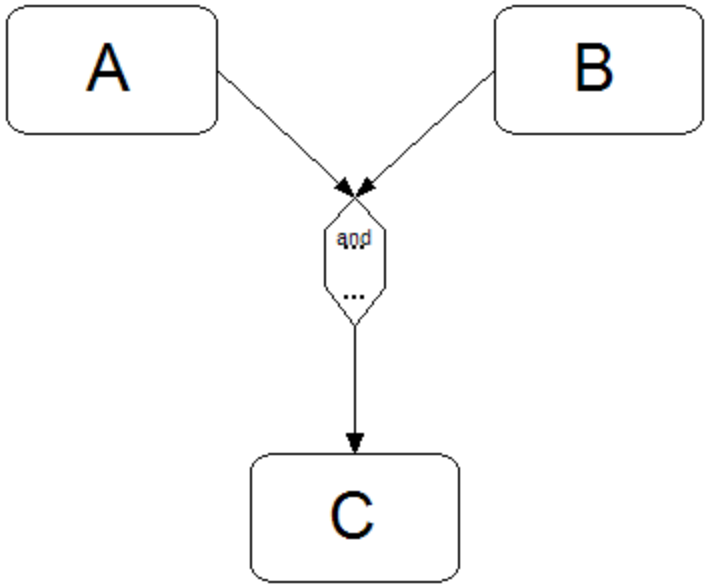


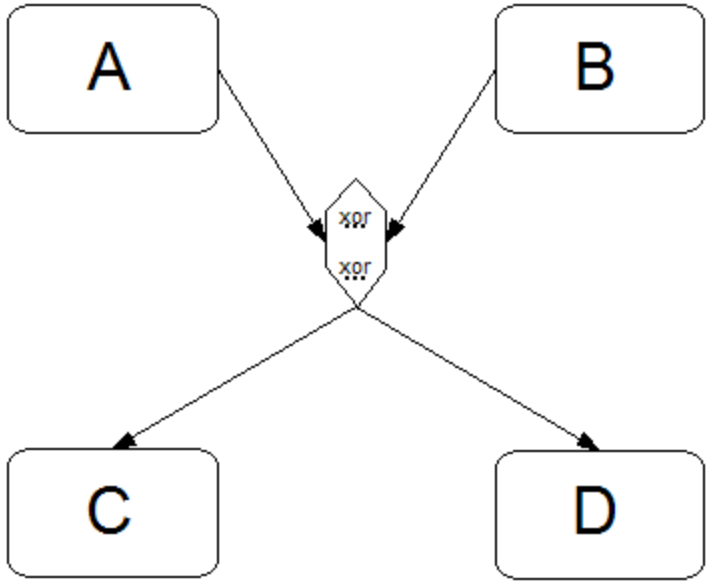
Goal modelling in EEML

- **Goal**
  - If context then deontic operator achieve state
  - *If age of person is below 18 it is forbidden for that person to drive a car*
  - Deontic or alethic operator: Necessitate, obligate, recommend, permit, discourage, forbid, contradict
- **Goal connectors for more advance goal-hierarchies**
- **Goal-relationships**
  - goal deontic operator goal (argument)
- **Goal applies to task/milestone/resourcerole/resource**
- **Goal is action-rule for task**
- **Goal is precondition/decision rule/postcondition**
- **Role/resource source of goal**


A
...
and
B
C


A
...
xor
B
C

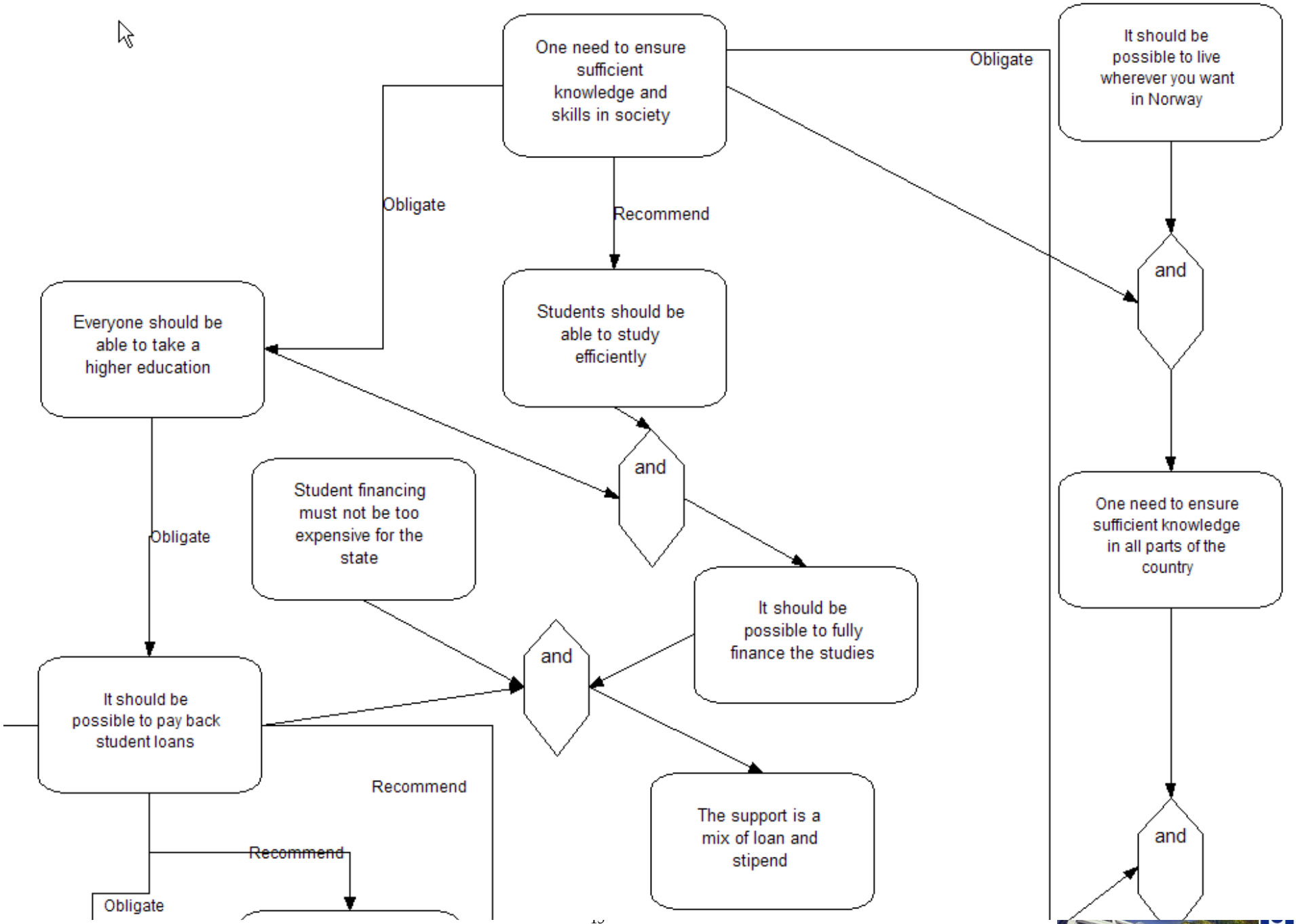

A
B
and
...
C

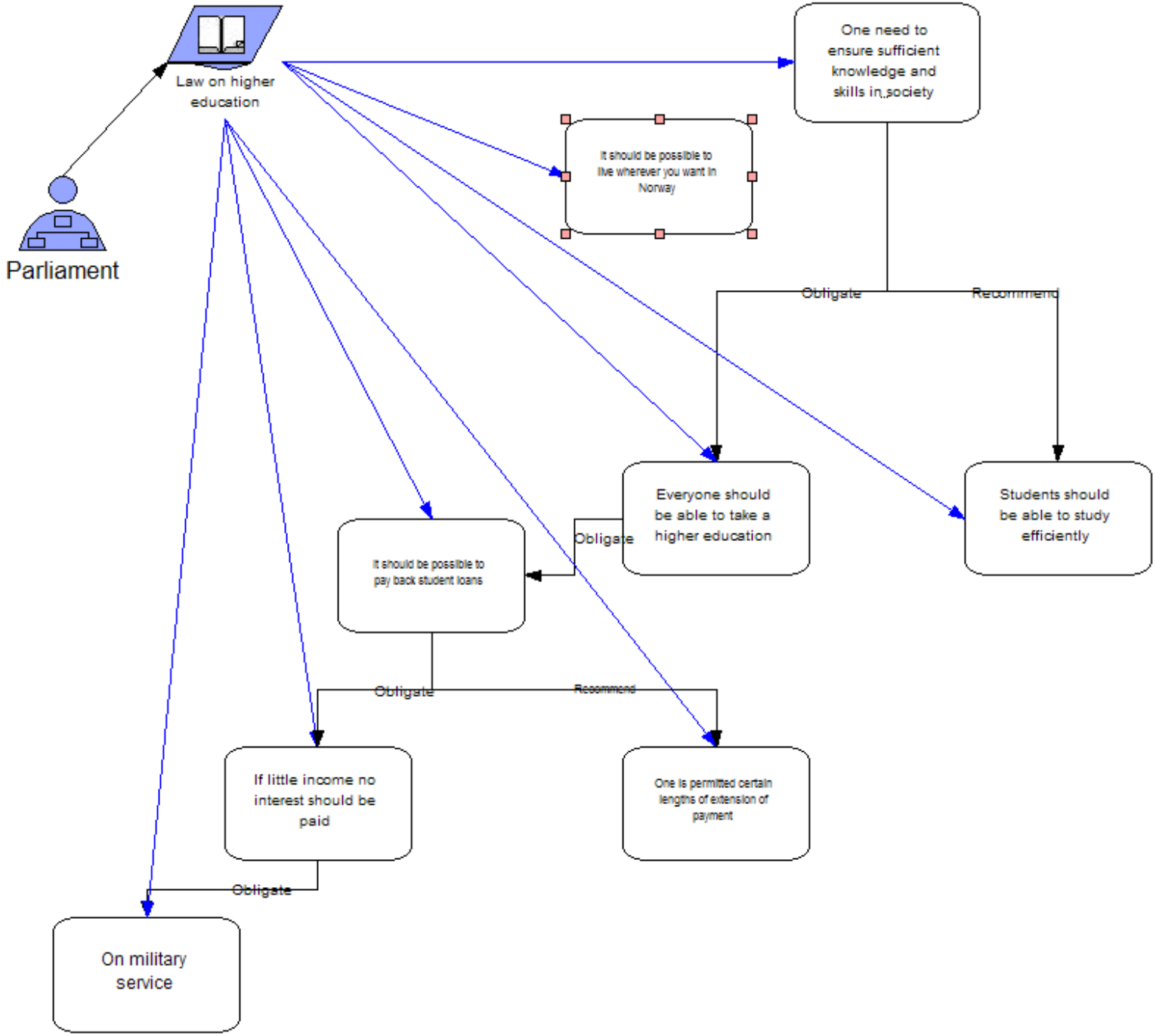

A
B
xor
xor
C
D

Levels of rule-making in the State Loan Fund case

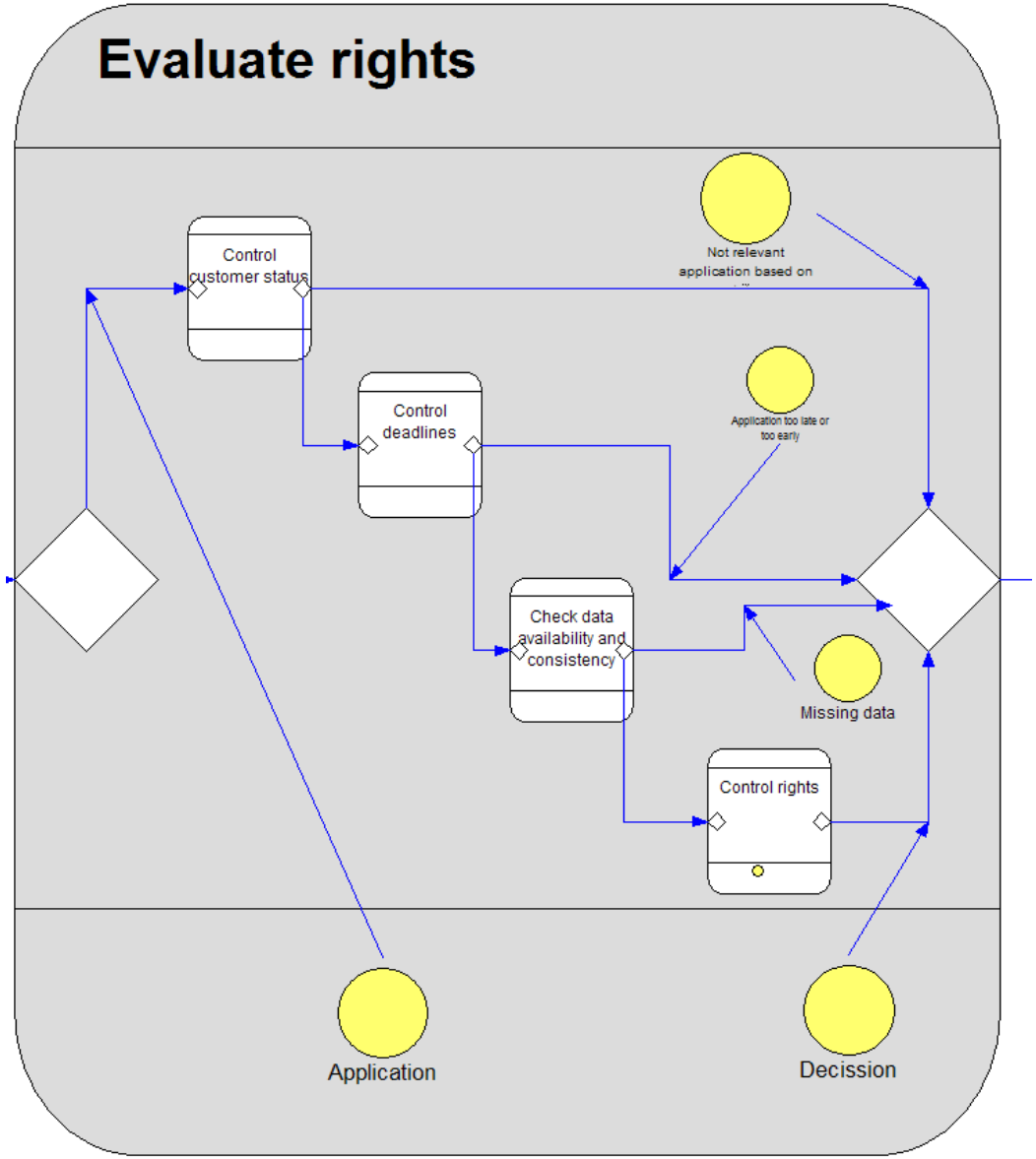
- Parliament - Law
- Ministry - Regulation
- Loan Fund Officials - Practice
- Programmers/ rule engineers - Automation

Goal hierarchy

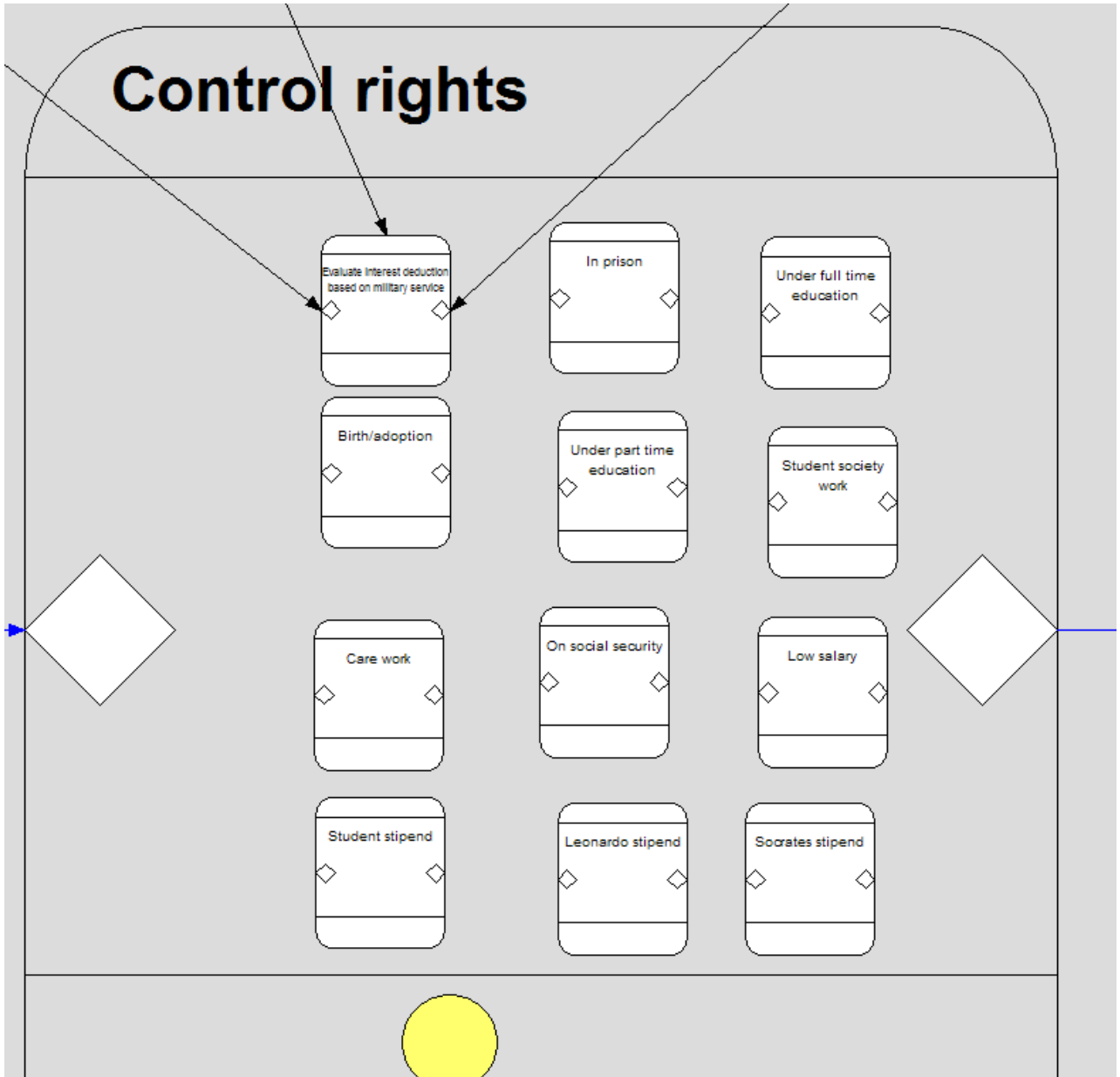




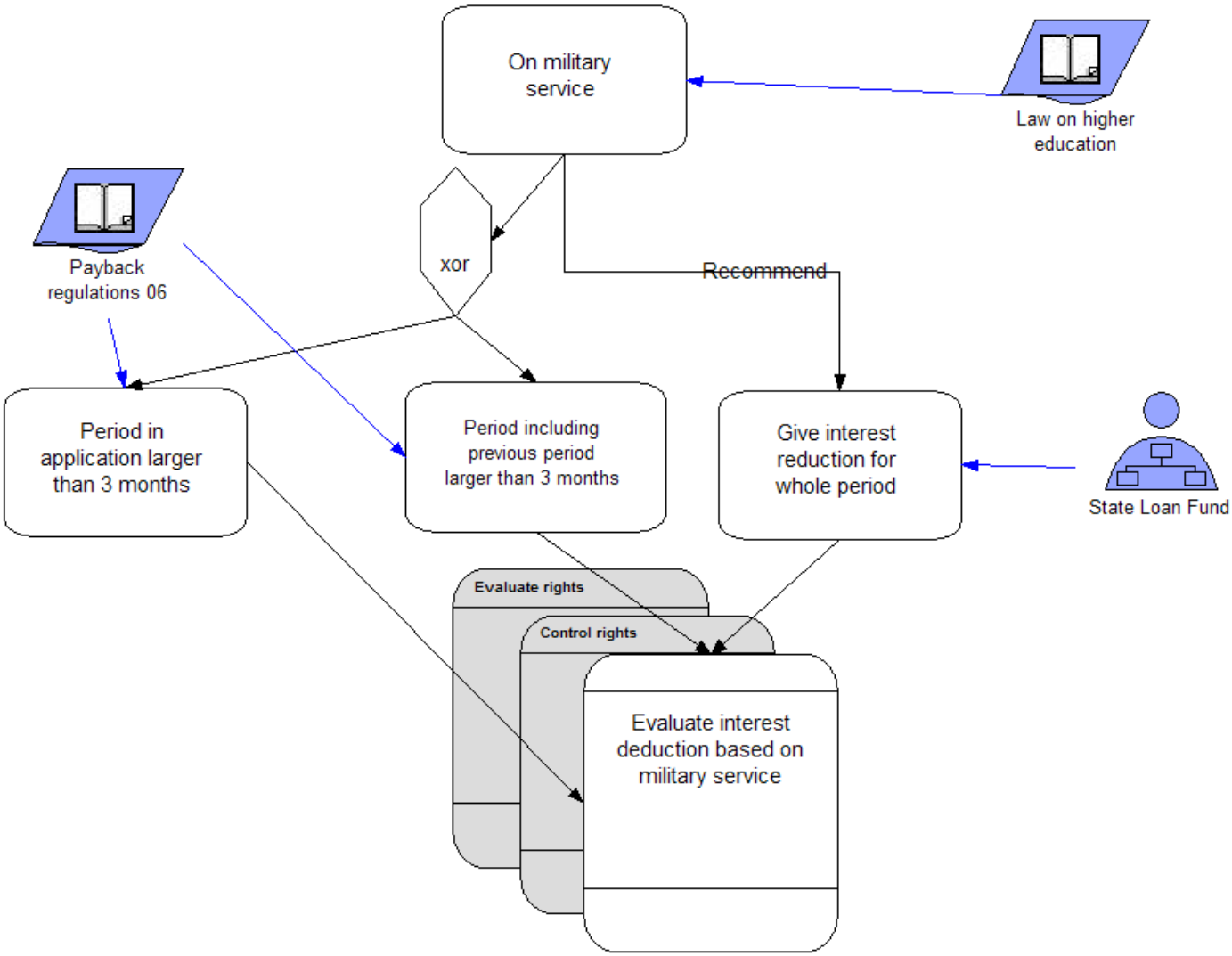
Control application for not paying interest



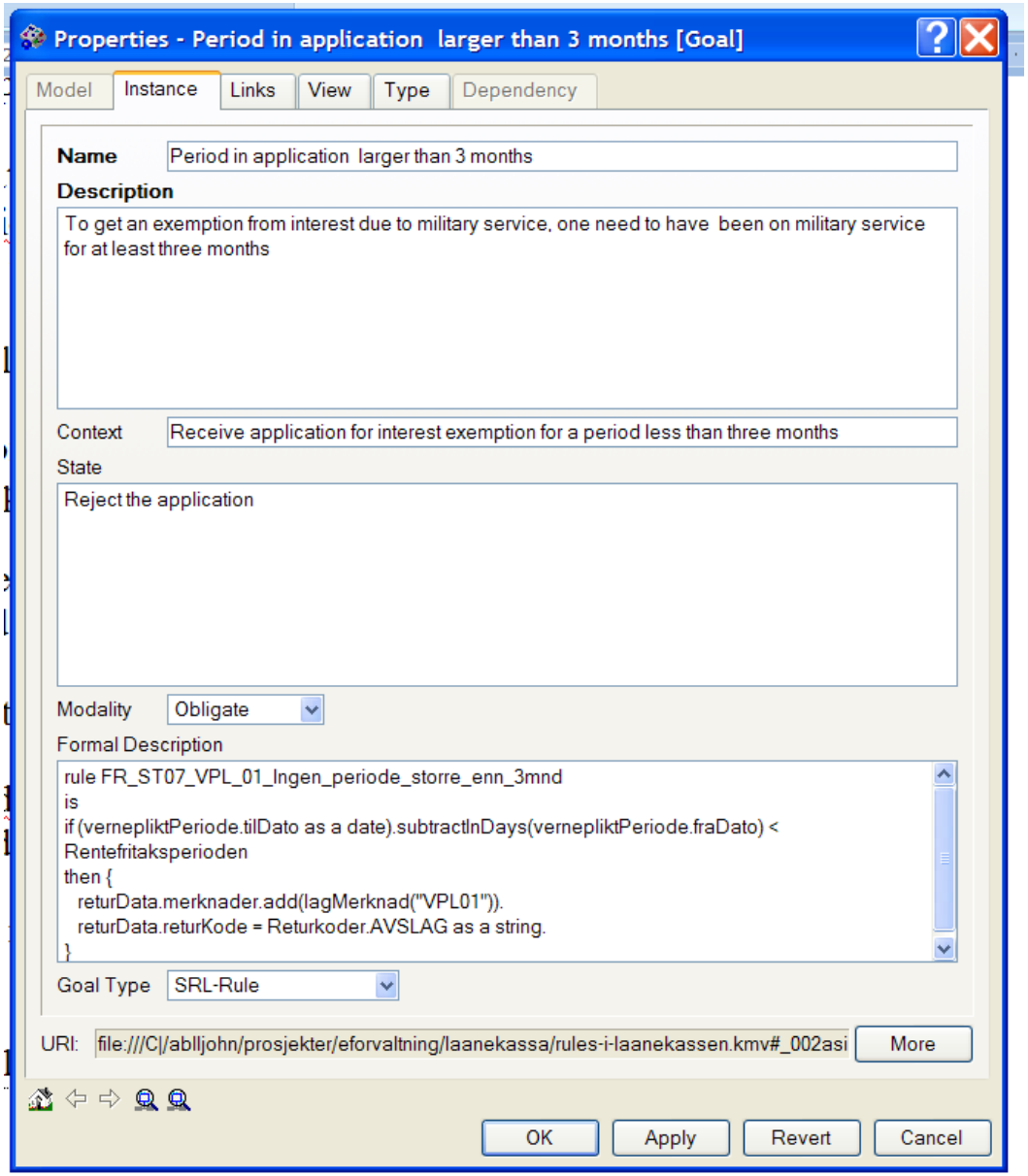
Different reasons for not paying interest



Goals vs. processes



Detailed execution rule revisited



High-level evaluation

- **Based on piloting the approach**
- **Limited time/resources available**
- **Structured according to SEQUAL**
  - Evaluation of rule language
  - Evaluation of rule-model
- **First evaluation based on interviews with different stakeholders and models/documentation provided**
- **Results/interpretations structured and presented to stakeholders for comments**

SEQUAL – A framework for understanding and assessing quality of models based on semiotics

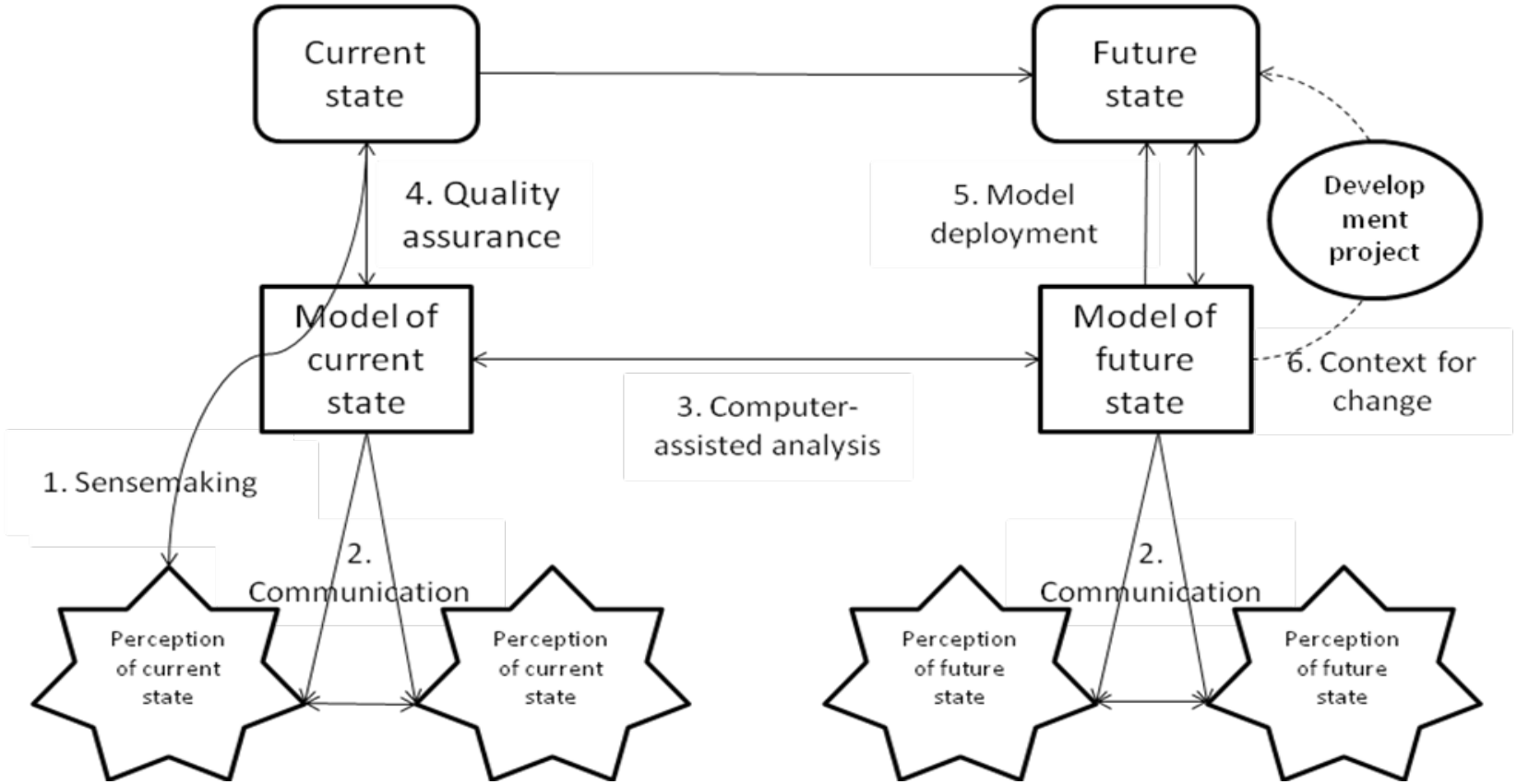
- For models as a knowledge representation in general
- Can be extended and specialised towards specific types of model and modelling languages
- Differentiate between goals of modelling (quality characteristics) and means to achieve these goals
- Set-oriented definition to enable a formal discussion of the different quality levels
- Differentiate between quality of different levels based on semiotic theory
- Takes into account that models are socially constructed

Sets in the quality framework

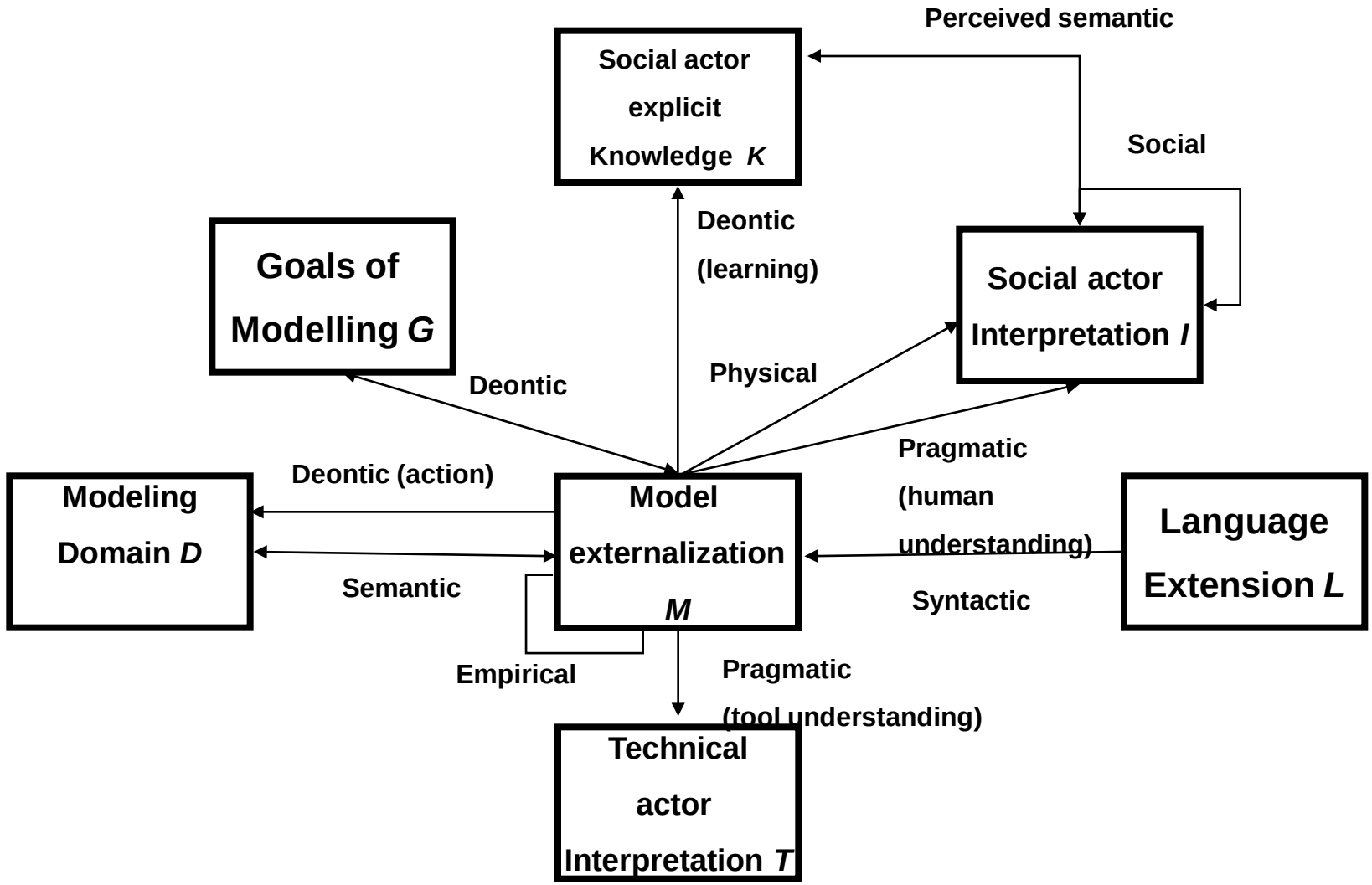
- A: Actors that develops or has to relate to (parts of) the model. Can be persons or tools (technical actors).
- L: What can be expressed in the modelling language
- M: What is expressed in the model
- D: What can be expressed about the domain (area of interest)
- K: The explicit knowledge of the participating persons
- I: What the persons in the audience interpret the model to express
- T: What relevant tools interpret the model to say
- G: The goals of the modelling

All of these sets evolves as part of modelling

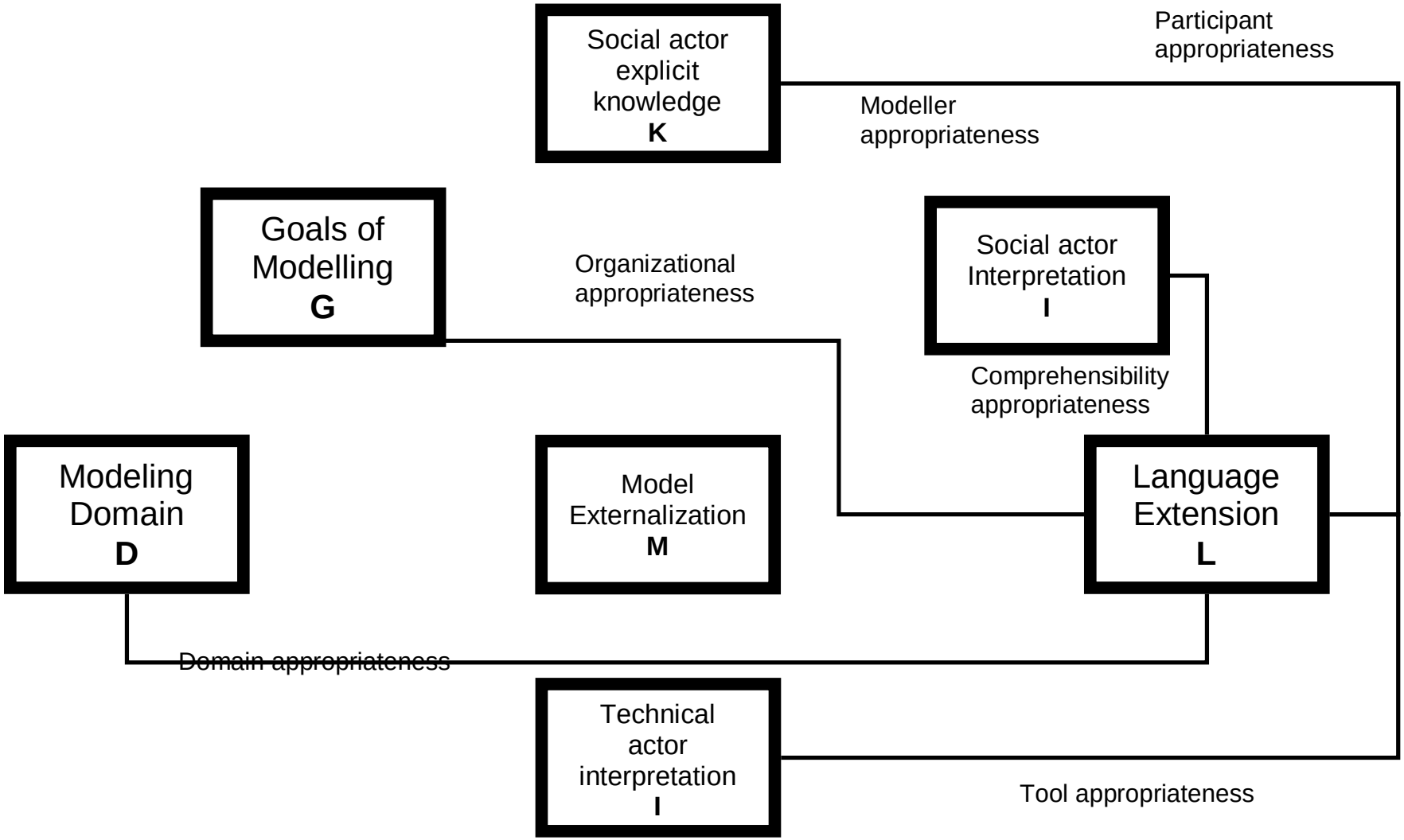
Usage of modeling and models



SEQUAL



Evaluation framework – language quality



Basis for evaluation

- Domain D: Primarily described through the laws and regulation for study financing. Although the domain seems to be fully externally given, in practice a many of the resulting rules to follow are based on internal deliberations within the Loan Fund
- Rule modeler (as a basis for K): The goals and rules were modeled in METIS and Blaze Advisor by professional rule designers and loan fund professionals in cooperation. For pre-defined changes the loan fund professionals could do this through a specific interface (RMA).
- Rule interpreter (vs. I): Views of METIS models and rules in Blaze were to be understood by those involved in the modeling. All loan fund personal were to be able to understand the rule documentation. RMA-rules being easier to understand were to be available for all. Through rule execution, texts including the reasoning of the decision made were produced, which are meant to be understandable by everyone.
- Language used for process and goal modeling was EEML in METIS. Formal rules followed the syntax of the proprietary rule language SRL (Structured Rule Language) supported in Blaze Advisor.
- Tool: Blaze Advisor, METIS.

Basis for evaluation

- **Model M:**
  - Data model (as a basis for the database-application, but also as basis for data definitions used in the case processing system and rule engine)
  - Process and goal model in EEML
  - Rule model (as a basis for the rule engine)
- **The rule model can be looked upon as four interrelated models:**
  - The laws and regulation as they are written in juridical terms. Here we look upon this as part of the domain
  - Goals in METIS
  - The rules as implemented in the rule engine (Blaze)
  - Some rules are made available through a web interface (RMA – Rule Maintenance Application)

Quality of rule language

- Domain appropriateness: It was possible to express all the execution rules in the PoC formally in SRL. Informal (deontic) rule and relationships could be expressed in EEML. No evaluation relative to standards (SBVR/PRR)
- Modeler appropriateness: The loan fund professionals were together with rule designers able to express the rules in SRL. Loan fund professionals were also able to use the RMA. EEML goal hierarchies could be developed through detailed discussions
- Participant appropriateness: EEML and SRL was only known by people from external companies, and it is found that specifically SRL represents a steep learning curve both for Loan Fund professionals as well as system developers internally.
- Comprehensibility appropriateness: Those closely involved in the process appeared to understand the rules, especially since navigation was supported the goal and process-models. Since the execution rules ended up as a mix of English keywords and Norwegian concepts used in the data model, they are somewhat hard to comprehend.
- Tool appropriateness: The tools where appropriate for enterprise modeling and rule execution, and other tests have been done supporting the scalability of the approach.
- Organizational appropriateness: A positive aspect here is that the language used (SRL) is according to an emerging standard (PRR). Expensive tools, with limited Norwegian expertise available

Quality of rule model

- **Physical quality: The rules are primarily available through the tool, which limits the availability. It is also possible to generate html-reports from the tools for wider availability, but it seems not appropriate for widespread dissemination. RMA includes standard authorization mechanisms, ensuring that only authorized personnel can change the rules.**
- **Empirical quality: Goal and process-model visualization was regarded as a useful way of getting an overview of the rule-base.**
- **Syntactical quality: The METIS model and SRL-rules implemented in the rule engine are syntactically correct**
- **Semantic quality: All the production rules are included, and rules as expressed in the underlying laws and regulations are included in the METIS model.**
- **Pragmatic quality: It was relatively easy to keep an overview of the implemented rules and how they were related to the laws and regulations . The METIS-model has made it easier to understand the underlying intention of the rules.**
- **Social quality: On some of the detailed rules there were discussions on the appropriate interpretation of these. This does not apply to the rules and regulations itself, but rather to how they should be follow up in practice in the Loan Fund.**
- **Deontic (Organizational) quality: The combined approach support all goals a to d (although not full support for b-simulation)**

Combination EEML and SRL vs. declarative modeling

- **Problem-orientation: The representation of business rules declaratively is independent of what they are used for and how they will be implemented. This is only partly the experience using the traditional production rule system in isolation (Blaze Advisor). The detailed expression of the rules in SRL is hampered by the needs of the implementation. A combination with a less formally defined rule language as we have illustrated with EEML is looked upon as beneficial instead of having to have different, not integrated representation.**
- **Maintenance: The benefits on this account is witnesses in the production-rule system, specifically with the added support of the RMA**
- **Knowledge enhancement: The explicit rule-representation, and the possibility to quickly test their effect has proved beneficially in this matter. The possibility to also relate the rules to more high-level goals in the rule hierarchy enables an even broader debate on these issues.**

Combination EEML and SRL vs. declarative modeling II

- Possible to distinguish between rules of necessity and deontic rules: EEML rules can include deontic operators.
- In many goal and rule modeling languages it is not possible to specify who the rules apply to: EEML-rules can be explicitly related to organizational actors.
- Flat rule-bases: Development of a rule hierarchy is supported, and it is also possible to link the rules to a hierarchical process model.
- Link to other models of the organization used to understand and develop the information systems, such as data and process models: Provided in EEML.
- The source of the rule can be indicated
- Support contradictory rules and goals: Possible in EEML. This is not shown in the case, but since the full range of deontic operators can be used between rules, it is possible to e.g. represent that the fulfillment of one rule forbids the fulfillment of another.

SEQUAL vs. Model characteristics

- Functional completeness : Part of semantic quality - completeness
- Generality: Related to social quality
- Efficiency: Tool appropriateness of modelling language used
- Perspicuity: Pragmatic quality
- Precision granularity:
- Minimality: Part of semantic quality - validity

Summary

- Promising results
- but another provider was selected for the overall reengineering project (100 Mill Euro)
- Aspects related to domain/tool appropriateness and organizational (deontic) quality not sufficient
- For modeling approaches of this kind to be taken up, one need to have interoperable tools and sufficient expertise available

Relation i* rule hierarchies with EEML goal modelling

- Possible to relate goals in a similar way in EEML
- EEML – Both goals of necessity and deontic goals/goal-relationships
- Mylopoulos: More focus on techniques for checking of goal-satisficing -> Possible extension of EEML
- EEML has also relationships from many to one goal (not only one-to-many)