Goal-oriented Analysis of Regulations

Robert Darimont[1], Michel Lemoine[2]

[1] CEDITI – B
Robert.Darimont@cediti.be
[2] ONERA, DPRS/SAE – F
Michel.Lemoine@onera.fr

Abstract This paper explains how goal-oriented requirements engineering can be transposed into regulation modelling. It motivates also why this way of modelling regulations is worthwhile for people responsible for preparing regulations. In addition, the paper recounts how the approach has been applied to model ICAO Security Regulation for Civil Aviation in the context of the SAFEE project.

1. GORE

Goal-Oriented Requirements Engineering (GORE) is an approach advocating the identification and analysis of goals as a prerequisite for writing of complete and consistent requirements documents.

One of the most prominent GORE methodologies is KAOS [1, 2]. In KAOS, the requirements engineer is prompted to build a requirements model before writing the requirements document by exploiting various information mines, such as interviews, documentation, observations, etc. The requirements model to build consists of four main integrated sub-models:

- The **goal model** captures the intentional view shared by all implied stakeholders. Goals are declarative properties on the system and its environment. Goals describe the problem to solve and are prescriptive. Goals are refined from high-level strategic intentions (i) into technical, low-level requirements on the system and (ii) into expectations on its environment. Conflicts between goals and obstacles preventing goals from being achieved are also recorded in the model.
- The **object model** captures the terminology needed to express the problem to solve (that is, the goals).
- The **agent model** provides an agent-centered view on the system-to-be.
- The **operation model** describes how agents have to cooperate to achieve the goals.

KAOS has been successfully used in many industrial or service contexts mainly to produce requirements documents, to define strategies and refine them into IT plans, to reengineer requirements on top of existing systems. It is supported by a tool: Objectiver [3].

2. GORE for Regulation Modelling

2.1 Regulations vs. requirements

According to the Oxford dictionary, laws are "*rules made by authority for the proper regulation of a community or society or for the correct conduct of life*". Merriam-Webster dictionary adds that "*regulations imply prescription by authority in order to control an organization or system*".

Laws and regulations are thus prescriptive assertions, the community or society implied has to follow. They play exactly the same role as requirements wrt information systems. In particular, regulations can suffer from the same kind of defects, mainly:

- **Ambiguities**: a lot of regulations contain articles that can be interpreted in several ways.
- **Inconsistencies**: they characterise conflicting articles in which an assertion and its converse should hold simultaneously. Inconsistencies can be internal to the regulation or imply other existing regulations.
- **Incompleteness**: the regulation forgets to cover some cases where it should be applied.
- **Unverifiable**: the way regulations are formalised do not enable authority to check how the community or society conforms to the regulations.

2.2 GORE again…

The similar nature between regulation rules and requirements advocates for a similar approach to synthesize them. This similarity can even be stressed on by renaming GORE (Goal-Oriented Requirements Engineering) as GORE (Goal-Oriented Regulation Engineering) with the following meaning:

- The **goal model** contains high-level goals explaining why the regulation is introduced (motivation clauses). It also explains how those high-level goals are refined into low-level, concrete regulation articles, each of which describing a specific regulation rule. Systematic refinements contribute to reduce the probability of incomplete regulations. In addition, obstacle analysis can be performed to anticipate and counter regulation deviations or bypasses.
- The **object model** is used to define the terminology used in the regulation goals, rules and articles. Ambiguities in regulations often arise out of a lack of such precise definitions.
- The **agent model** is used to specify who is concerned by the rules and who has to put it in force. Specific regulation agents can be introduced in the model to verify that the community or society conforms to the regulations.
- The **operation model** is used to check the impact of regulations on the community agent behaviours and to investigate how regulation agents can check how conformant the community or society behaves (contribution to verifiability).

3. Modelling Security Regulation for Civil Aviation

This section recounts a real case of regulation modelling in the context of the SAFEE project. Section 3.1 provides an overview on the SAFEE project. Section 3.2 explains how the user requirements have been built and in particular how the ICAO Security Regulation for Civil Aviation has been exploited.

3.1 The SAFEE project

The SAFEE project (Security of Aircraft in the Future European Environment) [4] puts together 30 companies from 12 European countries including Israel, all active in aeronautics (Airbus, British Aerospace, EADS, Thalès, SAGEM, Onera, …) to provide an answer to the 9/11 attack on Civil Aviation. The project is 4 years long for a total budget of 36 M€ partly funded by the EC.

The main goal of the project is to design a security system on board the aircraft to protect flights against acts of unlawful interference. The SAFEE system will be responsible for detecting threats during flights, assess them, report them to the crew and help them manage crisis situations either by advising some counter-measures or, in extreme situations, by taking control of the aircraft in order to land it safely on a dedicated airport in coordination with the ground.

3.2 Building Requirements

One of the ONERA&CEDITI contribution to the SAFEE project consisted in producing the requirements document for the TARMS subsystem. TARMS is the core of SAFEE responsible for assessing threats and for proposing responses to those threats.

TARMS clearly is an expected solution to security problems already identified or to come. Therefore, before investigating TARMS system requirements, it was decided to seriously investigate the problem to solve. As UML is more adequate for supporting solution-minded design activities than for supporting problem-minded requirements activities, the team decided to use the KAOS/Objectiver approach.

Two modelling iterations have thus been achieved to produce the TARMS System Requirements Document (SRD): the first iteration aimed at collecting the user requirements about SAFEE (see Section 3.2.1) and the second aimed at collecting system requirements for TARMS in order to address the user requirements identified during the first iteration (not detailed further in this paper).

3.2.1 TARMS User requirements

User requirements have been collected from three different sources:

- Interviews of a large set of stakeholders implied in the security of commercial flights: pilots, cabin crew, sky marshals, security managers, air traffic controllers, security authorities, airlines …

- Existing security regulations for air navigation from the ICAO (International Civil Aviation Organisation) [5] and the ECAC (European Civil Aviation Conference).
- Other security projects in progress (like the Eurocontrol ERRIDS project aiming at centralizing and dispatching security information about flights at the European level).

The model which has been derived from these information sources consists of three integrated parts:

- modelling the current situation by modelling the security goals to reach, how they are currently operated and who are the responsible agents for achieving or ensuring them. It is during this step that security regulations for air navigation have been investigated (see Section 3.2.2)
- modelling a set of threats which can occur on the ground or on board. Those threats aimed at putting the security goals identified in the first model into jeopardy. The threats have been modelled as KAOS obstacles representing: (i) anti-goals wished by offenders and (ii) vulnerabilities known on the system under attack, much in the sense of [6].
- Modelling how the SAFEE system is expected to contribute to security and improve it, that is, modelling the SAFEE requirements.

3.2.2 Regulation modelling

Let us now focus on how regulations have been considered during the first iteration.

The Security Manual for Safeguarding Civil Aviation against Acts of Unlawful Interference published by ICAO has been analysed with the Objectiver tool as follows:

- The Security Manual was imported into Objectiver as a source document.
- The imported document was browsed incrementally: each time a security goal appeared in the document, the text containing the occurrence was annotated with a hyperlink to a new or already existing goal in the KAOS/Objectiver goal model in order to provide traceability from the source documents to the goal model. For instance, when the regulations dealt with security measures for aircraft parked on the aprons, the original source document has been annotated with new security goals, a.o., (G1) "*No access to aircraft for unauthorized people*" and (G2) "*Doors sealed when aircraft left unattended*".
- Newly identified goals were added to the existing goal graph by finding the goals to which they contribute (see Figure 1). In the previous example, G1 has been attached as a subgoal of goal (G0) "*Aircraft secure while on the ground*" and G2 as a subgoal of G1.
- The terminology used in the document was also progressively acquired and defined. The text containing the main occurrences of concepts were annotated with an hyperlink to the object model. Each new concept was inserted in the object model by defining its relationships with other objects already in the model (associations, aggregations, specializations, …). Defining the terminology is important to flush out ambiguities. For instance, it was amazing to observe that, if

everybody agreed in the project team on the meaning of "*Cabin Crew*" and "*Cockpit Crew*", diverging opinions were raised about the meaning of the term "*Flight Crew*" as it was not clearly defined in the regulation statements. For some of us, the Flight Crew was the same as the Cockpit Crew while for others it included both the Cabin and Cockpit Crews. Flushing out such ambiguities is important to fix responsibilities about who has to do what in the system.

– Security threats have been identified by systematically challenging the security goals identified in the regulation. For instance, in the previous example, situations in which an access to the aircraft to unauthorized people could occur, was looked for. Situations like penetrating the aircraft by unsealing a door or because the door had not been sealed, or in extreme real or faked circumstances, etc. were investigated. Threats have been represented as KAOS/Objectiver obstacles obstructing goals in the goal model. All obstacles were then put together to build the threat model regarding the current security system. The obstacles were reviewed and classified into anti-goals and vulnerabilities much in the sense of [6].

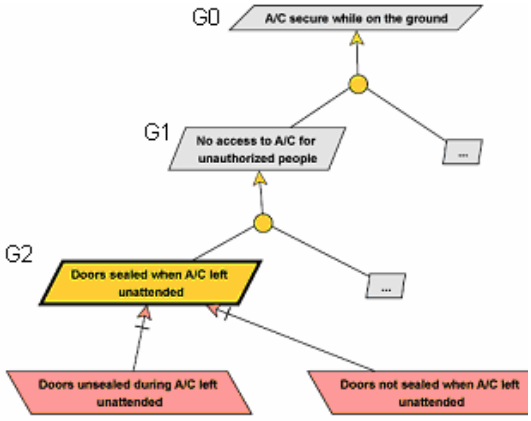

Fig. 1. A model fragment

4. Lessons learnt

Everybody knows existing incomplete, inconsistent or partially unverifiable laws or regulations. Corrective laws or new regulation releases are needed to address those defects. Those artefacts appear to be similar to patches produced by software editors to correct software bugs.

Our experience with the analysis of the ICAO regulations for Security of Civil Aviation has persuaded us that modelling regulations should reduce both the number and the seriousness of these defects, and is worth the value for the following reasons:

- Incompleteness cases are very hard to discover just by reading the regulation text. In KAOS/Objectiver, the refinement of goals into subgoals can be used to systematically identify and address those cases. Reviewing the goal model with domain experts triggers quick identification of missing cases.
- Goal models are perfectly understandable by domain experts as goals are expressed in terms of domain concepts the experts use to manipulate. In comparison, UML static diagrams reveal to be too IT-oriented for end-users. Use cases are perceived as too vague and sequence diagrams too concrete or instance-oriented while the regulation aims generally at covering all the possible cases.
- Obstacle analysis rooted in the goal model provides a systematic way to anticipate situations in which the regulation might be violated; one can decide subsequently how to address such situations by preventing occurrence of such situations or by introducing means to detect them and means to restore the broken rules.
- Defining the terminology precisely in the KAOS/Objectiver object model is also a great asset. A lot of ambiguities or unforeseen cases can arise from a lack of definition or from bad definitions. Once the terminology is fixed, it becomes possible to always use the same terms for the same concepts over the whole regulation document. Moreover, the object model provides knowledge on the domain which can be reused for writing other regulations in the same or closely related domain.
- The agent view is important to study who is responsible for what in the regulated system. For instance, it is important to know if a given rule is under the responsibility of pilots or cabin crews or to know who is responsible for sealing aircraft doors when the aircraft is left unattended. Considering agent reliability, availability and capability also can trigger identification of new obstacles.
- The operation model allows one to check the feasibility of the regulation by verifying that the assigned agents in the regulated system can always conform their behaviours to the regulation. We do not use the operation model to analyse the ICAO regulation as our prime objective was not to check the feasibility of a regulation in force but to reason on it to discover sources/causes of potential threats during flights. We believe however that the operation model is a corner piece for producing verifiable new regulations.
- Evolutions of the regulation should also be easier to implement as they will be based on the evolution of the underlying model. Impact analysis and consistency-preserving transformations are easier to be performed on the regulation model and then carried forward in the regulation text than directly performed on the regulation text.
- The construction of the regulation model requires skills both in the regulation domain and in model engineering. These skills are seldom concentrated in one head. Therefore a team putting together domain experts, lawyers and model engineers is needed to apply the methodology in an efficient way.

5. Conclusions

This paper outlines a methodology for modelling regulations inherited from requirements engineering. The paper shows an application of this methodology for analysing an existing regulation. The strong analogy between writing a requirements document for an IT system and writing a regulation document allows us to be confident that it could also be perfectly used for writing new regulations with the benefits of obtaining more complete, more robust, more verifiable and well-defined regulation documents.

6. References

[1] A. Dardenne, A. van Lamsweerde and S. Fickas, "*Goal-Directed Requirements Acquisition*", Science of Computer Programming, Vol. 20, 1993, 3-50
[2] A. van Lamsweerde , "*Goal-Oriented Requirements Engineering: A Guided Tour*", Invited Minitutorial, Proc. RE'01 - 5th Intl. Symp. Requirements Engineering, Toronto, August 2001, pp. 249-263
[3] Objectiver, http://www.objectiver.com
[4] http://www.safee.reading.ac.uk/, contract n° AIP3-CT-2003-503521
[5] ICAO, "*Security Manual for Safeguarding Civil Aviation against Acts of Unlawful Interference"*, Doc 8973/5 -- Restricted
[6] van Lamsweerde A., "*Elaborating Security Requirements by Construction of Intentional Anti-models*", in Proc. ICSE'04, 26th Int. Conf. On Software Engineering, Edinburgh, ACM-IEEE, May 2004 [2] *Security*, Amendment 11 of Annex 17, ICAO, November 2005