Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 14 (2016) 1957 – 1966

Transportation Research Procedia

www.elsevier.com/locate/procedia

6th Transport Research Arena April 18-21, 2016

Implementation of ERTMS: a methodology based on formal methods and simulation with respect to French national rules

Antoine Ferlin [a], Rahma Ben-Ayed [a], Pengfei Sun [a], Simon Collart-Dutilleul [a,*], Philippe Bon [a]

[a]*Ifsttar, BP 70317, 59666 Villeneuve d'Ascq Cedex, France*

Abstract

This paper presents the latest results of a three years project which aims at contributing to the validation and implementation of a European system for railway signaling called ERTMS "European Rail Traffic Management System". The management of railway trackside signaling in ERTMS is based on rules pertaining to each country and not on global rules. In order to perform a global safety assessment, a system analysis has to consider all these aspects. By the consequence, the main target of the project is to provide some methodological and software tools in order to perform consistency checking. The project named "Perfect" started in November 2012. The first step was to identify some critical scenarii leading to accidents or quasi accidents. The study leads us to focus on maintenance trains. In fact, even if there are not representing a big amount of the traffic, maintenance trains are involved in several critical situations. This intermediate result highlighted that there are no ERTMS running accidents because there were no ETCS2 implemented in France during the project. Consequently, the second step is to build a logical environment allowing replaying the identified scenarii under the ERTMS framework. This logical environment is obtained by a modeling task. Petri nets and B models may be used in order to provide some formal safety proofs. These two modeling tools are well accepted by some railway actors. Nevertheless ERTMS textual specification is not so formal. A semi-formal language like SysML seems to be well fitted to produce some intermediate models. A petri net model of the interlocking was proposed and a UML4secure Model was presented. The transformations of these two models into B machines have been published. Now we are able to prove some safety invariants on this global model of the system. In the last step of the project, based on the formal model some tests are generated, and then executed on the ERTMS simulation tool compliant with the official specifications. There are two kinds of motivations. The first one is to validate the assumption used by the safety proofs produced by the formal

* Corresponding author. Tel.: +33-3-20-41-83-20; fax: +33-3-20-41-83-98.
E-mail address: simon.collart-dutilleul@ifsttar.fr

Peer-review under responsibility of Road and Bridge Research Institute (IBDiM)
doi:10.1016/j.trpro.2016.05.163

models. The second one is to play the critical scenarii in the 3D environment of the ERTMS compliant simulation tool. The scenarii are obtained from two different procedures. Firstly, the accident database analysis allows defining some critical elements. Secondly, when the formal analysis failed at providing all proofs, the corresponding scenario has to be tested. The influences of infrastructure design and working context have been revealed in this last step. During this three years project, an instrumented methodology based on formal methods and a 3D simulation framework has been developed. This global framework does not avoid all difficulties, but it may allow producing a safety proof in shortest and more deterministic time.

Peer-review under responsibility of Road and Bridge Research Institute (IBDiM)

Keywords: B method; UML; RBAC Profile; Petri Nets; model transformations

1. Introduction

ERTMS is a normative and technological framework ensuing from European interoperability directives (ERA, 18/07/2008). In particular, official documents provide the specification of the on-board devices' behaviour and their communication with the trackside automatisms. In addition, trackside behaviours are specified by national rules. Implementing ERTMS in a country leads to dealing with these two-sided specification sources. For various reasons, both specification sources have their own logic and safety culture. The main proposition of the present approach is to use a formal method in order to model separately the trackside and on-board specifications. The main advantage of the methodology is to formalise a specification keeping some homogeneity in the definitions for a given domain. Assuming that two different modelling tasks are processed, it is interesting to select the appropriate formal method corresponding to a given domain. This selection considers the ability of the tool and the state of the art. Clearly, when a tool-based solution can be found in the state of the art, this tool is better considered. In fact, the basic principle is to obtain an appropriate formal model. Then, some Model Driven Engineering (MDE) (Bézivin, et al., 2006) technologies can be used in order to provide models of the specification which are expressed using a single modelling language.

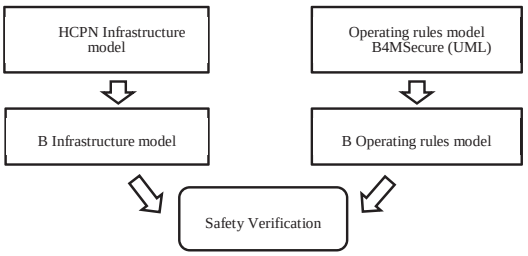

Fig. 1. Global framework of the approach.

Following this global description, the next section focuses on the infrastructure modelling part using Hierarchical Coloured Petri Nets (HPCN). The considered sub-system includes national interlocking and signalling specification. Then, another modelling task concerns ERTMS compliant operating rules. This work is performed using an access oriented UML profile (RBAC). For the two modelling tools above, automatic translation tools are provided in order to produce some abstract B machines. Using these B models some safety invariants may be proved using the embedded proof assistant. Finally, the presented results are discussed and some prospective works are identified.

2. Infrastructure modelling

A railway system consists of two main parts: the train and the ground infrastructure. In this part we broach the French national infrastructure. One of the most used modelling languages for railway is Hierarchical Coloured Petri Net.

2.1. Hierarchical Coloured Petri Net (HCPN)

We propose to use Petri Net pattern to model the behaviours of trains on their track. After, the obtained model will allow us to use a transformation to a *B* model in order to verify it. This section presents the automation of the generation of Petri Net models with respect to the safe interlocking principle using the track plan and the mapping route table as entry. The proposed approach deals with Hierarchical Coloured Petri Nets modelling and generalization, which is efficient about the definition of a large-scaled formal modelling framework. In addition, there are several benefits about the interlocking system modelling using Petri Nets:

- The mathematic semantics is precise and adapted for a rigorous specification of railway behaviours.
- Large-scaled models are efficiently treated using different coloured token.
- This is a graphic language easily understandable for railway component modelling.
- HPCN growing interest of industrial partners such as SNCF.

We propose to use a pattern of Petri Net to model all possible interlocking structures with only one customizable model. An interlocking system is composed of three essential elements: the track, the signal system and the train. ***A track*** is a set of topologic paths, including right sections, railway track point, traffic lights and automatic signal elements on the ground such as track electronic circuits. These elements should autonomously operate without any control from the signal control centre. Actually, they are treated as topologic path components. ***The signal system*** is a high level control system delegated to the monitoring in order to ensure the safety and efficiency of the trips. It contains operations rules, control processes for the interlocking system which are automatically computed and verified by an operator. For each system, the computer answers most operations such as route selection, the mode of route selection, the manual destruction of a route… and other unusual operations such as closure operations. ***Trains*** need a route with safe interlocking systems. They contain both the conditions about the itinerary run and the conditions about the railway net state at a given moment.

In our framework, trains are modelled in tokens which are systematically updated. The track and the signal systems are modelled by the Petri net topology, in order to express complex connexions and logical relations between different components. The Petri Net is separated in different parts according to three types: track, control and simulation. The first two basic types of an interlocking system are track and control. The last type concerns the data needed used for the simulation. It can refer to an initial situation or an action of an agent. Signal elements are common to the track net and to the control net. They act as indicators inside the control net and drive the train movement along the railway track. After defining the concepts needed, we can structure the Petri Net pattern using vertical layers.

2.2. A layered modelled infrastructure

The infrastructure modelling can be found in (Sun, Collart-Dutilleul, & Bon, A Model Pattern of Railway Interlocking System by Petri Nets, 2015). It consists in structuring the model in several vertical layers, because a marking railway system is a control system with several inputs/outputs. A model of such a system takes into account different aspects: components, scenarios and functions.

The ***component level*** is centred on communication and interaction between the different sub-systems. A component net highlights the sub-system and its interfaces. Each sub-system can be detailed in an additional level.

The ***scenario level*** is about the procedure modelling. It contains interactions between the train and the track equipment and the sequence of events which is required for the continuation of the operations.

The ***detailed function level*** is a low level model. Functions are about activity of process and not about the definition of scenario. Each function is modelled as a separated Petri net and can be used for different scenarios. Some functional modules can be used in different objects and are called detailed blocs.

Moreover, for simulation purposes and compatibility reasons, two additional levels should be added into the hierarchical structure. The “Elementary” level is used to replace the preliminary transition of the top level. The “Pre/Post” level concerns the relations between different components. They are used for preprocessing incoming messages and postprocessing outgoing messages.

An interlocking system is composed of several similar components: track components, railway-track-point components, and marking components. The framework of interlocking system layers is a topologic structure which can be considered as distributing the knowledge of the interlocking rules to objects of which the geographic place corresponds to physical elements. More details can be found at (P. Sun, 2014).

2.3. Generalization of the interlocking system layer

The stations that are equipped with the same type of RIS follow the same national rules. The only differences are the layouts of their fixed installations.

The expected structure should be both general and parameterized, which allows the specifications of stations to be derived from the same model with diverse configurations. That is to say, in this structure, the unmarked coloured Petri net is a set of RIS functional rules, while the initial tokens are the concrete performance of stations. In such a model framework, the configurations (tokens) represent all the scenario information, based on the formation of the RIS layout and the "condition table" (or control table). When modelling a new station, the only work is to change the initial tokens on the expected structure.

To have this general structure, the railroad layouts cannot be performed by the physical location of places and connection of transitions. However, this information is indeed important for train movements, so all this diverse information must be represented in the form of tokens, ensuring the PN structure itself remains universal. For a better understanding of the generalization concept, we use an incremental process and comparison examples to illustrate how to generalize the railroad structure.

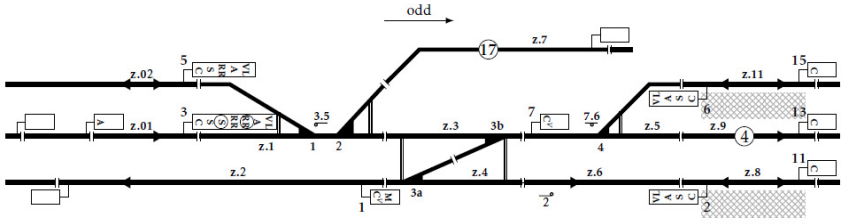

Fig. 2. Case study of a station layout (Rétiveau, 1987).

Basis Track segment: A token in a "train location" place indicates the train ID and its current location. Each time the transition occurs, the value of the train token will be refreshed according to the enabled binding elements. The "track connection" place is the constraints of train movement, which guide the train in moving forward.

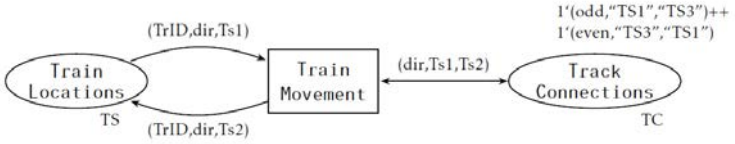

Fig. 3. Generalized representation of a track segment.

Adding points: When we introduce the points into the generalized structure, a place is first needed to "store" all the point information, including point IDs and the positions. Meanwhile, the points will have an impact on the train movements, so the configuration of track connection should be modified. The new colour set of *TC (Track Connections place* Fig. 3, Fig. 4 and Fig. 5) contains the point constraints. The train can move only when the point stored in the *Point List* place satisfies the point constraints.

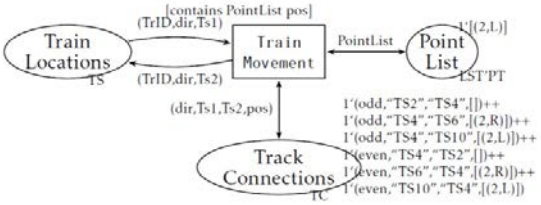

Fig. 4. Generalized representation including points.

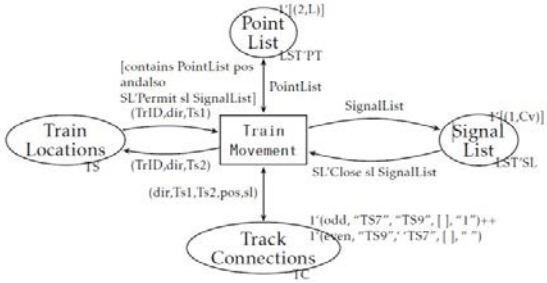

Fig. 5. Generalized representation including signal lights.

Adding signal lights: Similarly to a point, a signal light is also a movement constraint. So the introduction of signal lights comes with a new place and a modification to the colour set of *TC*. The function *SL'Permit* checks the corresponding signal indicator. If the indicator is *red*, then it returns *false* to prevent train movement. Otherwise, it returns *true* to permit the transit. The function *SL'Close* switches off the corresponding signal lights after firing the transition.

Even if we can express the components of rail tracks and their combinations, there are, in fact, more constraints (appliances) and rules. First, we should list all the scenario-related elements, and prepare their specification forms for the expected model. In Fig. 6, train, track, point and signal light are normal components that we have introduced in previous parts. In this table, we give them several attributes to distinguish between each token. The *Track Connection* stores all the connection information between different tracks, considering the constraints of points, signal lights and formation release triggers (the pedals). The *pedal* is the prerequisite condition for the *DA* mode interlocking route. The *Destruct Auto* is the automatic unlock mechanism and its devices. It contains the related unlock conditions and the unlock actions.

With all these variables and their notations, the next step is to describe the movement of a train. Although the expected model does not have visible routes, we can determine train movement by token values. If the value of the train position changes, that means this train actually moves. Generally, there are two types of routing routes, DA and TP, in the French national context. We also consider the route for shunting (OM), and the *staff responsible* mode (SR) for override operations. Without loss of generality, in this section, only DA mode will be discussed.

The conditions for enabling DA movement are:

1. There should be a pedal (passage detector for DA mode) in the current track;
2. Points of the route must be properly positioned;
3. Signal light (if any) in front of the train should be green;
4. The train's movement authority allows it to move onto the next track.

The actions which release the formation of the route along with train movement:

1. Release tracks of the route behind the train;
2. Release points of those tracks;
3. Switch off signal light (if any) after passing.

For analysis purposes, we introduce a security guard function, which constantly checks the occupation of the front track. The train's movement is safe, provided that the front track is clear. Otherwise, if the front track is occupied, there will be a "face to face" or "face to tail" collision.

From what has been mentioned above, the more formal definition of the enabling rules of the DA movement is shown in the . Fig. 7. With the help of *CPN ML* language, all the conditions above can be embedded into one transition and can combine into a single model to represent all the DA mode movements.

Using the generic pattern just defined above, a designing methodology may be applied for any kind of railway infrastructure. The Petri net modelling tool is familiar to the electro-mechanism domain (Antoni, 2009). It may be interesting to build a bridge towards other technical domains. The proposition developed in the next section concerns the transformation of coloured Petri nets into a B machine.

2.4. Transformation of Coloured Petri Nets into a B machine

(Sun, Model based system engineering for safety of railway critical systems, PhD Thesis, 24/07/2015) proposes to transform the CPN into a *B* machine to verify the correctness of the model. This method could help people to quickly shift from a valid design solution to a valid input of *B* development process in the design phase.

On the one hand, a fine behavioural specification has to be able to be assessed by an expert. On the other hand, the implementation result should respect some common industrial constraints. In the French railway context, the PNs and the *B* are two industry recognized tools. The PN models are understood and widely used by expert engineers, because of their powerful properties, especially the graphical "place-transition" notations. Therefore, many practical systems and valid solutions are specified with PNs and other high-level PNs. However, successful engineering stories convince people of the reliability of the *B* method, because the final implementation code generated from abstract *B* machine is considered safe and is proved to be safe. So in the French railway context, *B* proved model is accepted as a strong safety proof (Boulanger, 2013). Let us remark that the assessment of high--level design is a major safe element on its own. In this case, you may not need to execute the complete *B* refinement processes. The transformation is composed of several steps:

- Coloured tokens become naturals. The different colours are transformed into a non-empty finite set.
- A transition has an *id* and a state (valid or not valid). The guard of the transition is treated as an operation using predicates.
- Initial configuration of the Petri net is transformed into initialisations

This transformation is necessary to construct a full model that includes the infrastructure of a railway system and the operating rules of this system.

Element	Content	Notation
Train	train name	NmTr
	train direction	DirTr
	route name	NmRt
	route type (DA,TP,etc)	TpRt
	train position	PosTr
	movement authority	MA
Track	track name	NmTs
	Occupation status	Ocp
Track connections	current track	CurTs
	connection direction	DirTs
	post track	PostTs
	points (number varies [0,2]) (with name and its position)	PtTs
	signal light name [0,1]	NmSl
	indication of pedal	Ped
Point	point list (contains name and its position)	LstPt
Signal light	signal light list (contain name and its colour)	LstSl
Destruct Auto	exiting track (where DA take place)	TsDa
	effective direction of pedal	DirDa
	tracks to be destructed	TsLstDa
	signal light to release	SlDa
	points to release	PtDa

Fig. 6. Scenario-related elements in general structure.

Condition	equation
Route type	TpRt = DA
	Ped=TRUE
Route formation	PosTr = CurTs
	DirTr=DirTs
	PtTs ⊆ LstPt
Singal open	(NmSl,green) ⊆ LstSl
Movement authority	PostTs∈MA
DA activated	TsDa = PosTr
	DirDa = DirTr
To release	TsLstDa
	SlDa
	PtDa
Security check	Ocp of CurTs = Occupied
	Ocp of PostTs = Clear

Fig. 7. Conditions and equations of "DA" movement.

3. Operating Rules modelling

The second aspect of this study consists in modelling the operating rules in the ERTMS context, especially about the incidence of their interactions.

3.1. Case study

There are three kinds of operating rules:

- ERTMS rules

- National rules
- Track rules

In this section, the track rules are used to illustrate our approach. The ERTMS system is composed of the ETCS (European Train Control System) and the GSM-R (Global System for Mobile communications – Railways) for the data transmission between the on-board-ETCS and the on-ground-ETCS.

In order to ensure the safety and the efficiency of the system, operating rules define interaction between the real time systems and the operators such as the train drivers and the traffic controller.

Our study is based on the principle and operating rules of the ERTMS/ETCS level 2 applied to the LGV-Est line (RFF, 2012) and on specification described in (Alcatel, 2006), available on the Era website[1].

Our study which is based on two scenarios: a nominal scenario of Movement Authority (MA) and an exceptional scenario of crossing a stop ETCS order called Override End of Authority (EOA). Only the level 2 of the ETCS system is concerned by the present study.

3.1.1. Definitions

In level 2 of ETCS, the train received an MA in nominal mode. This is a supervised authorisation provided to the train for a specific distance. The MA is refreshed during the train progression.

An MA is the ETCS translation of an itinerary defined on the infrastructure from which all or a part is assigned to the train. This signal rule is based on the train position, on the occupation of the infrastructure by the other trains, on the operating safety rules and on the timetable of each train which depends on the operating rules of each track.

An MA is characterized by:

A section, which is the distance from a geographic point to the train. A section can be composed of several subsections. An MA can be applied on one or several sections. The last section is called the end section.

The end of Authority, which is the place where the train cannot go over it, where the speed indicated to the Driver Machine Interface (DMI) is equal to zero. It can correspond to an ETCS stop marker.

The targeted speed at EOA, which is the authorized speed at EOA. When the targeted speed is not zero, the EOA is called the Limit Of Authority (LOA). This speed can be limited with time.

The danger point, which is a limit point after the EOA which can be accessed by the head of the train without causing a dangerous situation.

The time delay, which can be attached to each section. It would be used for the revocation of the associated itinerary when the train did not take it. It can be attached to the section of the end of the MA, too. In this case, it is used for the revocation of the last section when it is occupied by the train.

3.1.2. Nominal scenario of Movement Authority

The considered scenario is described in (Ben Ayed, Collart-Dutilleul, Bon, Idani, & Ledru, 2014). It takes place with interaction between:

- The machine responsible for the safety management on board called *OnboardSafetyManagement* which demands an MA
- The machine responsible for the on-ground safety management called *TracksideSafetyManagement* which has received the demand and proposes an MA after performed verification
- The driver who can read the proposed MA through the DMI after the *OnboardSafetyManagement* validation. The driver must follow the instructions through the DMI.

3.1.3. Exceptional scenario (Override EOA)

Override EOA is a scenario triggered by the driver in specific degraded situations in the case of MA absence. When this mode is activated by the driver, the train is allowed to cross over an ETCS stop mark or an EOA after having received an EOA authorization. This authorization must be written and delivered as a safety message by the

[1] European Railway Agency: http://www.era.europa.eu.

traffic agent, in order to provide instructions. This message can be physically delivered or verbally delivered depending on the application modality of the safety technical rules about the communication. There are several kinds of written order from ETCS01 to ETCS07. An order contains at least the type of authorization, the authorization number, the timestamp of the deliverance, the post which delivers the authorization, the train impacted by the authorization, the place of application and a list of clear precise unambiguous actions to perform.

The scenario is:

OverrideEOA.1: the driver asks an Override EOA through the DMI. He is allowed to continue, to brake and to stop the train through the DMI.

OverrideEOA.2: the machine responsible for the On board Safety Management treats the Override EOA request and send it to the Trackside System.

OverrideEOA.3: the traffic Agent receives the request from the trackside System, creates a written order and authorizes it. He can modify and/or delete it.

OverrideEOA.4: The traffic Agent sends the written order to the On Board System.

OverrideEOA.5: The machine responsible for the on board safety management processes this authorization and displays it on the DMI

OverrideEOA.6: The driver follows the instructions displayed on the DMI linked to this authorization.

These orders and instructions such as MA and Override EOA are displayed on the DMI as text or in symbol format. The DMI is an intermediate between the *on board system* and the driver who must fulfil the instructions on the DMI. He can provide information to the system too. The study based on the two scenarios reveals the existence of interactions between the system and the different agents which act on the system. As a modelling practice, functionalities of the systems are separated from the safety policies. This access control policy allows us to clearly define the responsibility of each agent for each action.

3.2. UML Modelling using a RBAC profile

3.2.1. Motivations

For each possible action on a system, there is only one agent authorized to perform it. In addition, an agent can only perform actions which are authorized for him. Hence, in the modelled system, permissions are associated to the action and roles are attributed to each agent. These concepts can be refined from the access control principle based on roles (RBAC) which are applied in information systems.

For these reasons, we use the RBAC principle in our modelling activity using the B4MSecure Eclipse-based platform. This tool allows the modelling of systems in UML using the RBAC profile and the transformation of the UML model into a *B* machine, in order to formally validate the model.

3.2.2. RBAC Modelling

3.2.2.1. The functional model

The functional model allows the description of system entities used in the scenarios, the descriptions of the characteristics of each entity, the operations applied on each entity, and the relations between the entities. This modelling activity is ensured using class diagrams through classes, attributes, methods, relations and associations and associative classes. Indeed, entities are modelled by classes. Characteristics of entities correspond to the attributes and operations on entities correspond to the methods. The relations between the entities are modelled using the relations, the associations and the associative classes.

RBAC is a model of the ETCS system containing the on-board sub-systems (class TrainETCS) and the on--ground system (TrackSideSystem). A movement Authority (MA) and the written order are respectively modelled by two classes, MA and ETCSOrder with their own characteristic attributes. Each attribute is associated to a specific ETCS train and is displayed on the DMI (Display Interface) of the on-board system, after being processed and validated.

3.2.2.2. The security policy model

Access permissions are allowed to system users according to attributed roles. The two concepts permission and role are modelled according to the RBAC model, using the SecureUML (Lodderstedt T, 2002) profile which is an expansion of UML. Indeed, an action is only allowed for a role if this role can execute it. According to the profile RBAC, a role is translated into a class stereotyped named “Role” and a permission is transformed into an associative class stereotyped named “Permission” between the role and the entity of the system for which the permission is applied. Security policy models of the study enrich the functional model with four role models: the Driver, the Traffic Agent, the On board Safety Management, and the Trackside Safety Management. Fig. 8 is a security model which describes the permissions associated to the roles on a written order. A driver only can read a written order and following the appropriated instructions on the DMI, whereas the traffic agent can create, modify, authorize or delete a written order. For each entity of the functional model, we describe a security model about the permissions allowed to the roles which act on this entity.

3.3. Transformation into B Models

The B4MSecure Platform provides a set of rules to transform an RBAC model into a *B* machine. If necessary, operations can be enriched with pre-conditions and substitutions. The *B* machine can be improved by adding invariants. These improvements can be done inside the UML model using *B* annotations, in order to automatically provide them in the *B* machine.

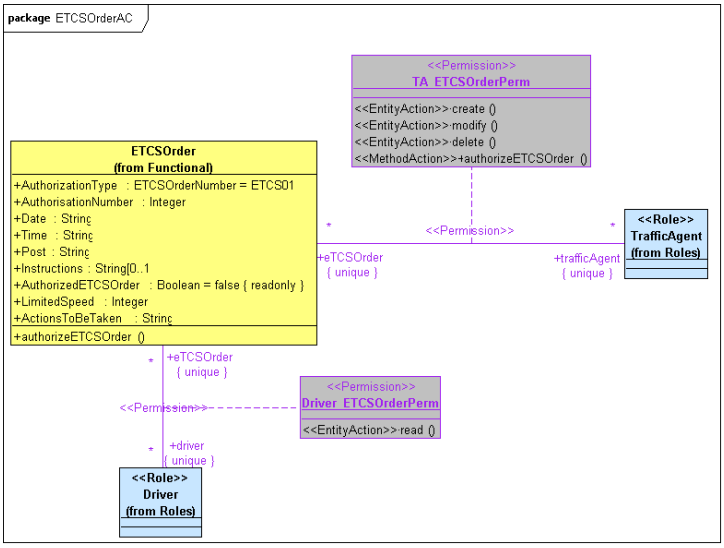

Fig. 8. Security policy model.

3.4. Validation of the B machine

The next step consists in verifying and validating the *B* machine, in order to be sure that models are compliant with the initial requirements. *ProB* and *Atelier B* are used to do that. *ProB* provides two approaches of validation: animation of model and model checking. We used the animator on the two scenarios of the LGV-Est line in order to detect fault consistency or imprecisions. The animation begins with initial values as first state and evolves according to the different given preconditions of the operations. This technique allows us to add several preconditions in the UML model as *B* annotation. The model has been transformed again and replayed until its completion. When the animation successfully ends, the second technique will be used. *Atelier B* is an industrial tool that generates proof obligations. An automatic proof system is provided by the tools and when this system fails, an interactive system

will be used to ensure the proof obligation is true. Currently, an open sources alternative called Rodin[2] is available. Currently, proof obligations about the functional model are proved. The security model is considered as a filter which does not add additional operations or additional behaviours. It allows the access limitation on operations depending on the role.

4. Conclusion

The paper has presented a tool-assisted global strategy for implementing ERTMS in a given country. The High Level Petri Nets have been used in order to formally model the interlocking and signalling logic of the considered country. Using a specific pattern is recommended in order to provide a safe and efficient process to the modelling task. Considering the operating rule modelling, the RBAC UML profile provides an interesting access oriented formalisation, whereas the B4MSecure framework may produce the corresponding abstract B machines. As the same kind of abstract B machine can be given from High Level Petri Nets transformation, it is possible to express the global specification using a unique formalism. The different model integrations may be applied using the tools and approach presented in (Behnia & Waeselynck, External Verification of B Development Porcess, 1998) (Behnia & Waeselynck, Test Criteria Definition for B Models, 1999). Consequently, the tool aspect is consistent, but, considering the knowledge engineering part, assisted ontology-based methodologies are to be developed. Considering the track modelling task, a more productive approach may be developed using the RailML standard. For operating rule modelling, the RBAC profile lacks context integration. Consequently, the use of an ORBAC profile is to be considered.

References

Alcatel, A. A. (2006). ETCS-Baseline 2, System Requirements Specification.

Antoni, M. (2009). Formal validation method and tools for french computerized railway interlocking systems. *International journal of railway*, (pp. 99–106).

Behnia, S., & Waeselynck, H. (1998). External Verification of B Development Porcess. *The 9th European Workshop on Dependable Computing*, (pp. 93–96). Gdañsk Poland.

Behnia, S., & Waeselynck, H. (1999). Test Criteria Definition for B Models. In LNCS (Ed.), *World Congress on Formal Methods, 1708*, pp. 509–529. Toulouse (France).

Ben Ayed, R., Collart-Dutilleul, S., Bon, P., Idani, A., & Ledru, Y. (2014). B Formal Validation of ERTMS/ETCS Railway Operatin Rules. In e. LNCS (Ed.), *4th International Conference ABZ, 8477*, pp. 124–129. Heidelberg.

Bézivin, J., Büttner, F., Gogolla, M., Jouault, M., Kurtev, F., & Lindow. (2006). Model Transformations? transformation models! In S. B. Heidelberg (Ed.), *Model driven engineering languages and systems*, (pp. 440–453). Berlin.

Boulanger, J.-L. (2013). Industrial Use of Formal Methods: Formal Verification. (J. W. Sons, Ed.)

ERA. (18/07/2008). Interoperability directives on the interoperability of the rail system within the Community. Legislation.

Lodderstedt T, B. D. (2002). SecureUML: A UML-Based Modeling Language for Model-Driven Security. In S. LNCS (Ed.), *5th International Conference on the Unifed Modeling Language (UML), 2460*, pp. 426–441. Heidelberg.

P. Sun, S. C.-D. (2014). Formal modeling methodlogy of French railway interlocking system via HCN. *COMPRAIL, International Conference on Railway Engineering Desing and Optimization.*

Rétiveau, R. (1987). La signalisation ferroviaire: Presse de l'école nationale des Ponts et Chaussées.

RFF. (2012). Principes et règles d'exploitation du système ETCS – Particularités en cas de superposition à un autre système de signalisation.

Sun, P. (24/07/2015). Model based system engineering for safety of railway critical systems, PhD Thesis. Villeneuve d'Ascq.

Sun, P., Collart-Dutilleul, S., & Bon, P. (2015). A Model Pattern of Railway Interlocking System by Petri Nets. *Models and Technologies for Intelligent Transportation Systems (MT-ITS).* Budapest.

[2] http://www.methode-b.com/en/tools/rodin/.