Product Line Engineering Applied to CBTC Systems Development

Alessio Ferrari[1], Giorgio O. Spagnolo[1],
Giacomo Martelli[2], and Simone Menabeni[2]

[1] ISTI-CNR, Via G. Moruzzi 1, Pisa, ITALY,
{lastname}@isti.cnr.it,
WWW home page: http://www.isti.cnr.it/
[2] DSI, Università degli Studi di Firenze, Via di S.Marta 3, Firenze, ITALY,
{lastname}@dsi.unifi.it,
WWW home page: http://www.dsi.unifi.it/

Abstract. The degree of automation in rail transport systems is constantly increasing. The gradual replacement of the control role of the human agent requires these systems to guarantee an enhanced level of safety and reliability. On the other hand, increased automation implies increased complexity. Dealing with such complexity requires to analyse the different system aspects at the proper level of abstraction.

The semi-formal and formal modelling methods can provide crucial support to meet the safety and reliability requirements and ensure that a proper degree of abstraction is maintained during the analysis. Furthermore, the product line engineering technology provides a suitable tool to integrate formal modelling with system modularity.

Communications-based Train Control (CBTC) systems are the new frontier of automated train control and operation. Currently developed CBTC platforms are actually very complex systems including several functionalities, and every installed system, developed by a different company, varies in extent, scope, number, and even names of the implemented functionalities. International standards have emerged, but they remain at a quite abstract level, mostly setting terminology.

This paper reports intermediate results in an effort aimed at defining a global model of CBTC, by mixing formal modelling and product line engineering. The effort has been based on an in-depth market analysis, not limiting to particular aspects but considering as far as possible the whole picture. The adopted methodology is discussed and a preliminary model is presented.

Introduction

Communications-based Train Control (CBTC) is the last technological frontier for signalling and train control in the metro market [19, 11]. CBTC systems offer flexible degrees of automation, from enforcing control over dangerous operations acted by the driver, to the complete replacement of the driver role with an automatic pilot and an automatic on-board monitoring system.

Depending on the specific installation, different degrees of automation might be required. Furthermore, companies shall be able to provide complete CBTC systems, but also subsets of systems. The aim is to satisfy the needs of green-field installations, and address the concerns of the operators who wish to renew only a part of an already installed system. In this sense, the product line engineering technology provides a natural tool to address the need for modularity required by a market of this type [6, 9].

Entering the CBTC market with a novel product requires such a product to be compliant with the existing standards. Two international standards provide general requirements for CBTC systems. The first is IEEE 1474.1-2004 [11], while the second is IEC 62290 [12, 13]. The standards differ in terminology and structure. Therefore, a product satisfying the former is not ensured to accomplish also the requirements of the latter.

A novel CBTC product shall also take into account the existing similar products and installations to be competitive w.r.t. the other vendors. The CBTC market is currently governed by six main vendors, namely Bombardier [26], Alstom [24], Thales [27], Invensys Rail Group [14], Ansaldo STS [2], and Siemens [21]. Each vendor provides its own solution, and different technologies and architectures are employed.

In this paper an experience is presented, where domain analysis has been used to derive a global CBTC model, from which specific product requirements for novel CBTC systems can be derived. The global model is built upon the integration of the guidelines of the standards, and is driven by the architectural choices of the different vendors. The model is represented in the form of a *feature diagram* [16, 3, 7], following the principles of the product-line engineering technology. From the global feature diagram, we derive the actual product requirements. To this end, we draw graphical formal models of the product architecture, together with scenario models in the form of simplified sequence diagrams. Architecture and scenario models are finally used to define and enrich the natural language requirements of the actual product. Examples are presented throughout the paper to explain the approach, and to show the results of the current implementation of the proposed methodology.

The paper is structured as follows. In Sect. 1, the CBTC operational principles are presented. In Sect. 2, an overview of the approach is given. In Sect. 3, an analysis of the standards and of the architectures of the CBTC vendors is presented. In Sect. 4, the global CBTC model is described. In Sect. 5, the architecture and scenario models are derived, together with the requirements for the actual product. In Sect. 6, related works are discussed. Sect. 7 draws final conlusions and remarks.

1 Communications-based Train Control Systems

CBTC systems [19, 11] are novel signalling and control platforms tailored for metro. These systems provide a continuous automatic train protection as well as improved performance, system availability and operational flexibility of the train.

The conventional signaling/control systems that do not use a CBTC approach are exclusively based on track circuits and on wayside signals. Track circuits are used to detect the presence of trains. Wayside signals are used to ensure safe routes and to provide information to the trains. Therefore, the position of the train is based on the accuracy of the track circuit, and the information provided to the train is limited to the one provided by the wayside signals. These systems are normally referred as *fixed block* systems, since the distance between trains is computed based on fixed-length sections (i.e., the length of a track circuit - see upper part of Figure 1).

CBTC overcomes these problems through a continuous wayside-to-train and train-to-wayside data communication. In this way, train position detection is provided by the onboard equipment with a high precision. Furthermore, much more control and status information can be provided to the train. Currently, most of CBTC systems implement this communication using radio transmission [17].

The fundamental characteristic of CBTC is to ensure a reduction of the distance between two trains running in the same direction (this distance is normally called *headway*). This is possible thanks to the *moving block* principle: the minimum distance between successive trains is no longer calculated based on fixed sections, as occurs in presence of track circuits, but according to the rear of the preceding train with the addition of a safety distance as a margin. This distance is the limit distance (MA, Movement Authority) that cannot be shortened by a running train (see lower part of Figure 1).

The control system is aware at any time about the exact train position and speed. This knowledge allows the onboard ATP (Automatic Train Protection) system to compute a dynamic braking curve to ensure safe separation of trains, which guarantees that the speed limit is not exceeded. The ATP system ensures that the MA is not shortened by the train, in addition to the continuous protection of the train in every aspect.

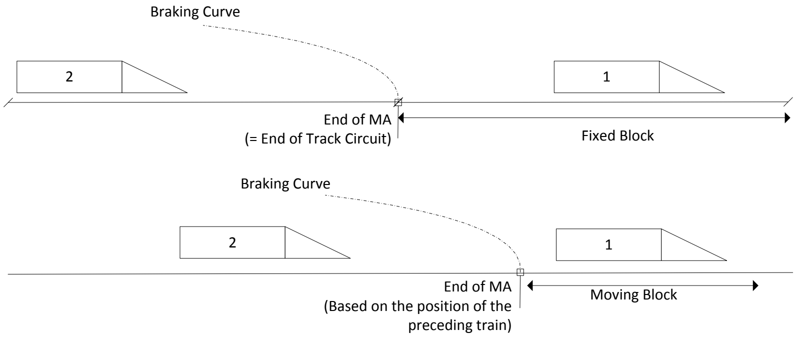


Fig. 1: Fixed block *vs* moving block

From the architectural point of view, CBTC systems are characterized by a division in two parts: onboard equipment and wayside equipment. The first is installed on the train and the latter is located at a station or along the line.

CBTC systems also allow automatic train control functions by implementing both the ATO (Automatic Train Operation) and the ATS (Automatic Train Supervision) systems. The ATO enlables the absence of the driver on board the train, ensuring the fully automatic management of the train in combination with ATP. The ATS offers functions related to the supervision and management of the train traffic, such adjustment of schedules, determination of speed restrictions within certain areas and train routing.

A CBTC system might include also an interlocking (noted in the following as IXL). The IXL monitors the status of the objects in the railway yard (e.g., switches, track circuits) and, when routing is required by the ATS, allows or denies the routing of trains in accordance to the railway safety and operational regulations.

2 Method Overview

In this work an approach has been defined to define a global model of CBTC and derive the product requirements for a novel CBTC system. The method starts from the available international requirements standard – IEEE 1474.1-2004 [11] and IEC 62290 [12, 13] – and from the public documents provided by the current CBTC vendors. Three main phases have been identified to move from these heterogeneous natural language description of the expected CBTC features to the actual CBTC product requirements.

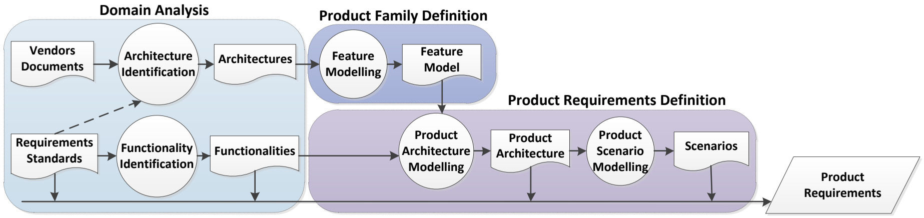

Domain Analysis
Vendors Documents
Architecture Identification
Architectures
Requirements Standards
Functionality Identification
Functionalities
Product Family Definition
Feature Modelling
Feature Model
Product Requirements Definition
Product Architecture Modelling
Product Architecture
Product Scenario Modelling
Scenarios
Product Requirements

Fig. 2: Overview of the product requirements definition process adopted

Figure 2 summarizes the approach followed. Activities are depicted as circles and artifacts are depicted as rectangles with a wave on the bottom side.

First, domain analysis is performed (Sect. 3). During this phase, the requirements standards are analysed together with the documents of the different vendors. The former are used to identify the functionalities expected from a standard-compliant CBTC system (Functionality Identification), while the latter are used to identify the system architectures adopted by the competitors (Architecture Identification). Requirements standards are also employed in the

Architecture Identification task to provide a common vocabulary to describe the architectures.

In the second phase, a product family for CBTC systems is defined (Sect. 4). The architectures identified in the previous phase are evaluated, and a feature model is derived to hierarchically capture all the different architectural options available in the market (Feature Modelling).

The last phase drives the definition of the actual product features (Sect. 5). From the feature model that represents the product family, a product instance is chosen. A detailed architecture is defined for such a product instance, taking into account the functionalities extracted from the standards (Product Architecture Modelling). Then, scenarios are derived to analyse the different behavioural aspects of the product (Product Scenario Modelling).

The final product requirements are the results of the adaptation of the standard CBTC requirements to the desired product. This adaptation is provided according to the (1) functionalities extracted from the standards, (2) the product architecture, and (3) the product scenarios.

3 Domain Analysis

3.1 Functionality Identification

In this phase, functionalities are identified for a generic CBTC system by evaluating the available international standards. Currently, the reference standards are IEEE 1474.1-2004 [11] and IEC 62290 [12, 13], which are briefly summarized below.

IEEE 1474.1-2004 The IEEE 1474.1-2004 has been defined by the Communications-based Train Control Working Group of IEEE (Institute of Electrical and Electronic Engineers) and approved in 2004. Such standard concerns the functional and performance requirements that a CBTC system shall implement. These requirements concern the functions of Automatic Train Protection (ATP), Automatic Train Operation (ATO) and Automatic Train Supervision (ATS), implemented by the wayside and onboard CBTC system. The ATO and ATS functions are considered optional by the standard. In addition to these requirements, the standard also establishes the headway criteria, system safety criteria and system availability criteria applicable to different transit applications, including the Automated People Movers (APM).

IEC 62290 The IEC 62290 is a standard defined by the IEC (International Electrotechnical Commission) gone into effect in 2007. This standard brings the fundamental concepts, the general requirements and a description of the functional requirements that the command and control systems in the field of urban guided transport, like the CBTC, shall possess. In reference to the fundamental concepts, the standard establishes four levels or Grades of Automation (GoA-1 to 4). The increasing GoA corresponds to increasing responsibility of the command and control system w.r.t. the operational staff. For example, a GoA-1 system simply enforces brakes when the driver violates the braking curve. A GoA-4 system does not have a driver, nor yet an onboard human supervisor.

Functionalies The standards have been evaluated to derive a complete set of CBTC functionalities. The approach adopted is as follows. First, the functionalities that the IEEE 1474.1-2004 standard specifies have been extracted. Such functionalities have been divided between ATP, ATO and ATS according to the anticipated classification provided by the same standard. Starting from this first group of functionalities, the activity continued with the analysis of the IEC 62290 standard, for identifying possible additional functionalities in comparison to those already extracted. Each functionality is traced to the paragraph of the corresponding standard from which it has been originally derived. Example functionalities, which are useful to understand the examples reported in the rest of the paper, are reported below together with the related subsystem and the reference to the standard documents.

Train Location Determination. (ATP onboard - IEEE 6.1.1) This functionality determines the position of the train;

Safe Train Separation. (ATP onboard - IEEE 6.1.2) This functionality uses the location information of the train to compute the braking curve and ensure safe separation of trains;

Movement Authority Determination. (ATP wayside - IEC 5.1.4.1) This functionality computes the MA message to be sent to the train based on the position of the other trains and on the railway status;

Route Interlocking Controller. (ATP wayside - IEEE 6.1.11) This functionality controls an external IXL and performs the route requests and locks. IXL systems are normally based on fixed block principles. This function is able to bypass the interlocking inputs concerning the position of the trains coming from the track circuits. In this way, the functionality is also able to ensure the increased performance guaranteed by the moving block principles;

Train Routing. (ATS - IEEE 6.3.4) This functionality allows setting the route for the train in accordance with the train service data, predefined routing rules and possible restrictions to the movement of the train;

Train Identification and Tracking. (ATS - IEEE 6.3.3) This functionality monitors the position and the identity of the trains.

3.2 Architecture Identification

In this phase, different possible architectures for a CBTC system are identified by evaluating the available information about the CBTC products on the market.

Several implementations of CBTC systems are offered by different vendors. In our work, we focus on the systems proposed by Bombardier, Alstom, Thales, Invensys Rail Group, Ansaldo STS, and Siemens. The CBTC of these producers are all made of subsystems that include a wayside equipment and a onboard equipment with a two-way wayside-train communication provided by a radio communication subsystem.

The major subsystems identified in the evaluated CBTC systems are ATP, ATS, ATO and IXL. There are also other additional subsystems, which include, e.g., the fire emergency system, the passenger information system, and the closed-circuit television. The system architectures are identified by analyzing the relationships among all these subsystems.

Bombardier The CBTC system proposed by Bombardier is called CITYFLO. The more advanced version of the system is the CITYFLO 650 [26], which introduces support for Driverless (DTO) and Unattended Train Operations (UTO). Like all CBTC systems, the architecture is composed by wayside equipment and onboard equipment. In particular, the wayside equipment of the system is distributed along the line and is divided into zones, called Regions. In this case, each Regions is responsible for safe movement of trains within its control limits and the safe delivery of the trains to the adjacent Region. For the determination of the position of the train the CITYFLO uses a model of the track based on entities called CITYFLO Segments. A Segment is a section of track identified by Region Number and Segment Number. The position of the train is identified as an offset in a Segment belonging to a Region.

The communication is provided by a Radio Frequency (RF) subsystem that uses a Spread Spectrum Code Division Multiple Access (CDMA) modulation technique at 2.4 Ghz. This RF communication uses either a leaky coxial cable or Line of Sight (LOS) antenna network along the wayside that transmits data to the trains via their onboard mobile antennas.

In case of failures, CITYFLO involves the use of a secondary backup system, based on track circuit, wayside signals and an IXL system. From a performance point of view, the CBTC system provided by Bombardier is capable of reaching a theoretical headway lower than 75 seconds, although the commercial headway achieved in the implementation of the Madrid metro [15] is around of 101/111 sec.

Alstom URBALIS [24] is the ultimate CBTC solution for Alstom that supports both UTO and DTO. Both wayside equipment and onboard equipment are integrated networks, based respectively on SDH (Synchronous Digital Hierarchy) Multi Service network and on the Ethernet network. Instead, communication between the wayside equipment and the onboard equipment takes place by means of a radio communication that is based on IEEE 802.11 g/a with OFDM carriers at 2.4 GHz or 5.8 GHz. The propagation media that are supported by URBALIS are Free Propagation, Leaky Feeder or Wave Guide. The onboard equipment of the ATP subsystem is responsible for determining the position of the train, using the information read from the balises (EUROBALISE) arranged along the tracks. The kinematic measures of speed and acceleration are always carried on board the train by means of odometric sensors installed on the axles of the train. For non-equipped trains and in case of failures, URBALIS involves the use of secondary detection devices, like track circuit and LED signals. From a performance point of view, the CBTC system provided by Alstom is capable of reaching a theoretical headway lower than 85 sec, even if the commercial headways obtained in the actual implementations are around 90 sec [10].

Thales The CBTC system of Thales is called Seltrac [27], and S40 is the latest version. This version provides support to UTO and DTO and is optimized to achieve a headway of 60 sec. Seltrac provides an architecture similar to that provided by Bombardier, where the wayside equipment of the system is divided into zones [25]. Each zone is controlled by a Zone Controller which is responsible for controlling and monitoring the train along the track. Based on this architecture, the system can be implemented either using inductive loops arranged into line segments (typically 1.5 km long), or using a wayside-train radio communication subsystem based on IEEE 802.11 [1].

In the first implementation, the position and speed of the train are provided through loops arranged along the line for ground reference calculations. Tachogenerators are employed for the calculation of speed, direction and distance in collaboration with the accelerometers. In the second implementation, the train position is determined with trackside transponder tags. From a performance point of view, the Seltrac S40 is capable of reaching theoretical headway less than 60 sec, however the commercial headway obtained in the implementation of Dubai metro is around 90 sec.

Invensys Rail Group SIRIUS [14] is the CBTC system proposed by Invensys Rail Group. This system uses a 2 out of 3 voting logic to identify failures and take appropriate actions. The continuous and two-way communication between trackside equipment and onboard equipment is guaranteed through a radio subsystem that uses spectrum modulation techniques and antennae or leaky feeder. The train position is determined using Absolute Position Reference (APR) passive balises arranged along the track and activated when the train passes over them. Gearbox or devices driven by the wheels of the train, in collaboration with Doppler radar unit, are responsible for measuring the speed and distance. From a performance point of view, the system proposed by Invensys Rail Group is capable of reaching a theoretical headway lower than 80 sec.

Ansaldo STS The CBTC system provided by Ansaldo STS has no proper noun [2]. In this system, the bidirectional train-to-trackside communication is provided by a radio communication subsystem based on IEEE 802.11. The position and speed of the train is determined with the use of balises arranged along the tracks and with the use of odometric sensors on board the train. In particular, the position of unequipped trains is obtained by the interlocking depending on whether the track circuits are occupied or not. This CBTC system is capable of reaching a theoretical headway lower than 60 sec [18], however, the commercial headway obtained in the implementation of the Copenhagen metro is around 90 sec.

Siemens The CBTC system developed by Siemens is called TRAINGUARD MT [21]. Siemens provides three control levels depending on whether the trains along a CBTC equipped line are equipped or not. In this way it is possible to let trains with different degrees of automation cohabit on the same line. The communication between the trackside equipment and onboard equipment is based on the Airlink subsystem [23], which provides a two-way radio transmission on the 2.4 GHz ISM band. The determination of the position and speed of the train is computed by the onboard equipment through a radar and an odometer pulse generator. Additionally, Trainguard MT involves the use of an ACM axle counting system as reliable track vacancy detection system. The system proposed by Siemens is capable of reaching a rather low commercial headway [22], such as the one obtained in the implementation of the Barcellona-line 9 metro which is around 80 sec.

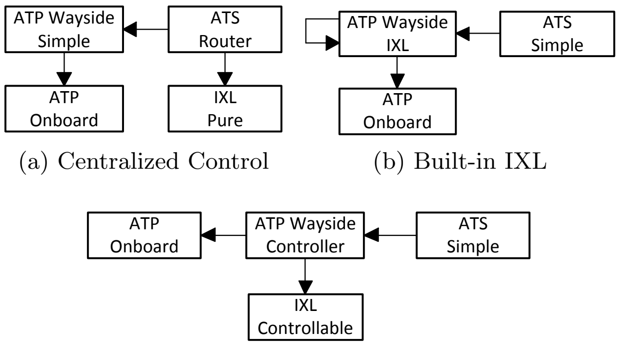
Architecture Identification The possible CBTC architectures have been identified by analyzing the relationship between the different subsystems. As examples, we focus on the relationships among ATP, ATS and IXL. The most relevant configurations identified for these systems are summarized below.

Centralized Control. (Figure 3a) In this configuration, the ATS controls both the ATP and the IXL. The ATS is called `ATS Router` since it has a direct interface with the IXL to perform routing. The wayside ATP is called `Wayside ATP Simple` since it has no direct interface with the IXL, and the communication among these two subsystems is managed through the ATS. Furthermore, the wayside ATP communicates with the onboard ATP, as in all the other configurations.

Built-in IXL. (Figure 3b) In this configuration there is no external IXL, since the ATP encapsulates also the functions of the IXL (`ATP Wayside IXL`). We call the ATS of this configuration `ATS Simple` since it has no direct interface with an IXL.

Controllable IXL. (Figure 3c) The wayside ATP has a control interface (`ATP Wayside Controller`) with an external IXL, and acts as intermediary between the `ATS Simple` and the IXL. We call the IXL of this configuration `IXL Controllable` since, unlike the `IXL Pure` of the first configuration, allows the `ATP Wayside Controller` to bypass some of its controls to achieve improved performances. It is worth noting that this solution would not be possible with an ATS controlling the IXL. Indeed, the ATS is normally not meant as a safety-related system, while the ATP and the IXL are safety-critical platforms.

The first and second configurations are both used by Bombardier. The second architecture is described in the Bombardier documentation as CITYFLO 650 with built-in IXL. Though the first architecture is not explicitely described,



(a) Centralized Control (b) Built-in IXL

(c) Controllable IXL

Fig. 3: Architectures extracted

the Bombardier documentation states that, when available, the IXL works as a backup system in case of ATP failure. Therefore, we can argue that the IXL control resides in the ATS and not in the ATP.

The third architecture has been derived evaluating the Alstom system. The IXL employed by Alstom is provided by the same supplier of the Bombardier IXL, but Alstom does not use this IXL as a backup system. Therefore, we can argue that the ATP is in charge of controlling the IXL, as in the third architecture.

4 Product Family Definition

The development of industrial software systems may often profit from the adoption of a development process based on the so-called *product families* or *product line* approach [9, 6]. This development cycle aims at lowering the development costs by sharing an overall reference architecture for each product. Each product can employ a subset of the characteristics of the reference architecture in order to, e.g., serve different client or jurisdictions.

The production process in product lines is hence organized with the purpose of maximizing the commonalities of the product line and minimizing the cost of variations [20]. A description of a product family (PF) is usually composed of two parts. The first part, called *constant*, describes aspects common to all products of the family. The second part, called *variable*, represents those aspects, called variabilities, that will be used to differentiate a product from another. Variability modelling defines which features or components of a system are optional, alternative, or mandatory.

The product family engineering paradigm is composed of two processes: *domain engineering* and *application engineering*. *Domain engineering* is the process in which the commonality and the variability of the product family are identified and modelled. *Application engineering* is the process in which the applications of the product family are built by reusing domain artefact and exploiting the product family variability [20].

4.1 Feature Modelling

The modelling of variability has been extensively studied in the literature, with particular focus on *feature modelling* [16, 3, 7]. Feature modelling is an important technique for modelling the product family during the domain engineering.

The product family is represented in the form of a *feature model.* A feature model is as a hierarchical set of features, and relationships among features.

Relationships between a parent feature and its child features (or subfeatures) are categorized as: *AND* - all subfeatures must be selected; *alternative* - only one subfeature can be selected; *OR* - one or more can be selected; *mandatory* - features that required; *optional* - features that are optional; *a require b*, if a and b are present; *a exclude b*, if a is present and b is not present and vice-versa. A feature diagram is a graphical representation of a *feature model* [16]. It is a tree where primitive features are leaves and compound features are interior nodes. Common graphical notations are depicted in Figure 4.

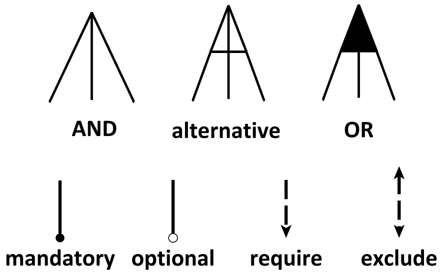


Fig. 4: Feature diagram notations

4.2 A Global Feature Diagram for CBTC

At this stage of the research, we have been able to define a global feature model for CBTC at the GoA-1 level, according to the IEC 62290 terminology [12]. In other terms, our model assumes the presence of a driver on board. The model has been defined by integrating the different architectural choices identified during the architecture identification task (Sect. 3.2).

A simplified excerpt of the global feature diagram associated to our model is given in Figure 5. The diagram includes the architectural components (which in our diagram becomes *features*) already identified in Sect. 3.2.

The *require* constraint requires a product to include `IXL Pure` and `ATS Router` whenever the product includes `ATP Simple`. Indeed, the control interface with the IXL has to be implemented by the ATS if the ATP does not include it, as in the case of `ATP Simple`. Also `IXL Controllable` is required whenever `ATP Controller` is used. In this case, a proper controllable interface of the IXL is required to let the ATP system control its functionalities.

The `ATP onboard` is required by any product of this family. On the other hand, the features `IXL Pure` and `IXL Controllable` cannot cohabit in any product of this family. The same observation holds for `ATS Router` and `ATP Simple`. Indeed, only one type of IXL and one type of ATS is allowed in a product.

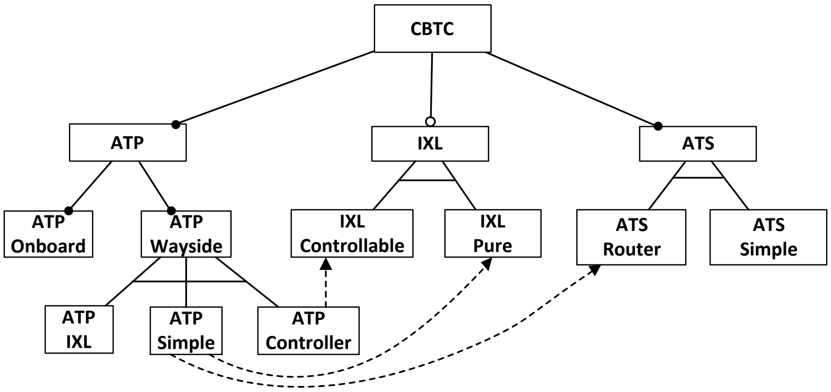


Fig. 5: Simplified excerpt of the CBTC global feature diagram

It is worth noting that the feature diagram allows new configurations that were not identified in the domain analysis phase performed. These configurations represent new possible products. For example, an `ATP IXL` can - optionally - cohabit with an `IXL` of any type. In this case, the additional `IXL` works as a backup system.

5 Product Features Definition

The provided feature model represents a global model for CBTC at the GoA-1 level. From this global model we choose a product instance, which in our example case corresponds to the Controllable IXL architecture of Figure 3c. Then, we model the *detailed architecture* of the product according to the functionalities extracted from the standards in the domain analysis phase. The architecture represents a static view of our product in the form of a block diagram. In order to assess the architecture, we provide realistic scenarios using architecture-level sequence diagrams. This phase can be regarded as the application engineering process of the product family engineering paradigm. Architecture and scenarios are employed to derive requirements for the actual product.

5.1 Product Architecture Modelling

The graphical formalism adopted to model the product architecture is a block diagram with a limited number of operators. We have designed this simple language in agreement with our industrial partner, and according to our previous experiences in the railway industry. Companies tend to be skeptical about the benefit given by the adoption of complex and rigid languages during the early stages of the development. Instead, they are more keen to accept a lightweight formalism that allows them to represent architectures intuitively and with a limited effort.

The diagrams are composed of blocks and arrows. Blocks can be of two types: *system blocks*, which represent individual hardware/software systems, or *functionality blocks*, which represent hardware/software functionalities inside a system. Two types of arrows are also provided: *usage arrows*, allowed between any block, and *message arrows*, allowed solely between functionalities belonging to different systems. If a usage arrow is directed from a block to another, this implies that the former uses a service of the latter. If a message arrow is directed from a functionality to another, this implies that the former sends a message – the label of the arrow – to the latter.

We describe the usage of this formalism with an example. Given the global CBTC model, we first select the features that we wish to implement in our final product. For example, Figure 6 highlights in pink (grey if printed in B/W) the features that are selected for a CBTC system that uses a controllable interlocking (see Figure 3c).

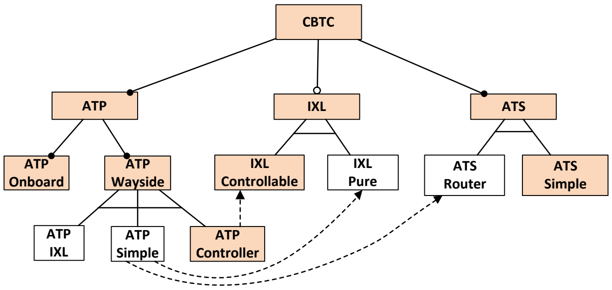


Fig. 6: Selection of features for our example product

An excerpt of the detailed architecture for the selected product is depicted in Figure 7. It is worth noting that the functionality blocks used are part of the functionalities identified during the domain analysis phase.

The `Train Location Determination` functionality belonging to the onboard ATP sends the train location information to the ATP wayside system. The `Movement Authority (MA) Determination` functionality forward this information to the ATS for train supervision, and uses this information to compute the MA. The `Train Routing` functionality of the ATS requires the routes to the wayside ATP, which controls the routing by means of the `Route Interlocking Controller` functionality connected to the IXL. We recall that the `Route Interlocking Controller` functionality is used to modify the interlocking inputs concerning the location of the trains – normally based on fixed block principles – to achieve the increased performance of the moving block paradigm.

5.2 Product Scenario Modelling

The architecture provided during the previous activity has been defined according to the functionalities extracted from the standards. Nevertheless, some

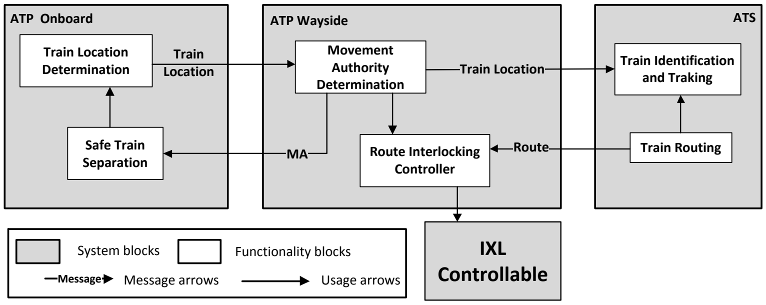


Fig. 7: Architecture example for a CBTC system

connections among functionalities, or some message exchange, might be missing from the model, since the architecture has not been evaluated against actual scenarios. In order to refine the architecture, and provide coherent requirements for the product, graphical scenarios are defined.

The graphical formalism adopted to model the scenarios at the architectural level is a simplified version of the UML sequence diagrams. Lifelines are associated to systems, while blocks along the lifelines are associated to the functionalities of the system. The arrows among different blocks are indicating message communication or service requests. In case of message communication, the arrow is dashed. In case of service requests the arrow is solid.

Figure 8 reports a scenario for a train that moves from a station to another according to a route defined by the ATS.

In the operational center, the ATS sends the `Route` information to the wayside ATP. The wayside ATP requests the IXL to move the switches in the proper position, and to lock the resources (the `setRoute` service request). Once the route has been locked by the IXL, the wayside ATP sends the `Movement Authority` to the onboard ATP. The onboard ATP allows the train departure, so the driver can start the train movement. While moving, the onboard system updates its position and sends the `Train Location` information to the wayside ATP. This system uses such information to compute new MAs for the current and preceding trains. Furthermore, the wayside ATP forwards the `Train Location` information to the ATS for identification and tracking.

It is worth noting that in this representation, we have added the `setroute` service request, which was not defined in the block diagram. This explicit request is an example of refinement enabled by the usage of scenarios: the relationship among the `Route Interlocking Controller` functionality and the `IXL Controllable` system has been clarified by means of the sequence diagram.

5.3 Requirements Definition

The information provided throughout the process are used to define the requirements of the final product. In particular, the requirements of one of the standard

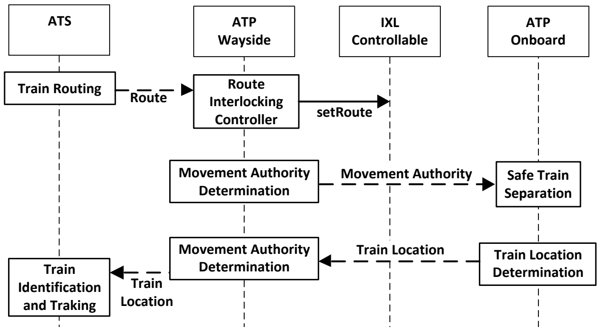

ATS
ATP Wayside
IXL Controllable
ATP Onboard
Train Routing
Route
Route Interlocking Controller
setRoute
Movement Authority Determination
Movement Authority
Safe Train Separation
Movement Authority Determination
Train Location
Train Location Determination
Train Identification and Traking
Train Location

Fig. 8: Example sequence diagram: a train moves from one station to another

are used as a reference for the definition of the actual product requirements. In our case, we take the IEEE 1474.1-2004 standard as a reference.

The requirements are tailored according to the functionalities extracted from the standards, and evaluating the product architecture and the scenarios. For example, consider the following requirement referred to the ATP system:

6.1.11 – **Route Interlocking.** A CBTC system shall provide route interlocking functions equivalent to conventional interlocking practice to prevent train collisions and derailments. [...]
Where an auxiliary wayside system is specified by the authority having jurisdiction, interlocking functions *may* be provided by separate interlocking equipment [...].

In our example product, the interlocking is an auxiliary wayside system external from the ATP. Therefore the Derived (D) requirement for our product is:

6.1.11(D) – **Route Interlocking.** Interlocking functions *shall* be provided by separate interlocking equipment [...].

Additional requirements on the actual behaviour can be derived from the architecture and the example scenario, as in the following:

6.1.11$(D-1)$ – **Route Interlocking Controller.** When a route is requested from the ATS, The ATP system shall require route setting (`setRoute`) to the interlocking to lock the interlocking resources. [...]

The behaviour expected from this requirement is clarified by the scenario, which is also attached to the requirement in the final specification.

Consider now a vendor that wish to accomplish also the IEC 62290 standard with his product. The product is already defined according to IEEE 1474.1-2004 following the presented approach. In this case, we argue that the compliance with the IEC 62290 standard can be demostrated by reasoning at functional level. Indeed, the functions identified in the domain analysis phase integrate the content of both standards, and traceability with the original functional requirements of IEC 62290 is therefore made easier.

6 Related Works

There is a large literature concerning the development methods of train control systems, including CBTC. Below some works are listed that represent the most relevant examples related to our work.

The MODCONTROL [4] project aimed to define a set of generic requirements for a new generation of Train Control and Monitoring Systems (TCMS). In particular, it has in common with our work the collection of requirements from different sources such as specifications of existing systems, standards or draft specifications from other EU projects. The second part of the MODCONTROL project differs from our work since it is more focused in finding linguistic defects in the requirements.

The work performed by LS Industrial Systems [29] concerns the software development of a CBTC system by means of a process based on model-driven development principles. In particular, the UML language is used to model the CBTC software, and source code for the model is derived through the IBM Rhapsody tool. Unlike our case, where requirements are represented in textual form and derived from the analysis of existing systems and standards, the authors use a UML notation (Use Cases) to represent the customer requirements, and do not give details concerning the domain analysis phase.

Wang and Liu [28] present an approach for developing a CBTC system based on a 3-levels hierarchical modelling of the system. The three levels are the functional model, the behavioural model of the train, and the model of all control actions. To illustrate this approach, authors use SCADE applied to a case study of a specific CBTC subsystem.

Essamé and Dollé [8] present the application of the B method in the METEOR project led by Siemens Transportation Systems. According to the authors, the use of the B method to realize the vital software system for the automatic control of the train, called METEOR, is economical if considered in relation with the entire development process of the CBTC system, which includes the validation of the specification and the product certification.

Yuan *et al.* [30] illustrate a modeling approach and verification of the System Requirement Specifications (SRS) of a train control system based on the Specification and Description Language (SDL). The application of this approach has allowed the authors to identify possible ambiguities and incompatible descriptions of SRS, useful for making changes on the SRS.

The first two works mainly concern the usage of semi-formal methods or structured approaches, while the other three works are focused on formal methods. Our work does not strictly employ formal techniques, and can be therefore attached to the first group. Besides other process-related differences, the current paper mainly differs from all the other works for the emphasis given to the product line aspects of the CBTC development. The main novelty is indeed the domain analysis performed, and the process adopted to define requirements for a novel CBTC system. We argue that this approach enables the development of a modular, competitive, and standards-compliant CBTC system.

7 Conclusion

In this paper, preliminary results are presented concerning the definition of a global model for Communications-based Train Control (CBTC) systems. The model is derived from existing CBTC implementations and from the guidelines of international standards, and is represented in the form of a *feature model*. A methodology has been also outlined to derive product requirements from the global model.

The current model is limited to the functionalities of a CBTC system that requires a driver. However, the most relevant safety-critical components are already detailed in our representation.

The approach has been considered higly valuable by our industrial partner, who acted as external supervisors for the presented work. The most promising commercial aspect is the value given to (1) the consideration of the competitor's choices, and (2) to the adherence to the standards. Indeed, though a migration strategy from a standard to the other is not fully defined yet, we expect the transition to be simplified by the consideration of all the available standards during the functionality identification phase.

Another aspect that has been highly appreciated by our partner is the choice of the modelling languages. The feature model by itself provides an abstract view of the product family that is easily understood by the stakeholders [5]. On the other hand, the block digram notation and the sequence diagrams defined allows focusing on the essential concepts, even employing a limited number of operators. Other languages, such as SysML or Simulink/Stateflow, have been considered too complex to be useful in this analysis phase.

Given the promising results of the current approach, we are presently working on an enhanced version of the model that includes also capabilities for driverless and unattended operation. Integration of the approach with natural language requirements analysis methods is also foreseen.

References

1. Alcatel. Integration of Wireless Network Technology with Signaling in the Rail Transit Industry, 2003.
2. Ansaldo STS. CBTC Brochure. http://goo.gl/3Kmb0, 2011.
3. D. S. Batory. Feature models, grammars, and propositional formulas. In *Proc. of SPLC*, pages 7–20, 2005.
4. A. Bucchiarone, A. Fantechi, S. Gnesi, and G. Trentanni. An experience in using a tool for evaluating a large set of natural language requirements. In *Proc. of SAC*, pages 281–286, 2010.
5. G. Chastek, P. Donohoe, K. C. Kang, and S. Thiel. Product Line Analysis: A Practical Introduction. Technical Report CMU/SEI-2001-TR-001, Software Engineering Institute, Carnegie Mellon University, 2001.
6. P. C. Clements and L. Northrop. *Software product lines: practices and patterns*. Addison-Wesley Longman, Inc., Boston, MA, USA, 2001.
7. K. Czarnecki and U. Eisenecker. *Generative programming: methods, tools, and applications*. ACM Press/Addison-Wesley, New York, NY, USA, 2000.

8. D. Essamé and D. Dolé. B in Large-Scale Projects: The Canarsie Line CBTC Experience. In *Computer Science*, volume 4355/2006, pages 252–254. 2006.
9. A. Fantechi and S. Gnesi. Formal modeling for product families engineering. In *Proc. of SPLC*, pages 193–202, 2008.
10. A. Guerra. Il nuovo sistema CBTC per metropolitane - L'automazione della linea 1 della Metropolitana di Milano. http://goo.gl/VGnzf, 2009.
11. Institute of Electrical and Electronics Engineers. IEEE Standard for Communications Based Train Control (CBTC) Performance and Functional Requirements. *IEEE Std 1474.1-2004 (Revision of IEEE Std 1474.1-1999)*, 2004.
12. International Electrotechnical Commission. IEC 62290-1: Railway applications: Urban guided transport management and command/control systems. Part 1: System principles and fundamental concepts. 2007.
13. International Electrotechnical Commission. IEC 62290-2: Railway applications: Urban guided transport management and command/control systems. Part 2: Functional requirements specification. 2011.
14. Invensys Rail. SIRIUS Brochure. http://goo.gl/YFUiL, 2009.
15. A. Jeronimo. Solving the Capacity Challenge - CBTC for Metro de Madrid. http://goo.gl/7Sy1C, 2010.
16. K. C. Kang, S. G. Cohen, J. A. Hess, W. E. Novak, and A. S. Peterson. Feature-Oriented Domain Analysis (FODA) Feasibility Study. Technical report, Carnegie-Mellon University Software Engineering Institute, 1990.
17. E. Kuun. Open Standards for CBTC and CBTC Radio Based Communications. In *APTA Rail Rail Transit Conference Proceedings*, 2004.
18. G. Pascault. Response to the Driverless and CBTC growing market. http://goo.gl/fAL5n, 2011.
19. R. D. Pascoe and T. N. Eichorn. What is Communication-Based Train Control? *IEEE Vehicular Technology Magazine*, 2009.
20. K. Pohl, G. Böckle, and F. J. v. d. Linden. *Software Product Line Engineering: Foundations, Principles and Techniques*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.
21. Siemens Transportation Systems. Trainguard MT CBTC. http://goo.gl/Xi0h0, 2006. The Moving Block Communications Based Train Control Solution.
22. Siemens Transportation Systems. High availabilitytrain to wayside communication system for metro applications. http://goo.gl/PKG9i, 2008.
23. Siemens Transportation Systems. Airlink - High performance wireless broadband onboard connection. http://goo.gl/oRNEB, 2010.
24. Signalling Solutions Limited. URBALIS Communication Based Train Control (CBTC) Delivery Performance and Flexibility. http://goo.gl/G3hEe, 2009.
25. I. Silajev. *Basic Signalling Principles for System Engineers*. Rail Signalling Solutions, 2010.
26. J. S. Stover. CITYFLO 650 System Overview. http://goo.gl/e26SZ, 2006.
27. Thales Transportation. Seltrac Brochure. http://goo.gl/OjhvK, 2009.
28. H. Wang and S. Liu. Modeling Communications Based Train Control system: A case study. In *Proc. of ICIMA*, pages 453–456, 2010.
29. C. Yang, J. Lim, J. Um, J. Han, Y. Bang, H. Kim, Y. Yun, C. Kim, and G. Cho, Y. Developing CBTC Software Using Model-Driven Development Approach. In *Proc. of WCRR*, 2008.
30. L. Yuan, T. Tang, and K. Li. Modelling and Verification of the System Requirement Specification of Train Control System Using SDL. In *Proc. of ISADS*, pages 81–85, 2011.