How the EULYNX Data Prep standard can improve railway robustness

Dr. Ir. Bob Janssen, AMIRSE, EULYNX

SUMMARY

EULYNX data prep provides a model for representing data needed to build and configure signalling systems. The primary use will be to support the automated exchange of to-build and as-built data between Infrastructure Managers and signalling suppliers. Other use cases that heavily rely on complex software and easy access to reliable and precise data will benefit such as simulation, traffic analysis, asset management or maintenance.

Presently, IM's produce paper plans and proprietary data. Signalling suppliers need to ingest and interpret the data, a process that is as error prone as the humans that do the work. In the future, signalling tools can ingest the data automatically. Moreover, the model prescribes the semantics of the data leaving little room for misinterpretation. The model development is carried by many European IM's, and by this virtue is likely to encourage the creation of software that can serve a large market as opposed to today's fragmented landscape. Widely used software packages based on EULYNX data prep can thus automate many data-heavy processes, preventing data-related mishaps that plague (re-)signalling projects. Shorter and more reliable cycles in (re-)signalling work will in turn make railway operations more robust. Reliable and precise data have other uses that increase robustness of the network. Preventively finding single points of failure and pinpointing root causes of failures can be automated if suitable data are available.

1 INTRODUCTION AND RATIONALE

EULYNX is a standing partnership of European infrastructure managers (IM) that standardises the interfaces between trackside Command, Control and Signalling systems (CCS) and peripheral field elements.

The aim is stable "plug-and-rail" interoperability between core CCS equipment and distributed field element controllers in the system fringes. CCS relies heavily on computers and software algorithms with a life cycle in the order of years whereas railway hardware in the field has a life cycle in the order of decades.

Legacy CCS, especially the interlocking subsystem, is intertwined with the field elements. Decoupling CCS from the fringes has some advantages; IM's can acquire field element controllers from different suppliers and CCS can evolve more quickly because it has been released from the drag from the fringe. The notion of decoupling subsystems by introducing abstraction layers between subsystems is well accepted in IT development without which the rapid evolution of self-contained software packages would be impossible.

The benefit to IM and signalling suppliers can be compared to what USB did for IT in the 1990's. USB rapidly replaced proprietary interfaces like vendor-specific printer ports but created a big market for plug-and-play devices that rapidly turned into commodities.

EULYNX is organised in clusters, each of which standardises the interface with a given type of field element, ranging from light signals to interfaces between two interlockings.

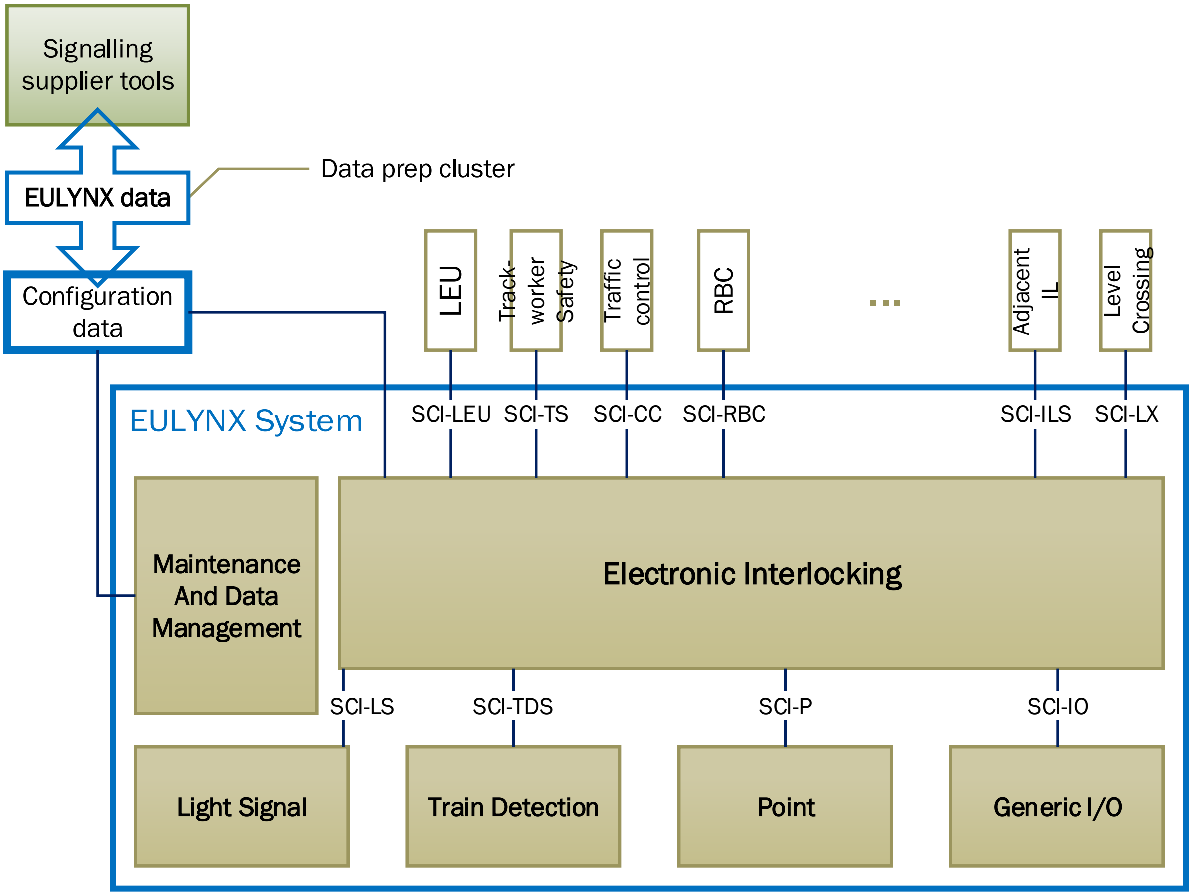

Figure 1: simplified EULYNX context diagram. EULYNX clusters are organised per standard communication interface (SCI)- interface. The data prep cluster defines the configuration data the supplier needs to build the signalling system.

A lesser known EULYNX cluster is the data preparation cluster. This is the exchange interface for configuration data. The central use case is to provide the information that a supplier needs to build a signalling system. This is a radical improvement from the present where the parties involved in (re-)signalling projects exchange heterogeneous and proprietary datasets (often on paper) like signalling plans and control tables. In the future, the signalling industry ingests standardized EULYNX data into their proprietary design tool set. As before, the signalling industry processes the data and then returns the enriched data, in EULYNX format, to the IMs who absorb this as-built data into their asset management systems.

IMs and signalling industry retain their proprietary formats and tooling but EULYNX harmonizes data exchange.

Automating the data transfer process through well-defined standard data structures will be beneficial not only to signalling suppliers and IM's but other data users stand to benefit from high quality data, for instance for building simulations for capacity management.

2 WHAT IS DATA PREP

Data preparation, also known as data engineering, dimensionnement in French and Projektierung in German is the process of customising the configuration data of a specific technical installation. Thus, data prep is the essence of tailoring a generic signalling system to control a specific railway yard.

Data prep involves compiling lists of controllable field elements and relations. Think of lists of signals, points, sections, block interfaces, etc. Relations can be topological or functional. Topological relations identify for instance the field elements a train encounters when travelling from an entry to the exit signal. A functional relation can state that a signal protects a given section or point. Obviously, a signalling system must be aware of both topological and functional relations.

Note that the logical rules by which a signalling system controls signals and points are not part of data prep because they are defined in national rules and regulations that barely ever change. Putting the logic rules in software is a one-off exercise but data prepping individual yards is a recurring exercise.

A good feel for the need for data can be had when considering the questions that suppliers and signalling engineers face when building an installation:

- determine a bill of quantity (BoQ)
- where to erect signals
- which are the desired routes and speeds
- What type of signals must be ordered
- Reliably compute the length of cable we'll need laying
- What routes are needed for the kind of operation we envisage

Experienced signalling engineers can think of many more use cases and the associated data needs. Also note that different IM's will use different subsets of the data. Suppliers of signalling equipment in the Netherlands are supposed to provide "up to the cable termination rack" whereas in Germany, a supplier can be tasked to install and commission the field elements. The EULYNX data prep model is designed to handle this pick-and-mix approach.

3 WHAT AUTOMATED DATA PREP IMPROVES

Traditionally, topological and functional information is stored in a wide variety of documents. Each Infrastructure Manager (IM) has its traditional way of drawing and formatting plans, lists and paper documents.

Suppliers of signalling equipment pick data from these documents. Putting the data into the supplier's tooling is a specialist job but needn't be overly expensive. However, this is mostly a manual process and finding the errors is expensive. An update to the input documents implies that the process of ingesting and checking the resulting system starts over. In other words, the data transfer process compares unfavourably to modern best practices, like for instance online configuring and ordering equipment from a web shop.

Legacy light signalling systems needs relatively little information compared to ETCS' need for profiles that allow trains to compute speed and braking curves. ETCS needs more accuracy, reliability and detail. Of course, one can reliably extract information from paper plans or from spreadsheets by setting up stringent control and signing off procedures. This adds complexity and scope for costly error.

The waterfall method of completely designing a yard and the set of data prep files that is handed over to the supplier seems outdated in the age of "Agile development". Fast design cycles with an efficient transfer of data sets between engineers from design all the way to signing off seems the only way of mastering increasingly complex signalling systems. Shifting a signal in the field costs thousands of times more than shifting a signal on a computer monitor. The need is paramount to remove fallible man from the loop and detect errors at the earliest possible design stage. (Bosschaart, 2013). The probability that data are corrupted somewhere in the process from collection, e.g. site survey, to RBC engineering dataset must be sufficiently low. This is easier to prove when the process is fully automated.

Finally, substantial cost is hidden in the supply industry's processes. Quoting to complex tenders presently involves tedious manual compilation of a Bill of Quantity. Suppliers who are active in different countries have insufficient incentive to invest in tooling for each national format and uncertainties in a BoQ will somehow increase the cost further down the line, e.g. in the shape of a risk premium. Suppliers need to train specialists capable of dealing with varying national signalling formats. This kind of specialisation impedes flexibility because switching between signalling projects in different countries and signalling cultures isn't easy.

Removing the human factor from the process by moving away from people picking data from maps and plans can greatly accelerate the data processing process as well as make it more robust.

4 HOW TO USE SITE DATA FOR ECONOMICALLY ROBUST DESIGN AND MAINTENANCE

Signalling failures have repercussions on operations. A failed power supply will typically affect a subset of the network, failed point detection will affect several routes. Designers take counter measures like redundant power supply, double cable paths, or alternative train routes. These technical measures are most often based on rules of thumb, but the economic benefits are hard to quantify. An economically robust design should offer optimal balance between the cost of lost travel time and the investment in the technical mitigation measures. Simulation programs can estimate the economic cost of train delays, e.g. due to the unavailability of a route. A simulation program can only be as good as the data on which it is based. Computer Aided Railway Engineering (CARE) applications can be leveraged to quickly compute the cost of mitigating measures like inserting a set of points. Easily exchanged site data are the link that is now missing between these simulation programs and CARE applications. In a foreseeable future, software programmers can merge simulation and CARE programs that use EULYNX data and estimate travel time gained against the cost of the mitigating measure. This allows a reasoned and quantified optimization of robustness of the network in both economic and technical terms.

The inverse approach can make an existing network more economically robust: one can rank potential failures according to economic cost. A potential failure that causes maximum havoc should receive more attention than a potential failure that causes little disruption. This "techno-economic" approach helps IM's to objectively identify and rank weaknesses in their network against the cost of merit mitigating measures. This approach can for instance unveil that a ruptured power supply cable would cause significant loss of travel time whereas a point, that is costly to maintain, barely contributes. This kind of stress-testing the network from a desktop simulation helps the IM to set objective priorities in spending scarce budget.

The availability of good data in a uniform format will entice competing software manufacturers to build smart algorithms that can serve a Europe-wide market. The fragmentation of the signalling systems, and the data that describe them, has held back the development of such a market. The upcoming EULYNX data prep format is likely to find uses like the ones described here. Hopefully, IM's will be able to buy applications off-the-shelf that help optimize allocation of scarce funding to the most robust technical designs and maintenance strategies.

The claims of this paragraph are based on the idea that all objects in a railway network are somehow linked and that the data must be structured in a way that allows algorithms to automatically answer what-if questions like "what signals and points will fail if power supply x fails, which routes are knocked out and which routes will still be available ?". Structuring and relating data is the subject of the next paragraphs.

5 INTRODUCING THE RAIL TOPO MODEL

The RailTopoModel (RTM) (UIC, 2017) provides the topological foundations on which EULYNX data prep is built.

Car navigation systems suggest that computers are good at reading roadmaps. However, a human-readable roadmap is just a picture of wiggly lines, dots and spots that a computer can't interpret like humans do.

So, how does your navigation system compute the shortest way home? The answer is of course that the computer doesn't represents the information in the map as a picture.

The first step is to try and describe in plain words what we actually observe in a map: a map shows a network, or graph, of roads that connect places. Places have attributes such as town name and geographical coordinates. Roads have lengths and are of given type and maybe a speed limit.

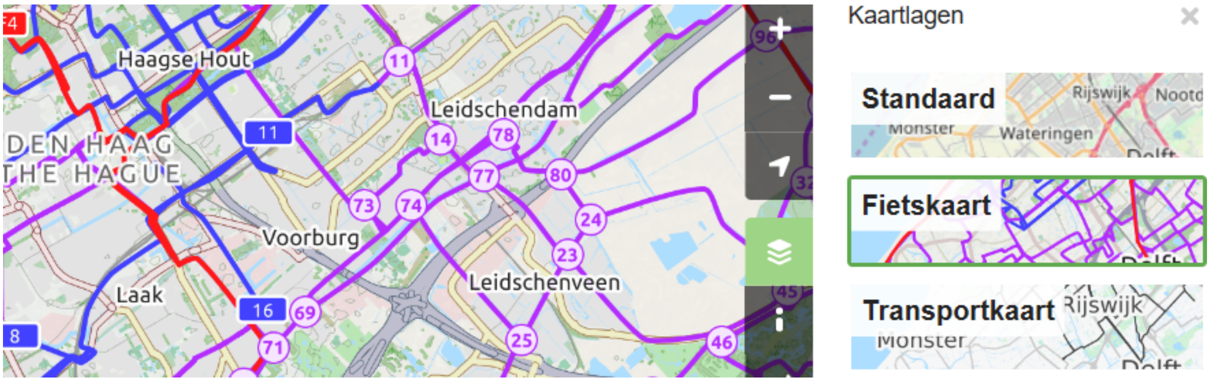

Figure 2: the map's topology carries business objects in a graph model (source: OpenStreetMap)

This way of thinking about information gives a first clue to the exercise of translating topology and topography into data useable for computing the shortest route.

We can easily translate places and connecting roads into a graph. Graph algorithms then compute the shortest path from location to destination.

A common pitfall when converting rail maps into graphs is to interpret nodes as points and tracks as edges. This makes graph-walking algorithms fail as below figure shows. A naive graph walking algorithm will trace a path through the network from edge to node to edge and so forth. However, whereas the algorithm will walk sharp angles, a train can't jump from one branch of a point to the other. This graph doesn't inform which node-edge connections are navigable.

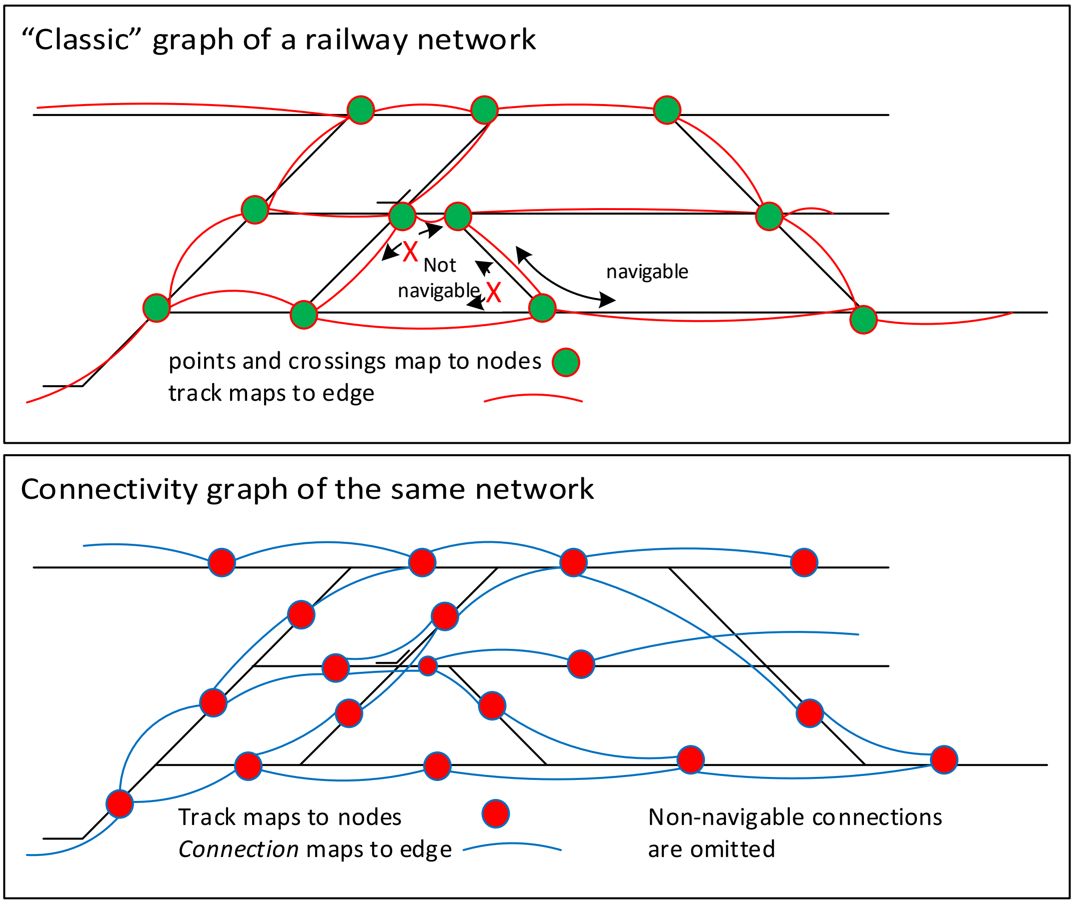

Figure 3: Two ways of representing a railway network in a graph. The classic graph doesn't inform about navigability. The RTM graph represents tracks as nodes and navigable connections as edges.

Various solutions have been invented to deal with this seemingly simple problem including double nodes and connections within nodes.

The RTM connectivity graph equates a segment of track to a node and the navigable connections between the nodes are the edges. Thus, a train moves from node via an edge to next node. Now, it is easy to find a path through the network by starting from a node via an edge to the next node. When there's no navigable connection, there's no edge.

Modelling half- and full-slip crossings is easy. In RTM, a node is a track segment and is called a *Linear Element.* An edge that relates two track segments is called a *Positioned Relation*. The navigability is an attribute of the positioned relation.

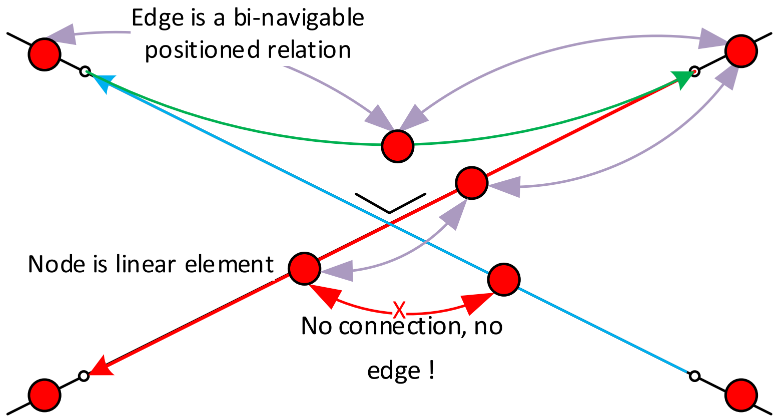

Figure 4: A single slip crossing in RTM graph notation (not all edges are shown).

Now that we have a topology, we can *attach* other objects to the topology. RTM calls these *Net Entities*, as opposed to the *Net Elements* that carry the topology. In a nutshell, a Net Entity with a spot location such as a signal is given an *intrinsic position on the Linear Element* in the range 0 to 1 where 0 is the start and 1 is the end of the track.

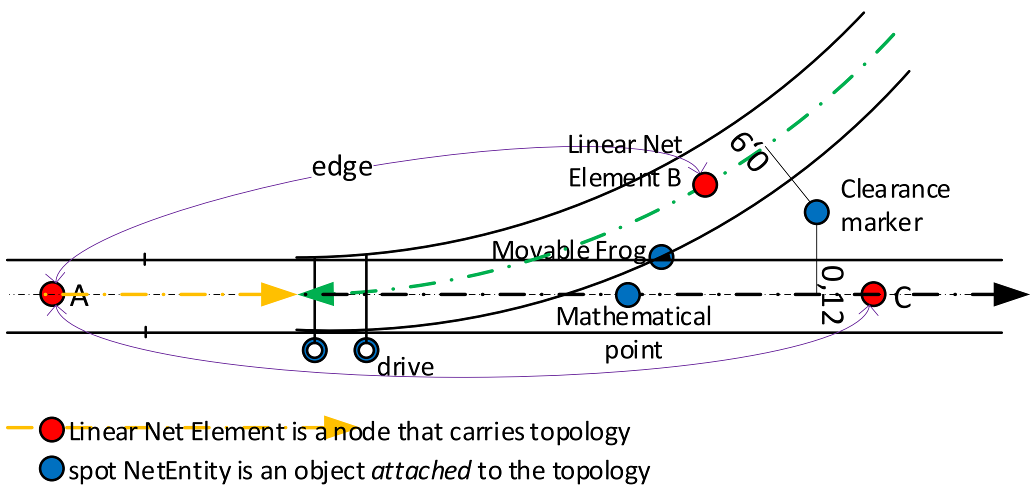

Figure 5: identifying objects in a point in terms of RTM

Many physical objects are punctual. Positions are given as a fraction of the length of one or more line segments. For instance, the clearance marker between the diverging branches of the point has a position 0,9 on linear element B and a position 0,12 on linear element C. We can also attach coordinates to the position, e.g. GPS coordinates or a chainage information.

A linear entity such as a speed segment or a route body is attached to the linear elements in a similar way by stating that it starts at position x on linear element A and ends at position y on linear element B.

This graph representation of the rail topology stored in the RTM format allows algorithms to find routes between entry and exit signal plus the elements that the train will encounter on the way. This is of course essential information for planning and configuring a signalling system. A linear entity can also have a length allowing computation of travel distances or compiling ETCS profiles.

EULYNX has adopted RTM as a basis on top of which we place overlays of information specific to signalling. Overlays are like transparent sheets that show information relevant to particular use cases.

6 HOW TO REPRESENT INFORMATION AND KNOW-HOW

People like drawing symbols on sheets of paper. A legend and a manual explain the human reader what he's looking at and how to interpret symbols. The first exercise is to describe in words this understanding in a way that human experts can understand and agree on.

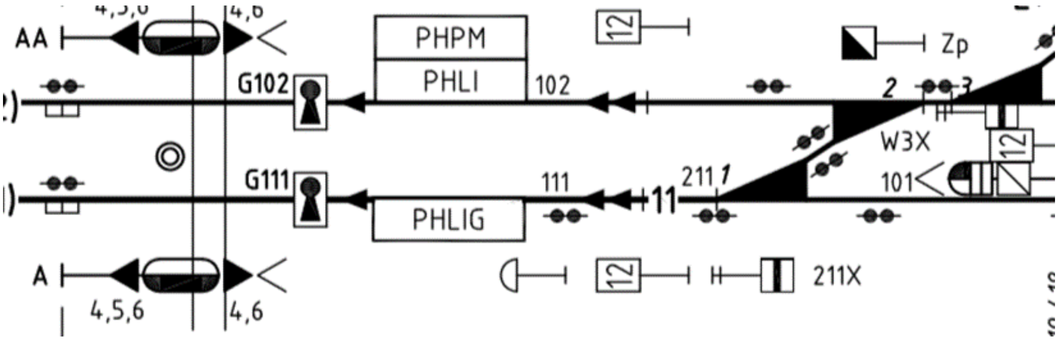

Figure 6: Excerpt of a signalling plan (source DB Netze)

An image is worth a thousand words, but our goal is to put in words what the signal plan is telling us:

- A is an object of type signal
- A is a signal of type *combined* and has speed indicators
- *1*, *2* and *3* are objects of type point
- Signal AA protects point *2* and *3*

And so on. These are short phrases with a subject (a noun), a predicate and an object. This kind of *painting by words* forces domain experts to explicitly state information that is implicit in plans.

We next visualise this information by means of the Unified Modelling Language. UML class diagrams capture classes and relations between classes, a class being a type of thing. A UML class diagram is a way of storing the meaning and relations between classes that is visually appealing and easily understood by humans.

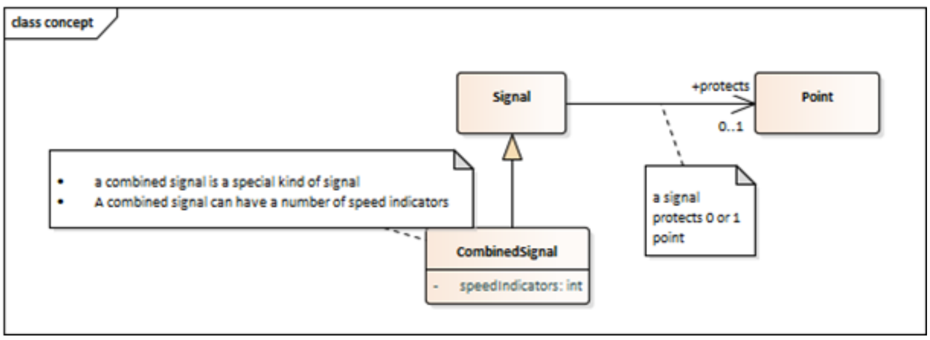

Figure 7: conceptual sample of a UML class diagram presenting domain knowledge

The amount of information and relations between classes that are stored in the signal plans is vast. A signal plan truly is worth thousands of words.

Note that the EULYNX model isn't restricted to tangible railway objects but also covers abstract notions like speed profiles, routes and flank protection.

7 HOW TO PUT INFORMATION INTO COMPUTERS

Computers can't read maps but they excel at parsing structured text. The best known format is of course the eXtended Markup Language, XML. The example from the previous paragraph above can be written like so

```
<CombinedSignal speedIndicators="3">A
        <protects ref=idPoint1/>
</CombinedSignal>
<Point id=idPoint1/>
```

Careful inspection reveals how the information stored in this XML sample is written according to the structure defined in the previous paragraph. XML uses tags to classify objects, like `CombinedSignal` and a relation like `protects`. Note that the "protects" tag is enclosed within the `CombinedSignal` to indicate the fact that the protection relation is an intrinsic part of the signal.

Another popular format that is easy to read is yaml ("YAML ain't another mark-up language") that uses key-value pairs and indentation.

```
---
CombinedSignal: A
    protects: *idPoint1
Point: &idPoint1
```

This demonstrates how the statement "signal A protects point 1" can be written in structured text.

The notation `*idPoint` is a reference to the point, just like a hyperlink in a text document.

The line with `protects` is indented with respect to the previous header line `CombinedSignal: A`. This indicates that the relation `protects: *idPoint1` is part of `CombinedSignal A`. If we delete the header `CombinedSignal: A`, the `protects: *idPoint1` relation also disappears but the `Point: &idPoint` header remains. This is just like in real life: when we remove a signal, the point is still there but the protection relation between signal and point is gone.

A computer algorithm can parse text files like the above. Algorithms can leverage the semantics and relations that are present in the structured text data to answer queries like

- how many signals, points, sections in the yard (interesting for quantity survey)
- which signals protect point 2 ? (interesting for implementing flank protection)
- which signals can a train reach departing from signal 1 (route finding)
- which signals will the train encounter en route ?
- what position are needed by points in a given route ?
- which routes are mutually conflicting ?

The aim of the EULYNX data preparation cluster is to capture all the domain knowledge needed to answer the requirements that need answering so that a supplier can build a signalling system for the specific yard.

8 EULYNX DATA PREP DELIVERABLES

The UML class diagrams contain all the domain knowledge, the class semantics and relations between classes. The next step is to transform the UML into templates for structured text.

The template for XML text data is called XML Schema Definition language (XSD, the main deliverable of EULYNX data prep). In a nutshell, an XML file that is an *instance of the XSD template* must be formed to respect the associated XSD schema.

EULYNX transforms the UML into XSD. Given the XSD, tooling checks that XML files that claim to match the UML are correctly formed. This verification is important so that tools that read the XML files can trust to find and *interpret* the information they're looking for at the correct spot inside the structured text data.

XSD is only one EULYNX deliverable that can be used to leverage the UML. In the foreseeable future, we may choose to transform the UML not only into XSD but also into Object Oriented programming languages like Java or python. This opens the door to other interesting uses like simulation programs that need information about railway objects and their mutual relations. Information that is stored ("persisted" in IT-speak) in XML files, can be read into "live" software objects in computer memory.

Currently, the UML classes only carry static information in the shape of attributes and relations with other classes. One can add methods (or functions), for instance, a method "open" or "close" to a class Signal and "throw(left/right)" to a Point class. This turns the static UML model into a dynamic model that is readily transformed into executable code. Combining static configuration obtained from data prep with dynamic software code, one can generate simulation software that is guaranteed to match the real yard by the push of a button.

The fact that so many IM's support EULYNX standard covers make this proposition even more attractive to software developers because EULYNX is designed to be Europe-wide and is likely to become a worldwide standard.

9 HOW TO DEAL WITH NATIONAL VARIATIONS

Signalling systems between countries seemingly have very little in common. Signals vary widely between countries. How to deal with these variations ?

The answer has been hinted at briefly above. The IT best practice for dealing with variability is specialisation. The reasoning follows the lines set out in below example:

1. Physical and non-physical signals are a special kind of signal.
2. Physical signals specialise into active signals and (passive) signs
3. light and mechanical signals are a special kind of active signal.
4. A sign is a kind of physical signal
5. A milepost is a kind of sign that shows a value as a string.

Every IM has these signs, light- and mechanical signals so we can safely say that these classes of signals are common to all.

Breaking down further, one enters national realms because specific IM's need extra information.

6. A ProRail fictitious signal is a special kind of fictitious signal that has properties *specific to the Netherlands*.
7. A DB fictitious signal has properties *specific to Germany*.

The latter two suggest the need to attach specific national functions to a fictitious signal that has been specialised from the EULYNX fictitious signal. For this purpose, EULYNX has created a set of national domains, also known as namespaces, where information that is of national concern is stored separately.

This approach of specialisation and delegating information to the national realms only when needed has proven to be very effective. The common part of the EULYNX model is so far substantially bigger than the national namespaces.

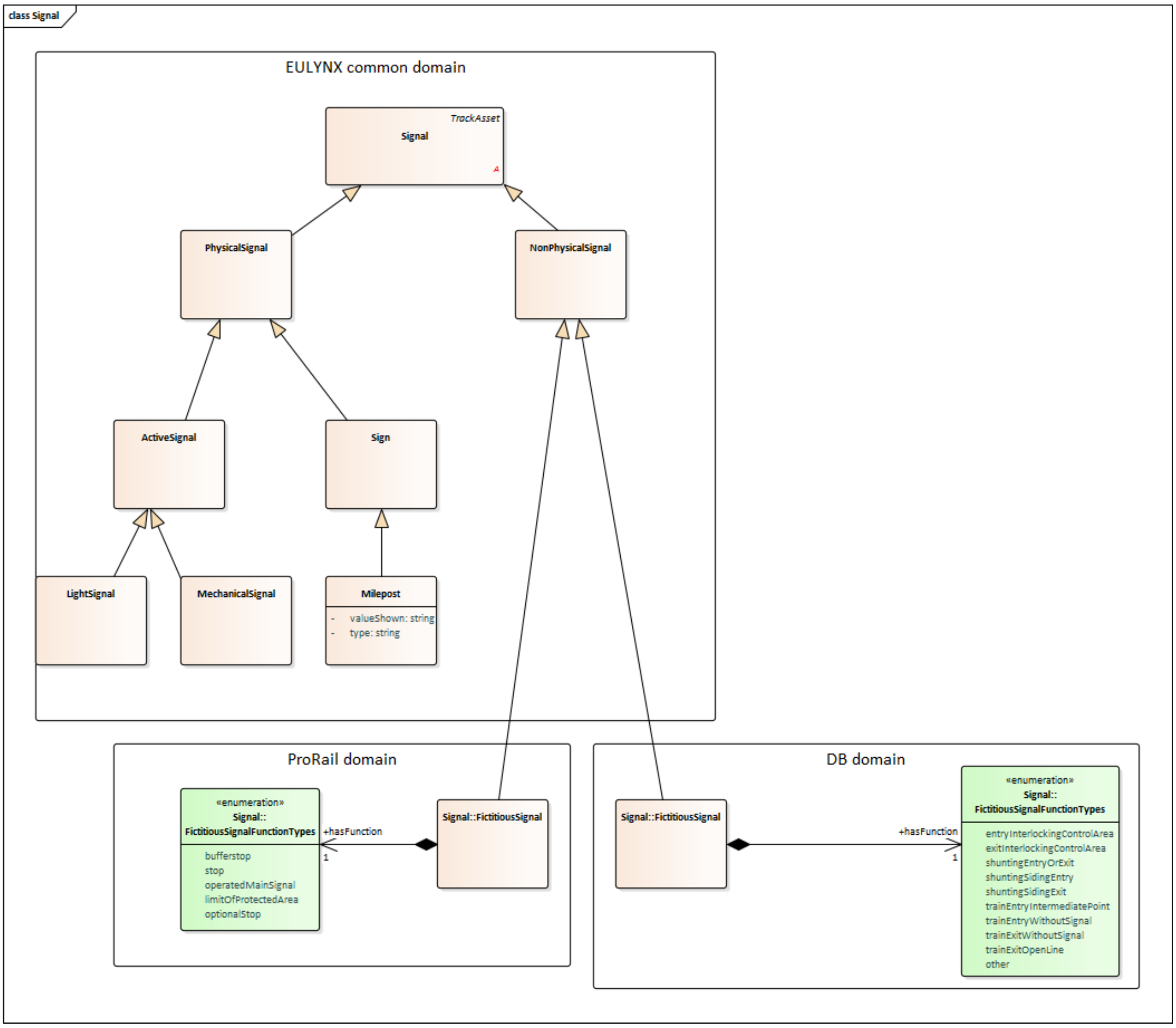

Figure 8: taxonomy of railway signals and specialisation into national realms

The approach of taxonomy is widely used, for instance in biology, to classify super-classes of species with common properties. These species are then sub-classed to capture all kinds of species that inherit properties from the super-classes.

10 ROUTE MAP AND OUTLOOK

The effort of reuniting 160+ years of signalling traditions is considerable. Eliciting domain knowledge from IM's that has grown over such a long period of time and describing in a common model takes time. At the time of writing, the route map is being revised and so far, the EULYNX data prep cluster has modelled the core classes of route, signal and points.

The EULYNX data prep model has gained substantial momentum and attention in the rapidly expanding universe of rail data modelling. Notably, there's a relation to BIM that becomes clear when considering a requirement that is exceedingly relevant to signalling: there mustn't be any obstructions between the driver and the signal. Obviously, BIM with its longstanding know-how of 3D modelling of buildings and conflict detection is well placed to answer this query. The same goes for detecting conflicts when installing cables, checking space for point drives or foundations for signal posts. This is where EULYNX business ends and BIM business starts.

Other data related initiatives are Shift2Rail who need high quality data for software development for simulations and timetable optimisation.

Automatic Train Operation (ATO) is another obvious important consumer of data. ATO data differ somewhat from EULYNX signalling data which is revealed when one thinks about ATO use cases like "the train stops at a platform so that the doors open at a *defined* location" or "the *optimal* speed profile for an ATO train depends on the delay". This indicates that not all operational aspects of ATO are necessarily of concern to the safety related signalling business that is EULYNX'.

Finally, the Reference CCS Architecture (EUG and EULYNX partners, 2019) is a newish initiative that aims to streamline, unify and accelerate the signalling business. EULYNX' approach of factoring out and hiding the signalling field objects away from the CCS system behind an abstraction layer plays a central role in RCA. And so does data engineering using the EULYNX data prep because it is seen to be crucial to automate the process of capturing and validating data.

11 REFERENCES

1. Bosschaart, M., 2013. *Lean Engineering of Rail Interlocking Systems,* Delft: TU Delft.
2. EUG and EULYNX partners, 2019. *RCA Alpha – Architecture Overview,* Brussel: s.n.
3. UIC, 2017. *RailTopoModel.* [Online] Available at: https://uic.org/railtopomodel [Accessed 2019].