Horizon 2020
European Union Funding
for Research & Innovation

In2Rail

In2Rail

Project Title:	**INNOVATIVE INTELLIGENT RAIL**
Starting date:	01/05/2015
Duration in months:	36
Call (part) identifier:	H2020-MG-2014
Grant agreement no:	635900

Deliverable D9.1

Asset status representation

Due date of deliverable	Month 18
Actual submission date	31 October 2016
Organization name of lead contractor for this deliverable	ASTS
Dissemination level	PU
Revision	Final

Authors

		Details of contribution
Author(s)	**ASTS** C. Dambra M. Giaroli G. L. Zanella M. Garresio F. Papa	Responsible for the deliverable. Sections on: • Switch • Level crossing Contributions to sections on model choices and asset status (switch)
Contributor(s)	**UNIGE** E. Fumeo	Support in the design of the approach and in the categorization of railway assets
	MERMEC E. Sgura	Sections on • Catenary • Track • Addition of measurements from in service trains or measuring vehicles
	UoB J. Easton	Sections on • Bridge • Tunnel • Embankments • Line sections Contributions to sections on review of models, model choices, and asset status (level crossing & switch)
	DLR C. Rahmig	Interface to railML, contributions to model choices
	NR I. Coleman I. Dean N. Hoodbhoy	Contribution to the definition of the TMS-critical variables Review of the document
	RFI R. Canepa S. Petralli	Contribution to the definition of the TMS-critical variables
	TRV E. Neldemo	Contribution to the definition of the TMS-critical variables
	SICS M. Joborn	Contribution to the definition of the TMS-critical variables
	SIE Stefan Wegele	Review of the document

Executive Summary

This report presents the sets of variables that are of interest to the TMS when considering the availability and status of nine key infrastructure assets on the railways, switches, crossings, track, catenary, bridges, tunnels, embankments, line sections and level crossings. Data related to these variables are described and classified as either static or dynamic.

Existing models capable of representing the static and dynamic data elements are then reviewed, and with none found to adequately represent both classifications of data independently a hybrid approach is proposed, under which the static elements of the data are described using railML, while the dynamic elements are described using the Open Geospatial Consortium's SensorML model, part of the Sensor Web Enablement suite of standards.

Finally, worked examples are provided showing how the approach may be applied to two of the asset types, the level crossing, and the switch.

TABLE OF CONTENTS

Abbreviations and acronyms

Abbreviation / Acronyms	Description
C4R	Capacity 4 Rail, an EU FP7 research project
EPSG	European Petroleum Survey Group
ERA	European Rail Authority
GML	Geography Markup Language
GNSS	Global Navigation Satellite Systems
GPS	Global Positioning System
IM	Infrastructure Manager
INSPIRE	Infrastructure for Spatial Information in Europe
JSON	JavaScript Object Notation
LC	Level Crossing
LVDT	Linear Variable Differential Transformer
N.D.	Not Defined
OGC	Open Geospatial Consortium
OP	Operational Point
POE	Power Over Ethernet
REST	Representational state transfer
RINF	Register of Infrastructure
SSN	Semantic Sensor Network
SWE	Sensor Web Enablement
SWEET	Semantic Web for Earth and Environmental Terminology
TMS	Traffic Management System
UIC	Union Internationale des Chemins de fer
W3C	World Wide Web Consortium
WGS84	World Geodetic System 1984
XML	eXtensible Markup Language
XSD	XML Schema Definition

1. Background

The present document constitutes the Deliverable D9.1 “Asset Status Representation” in the framework of the Project titled “Innovative Intelligent Rail” (Project Acronym: In2Rail; Grant Agreement No 635900).

This document has been prepared to provide recommendations for a data notation that can be used to represent the dynamic status of infrastructure assets within the In2Rail system, and specifically within the context of Work Package 9 (WP9) of the project. The notation is building on existing work wherever practicable, and, in particular, on existing open standards in the area supported by key stakeholder groups such as the UIC and ISO.

The document will be the basis for a more extended work in Shift2Rail TD3.6 where all main IMs are present either directly (DB, SNCF) or through consortia (EUROC).

2. Objective / Aim

This report aims to describe a data representation for the status of assets within the railway infrastructure.

The approach to be taken will involve:

- Identification of the attributes needed to represent the operational status of a set of nine railway assets relevant to the TMS, as defined in other work packages within the In2Rail project;
- A review of existing modelling approaches to the problem area, and production of recommendations for modelling of assets within In2Rail WP9;
- Production of "proof of concept" examples illustrating the use of the proposed approach.

3. Asset status data

3.1. Governance Structure

This section aims at identifying, for each asset of the railway infrastructure, the variables to be monitored and that could play a role in the TMS and Maintenance Management decision-making processes.

The assets considered in this document are:

1) Switch
2) Crossing
3) Track (Rail)
4) Catenary
5) Bridge
6) Tunnel
7) Embankments
8) Line sections
9) Level crossing

3.2. Disclaimer

This document introduces a taxonomy of the assets of the railway infrastructure referring to an assets' nomenclature widely used in the railway sector. The taxonomy has the only purpose of logically structure the subdivision of the variables to be monitored in the railway infrastructure per macro components of the involved assets.

Therefore, taxonomy does not pretend to be recognised as a standard and does not consider all possible national variations and languages.

3.3. Asset data classification

First of all, the following distinction is provided between:

- **Static** data: related mainly to static characteristics of the asset under examination (e.g. GPS absolute position, ...);
- **Dynamic** data: data coming either from recordings of the usage of the asset (e.g. number of trains passed over the asset, ...) or from external devices/sensors (e.g. environmental temperature, rail profile measurements, ...) and maintenance operations (e.g. number of maintenance operations during lifetime, ...).

Moreover, dynamic data can be classified in the following way:

- Internal: measurements collected internally from the asset;
- Asset-related: measurements collected by sensors attached or strictly related to the asset;
- External: measurements from external sensors;

- Diagnostic: measurements collected by a passing diagnostic train and other diagnostic devices;
- Maintenance: data related to maintenance operations / actions on the asset.

The dynamic data is also characterised by their criticality when dealing with Traffic Management System (TMS) decisions. Each IM in WP9 (RFI, TRV and NR) has expressed its evaluation of each variable in terms of TMS-critical variable (a tick in the corresponding table).

3.4. Switch

This section includes the representation of the railroad switch based on its classification among all the physical railway assets, on the identification of its subcomponents and on the relevant data that should be collected to identify the functional status of the asset.

3.4.1. Asset classification

Switches are part of the infrastructure of a railway network, and they can be classified as specialized Track Equipment. Figure 3.1 shows the complete classification of the asset with respect to all the physical railway assets.

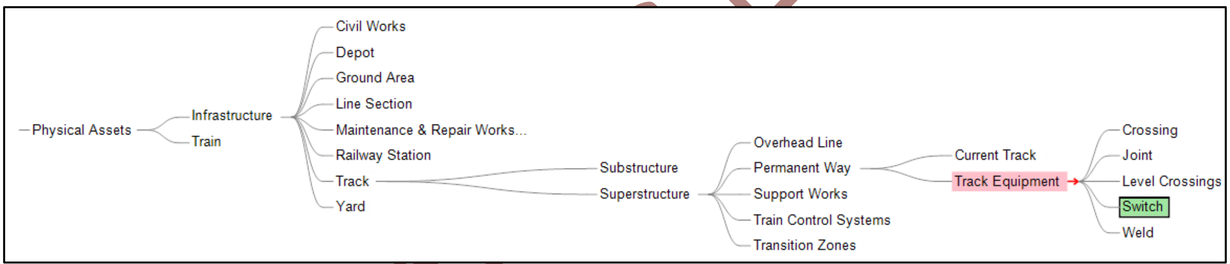


Figure 3.1: Position of Switch into the railway taxonomy

3.4.2. Asset sub-components

Railroad switches are composed of many sub-components, which are depicted in Figure 3.2. This picture shows the classification of the asset sub-components and their categorization in five different classes.

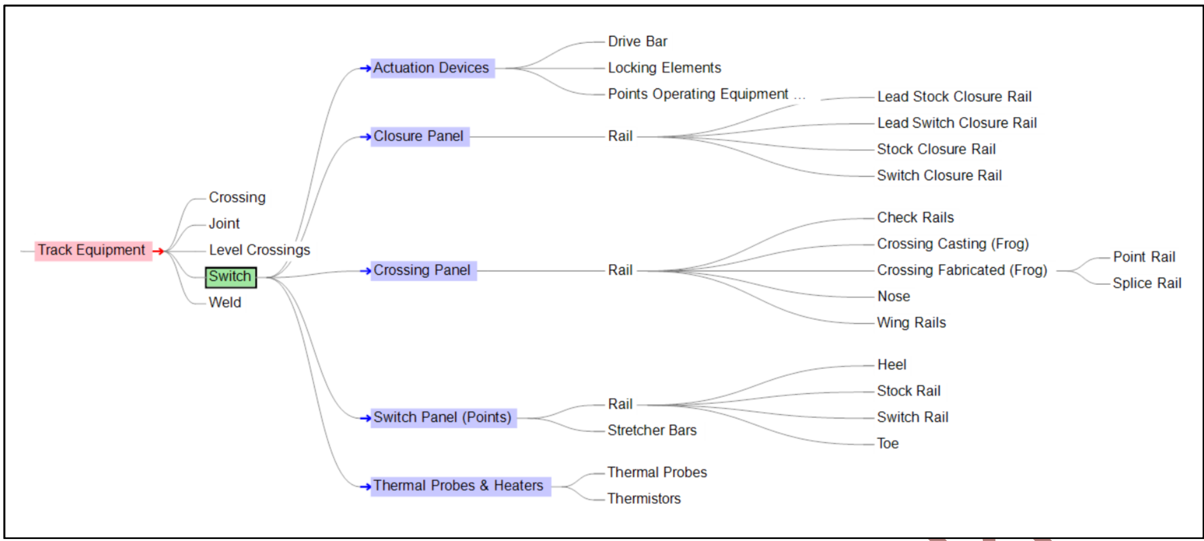


Figure 3.2: Switch sub-components classification

Moreover, Network Rail provided the technical picture of Figure 3.3, which depicts a switch and the associated labels for its components. The technical picture reflects the same subcomponents identified in the proposed asset classification.

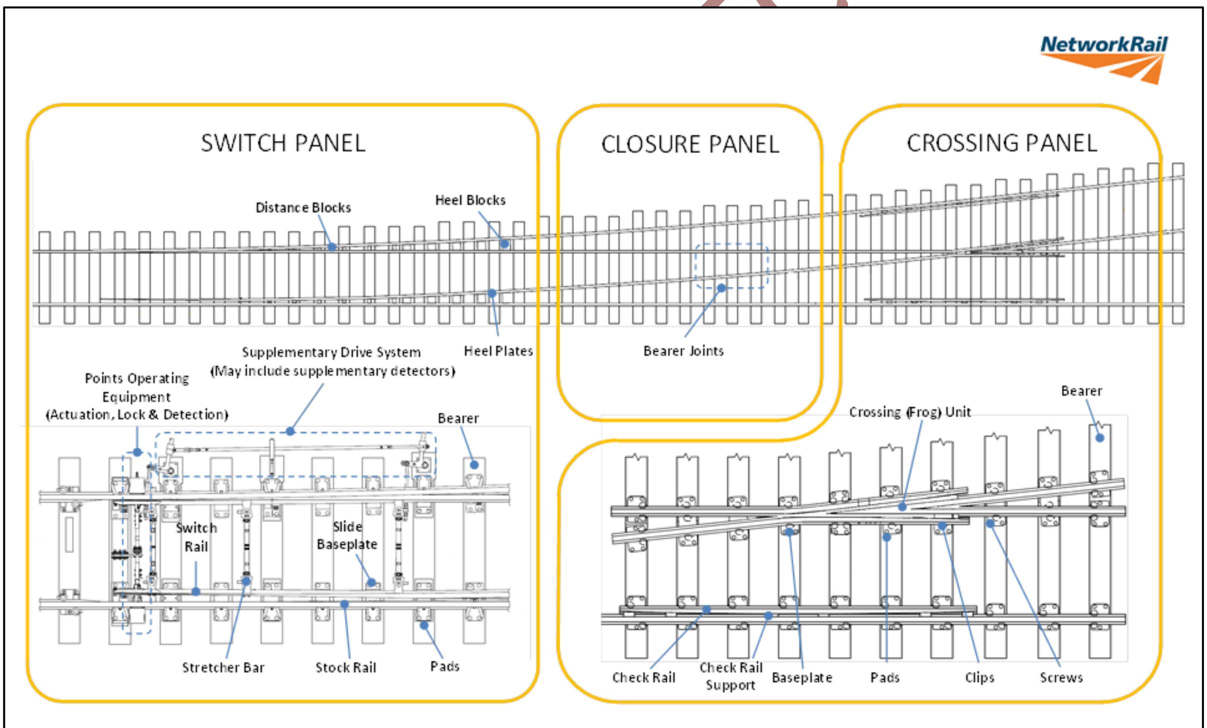
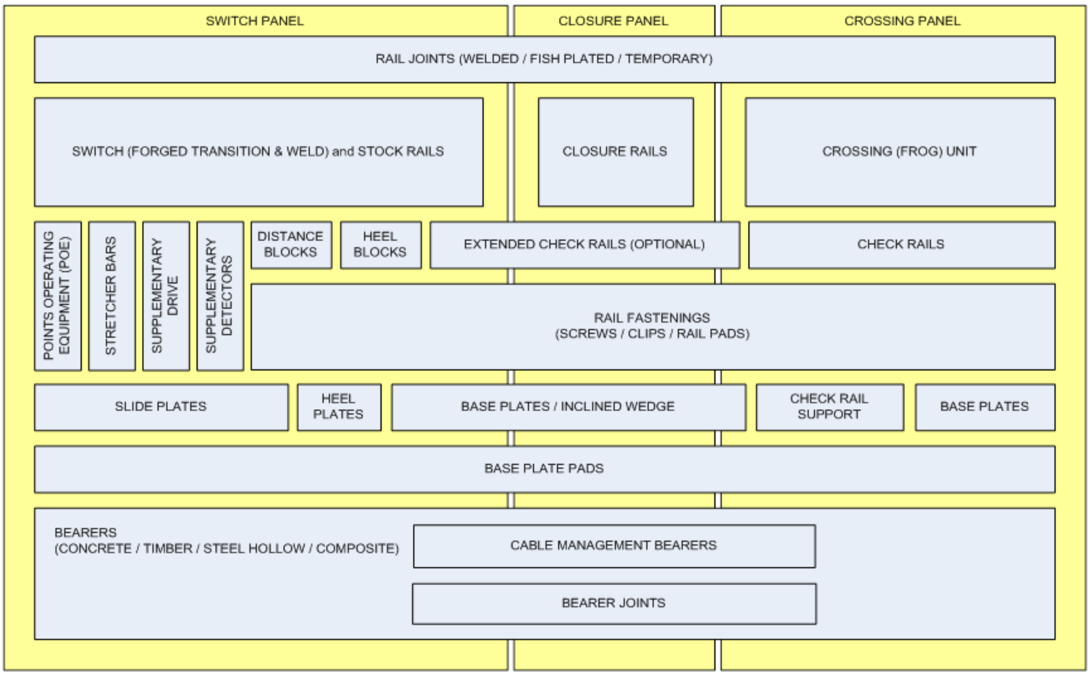


Figure 3.3: Technical picture of a Switch

Since this document does not have to be intended neither as a finalized document for a fully comprehensive classification of railway assets, nor as a new standard for a railway taxonomy, an alternative classification (depicted in Figure 3.4) is included for reference purposes. This classification of the switch has been developed by Network Rail, and contains some slightly differences from the one proposed in this document.

Finally, it is worth to recall that depending on countries, the terminology might differ significantly.



Schematic developed by the Network Rail S&C Systems Team

Figure 3.4: Graphical representation of the Network Rail classification

The following list recalls in textual form all the most relevant components of a railway switch, divided by the functions they implement or grouped by super-component:

- Actuation devices:
  - Drive bar,
  - Locking elements,
  - Point operating equipment – POE;
- Closure Panel:
  - Rail:
    - Lead Stock Closure Rail,
    - Lead Switch Closure Rail,
    - Stock Closure Rail,
    - Switch Closure Rail;
- Crossing Panel:
  - Rail:
    - Check Rails,
    - Crossing Casting (Frog),
    - Crossing Fabricated (Frog):
      - Point Rail,
      - Splice Rail;
    - Nose,
    - Wing Rails;
- Switch Panel (Points):
  - Rail:
    - Heel,

        - Switch Rail,
        - Stock Rail,
        - Toe;
    - Stretcher bars;
- Thermal probes and heaters:
    - Thermal probes,
    - Thermistors.

3.4.3. Data Tables

All the data mentioned in the following tables (or, at least, most of it) should be stored and collected regularly to create an historical database containing all the possible information related to the functional behaviour of the monitored devices over their entire lifetime.

STATIC Data			TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Type	Enumeration (ordinary, inside curved, outside curved, three way)	Switch			
Manufacturer Model[1]	String	Switch			
Normal Position	Enumeration (straight, left, right)	Switch (Rail)	✓		
Length	Decimal (m, mm, etc.)	Switch	✓		
Absolute Position	Geospatial	Switch	✓	✓	
Mileage	Decimal (km)	Switch		✓	
Nominal Switch Motor Voltage	Decimal (V)	Actuation devices - POE	✓		
Nominal Switch Motor Current	Decimal (A)	Actuation devices - POE	✓		
Nominal Switch Motor Power Consumption	Decimal (W)	Actuation devices - POE	✓		
Nominal manoeuvre time	Time	Actuation devices			
Locking nominal electromagnetic power	Decimal	Actuation devices – Locking elements			
Nominal Max Speed (different for each direction)	Decimal (km/h)	Switch	✓	✓	
Joints Type	Enumeration (welded, insulated, glued, etc.)	Switch - Points - Rail			
Joints Nominal	Decimal (mm)	Switch - Points - Rail			

[1] This variable is added for categorization purposes, so to be able to cluster different switches in to groups with similar characteristics.

STATIC Data			TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Displacement (if applicable[2])					
Construction date	Time	Switch			
Construction series	string	Switch			
Original Test Date	Time	Switch			
Installation Date	Time	Switch			

Table 3.1: Switch: static data

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Current position	Enumeration (straight, left, right[3])	Switch - Points - Rail	Internal	✓		
Device temperature	Decimal (°C)	Switch - Thermal probes and heaters	Internal	✓	✓	
Device status	Enumeration (OK, KO, No-Response, Sufferance[4], etc.)	Switch	Internal	✓	✓	✓
Environmental Humidity	Decimal (%)	Switch (environment)	External			
Environmental Pressure	Decimal (bars)	Switch (environment)	External			
Wind Speed	Decimal (m/s)	Switch (environment)	External			
Wind Direction	Decimal (degrees)	Switch (environment)	External			
Ballast status	Radar – Image	Switch (under-structure)	Diagnostic (Ground penetrating radar, etc.)			
Ballast inclination[5]	Decimal (degrees)	Switch (under-structure)	Asset-related (inclinometer)			
Ballast temperature	Decimal (°C)	Switch (under-structure)	Asset-related			
Flood alert - water	Decimal (mm)	Switch (under-	External	✓	✓	✓

[2] For example, welded joints do not have, obviously, any displacement between one rail and the next ones.
[3] This enumeration is designed to cope also with 3-way switches, although in many cases it might be sufficient to use "normal" and "reverse" as position categories (for example in UK, where there are very few 3-way switches).
[4] This value indicates a degraded state of the switch. The terminology might be different depending on the reference country.
[5] This variable allows monitoring any unwanted variation of inclinations of the ballast.

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
level		structure)				
Flood alert – digital	Binary	Switch (under-structure)	Asset-related (Track circuits)			
Snow/Ice detection	Binary	Switch (environment) Actuation devices	External	✓	✓	✓
Switch Motor Voltage	Decimal (V)	Actuation devices - POE	Internal			
Switch Motor Current	Decimal (A)	Actuation devices - POE	Internal			
Switch Electromagnet Voltage	Decimal (V)	Actuation devices – Locking Elements	Internal			
Switch Electromagnet Current	Decimal (A)	Actuation devices - Locking Elements	Internal			
Manoeuvre time	Time	Actuation devices	Internal	✓		
Electromagnet Peak Time	Time	Actuation devices - Locking Elements	Internal			
Manoeuvre total number	Integer	Actuation devices	Internal	✓		
Total axle passages (related to each branch) per direction	Integer	Switch - Points - Rail	Asset-related (Axle counter)	✓		
Total weight transited (related to each branch) per direction	Decimal (kg)	Switch - Points - Rail	Asset-related (Weight in motion)	✓		
Load per axle (each axle passed or aggregated measure, related to each branch) per direction	Decimal (kg)	Switch - Points - Rail	Asset-related (Weight in motion)			
Total train passages (related to each branch) per direction	Integer	Switch - Points - Rail	Asset-related (Traffic Management System)	✓		
Speed of passed trains (related to each branch) per direction	Decimal (km/h)	Switch - Points - Rail	Asset-related (train odometry)	✓		
Wheel weight transited (related to	Decimal (kg)	Switch - Points - Rail	Asset-related (Weight in			

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
each rail) per direction			motion)			
Gauge	Decimal (m, mm, etc.)	Switch - Points - Rail	Diagnostic	✓	✓	✓
Rail profile (height, width, etc.)	Decimal (mm)	Switch - Points - Rail	Diagnostic	✓	✓	✓
Rail profile	Image	Switch - Points - Rail	Diagnostic	✓	✓	✓
Joint status	Enumeration (OK, KO, creeped, etc.)	Switch - Points - Rail	Diagnostic	✓	✓	✓
Joint Displacement (if applicable)[6]	Decimal (mm)	Switch - Points - Rail	Diagnostic	✓	✓	✓
Joint visual status	Image	Switch - Points - Rail	Diagnostic	✓	✓	✓
Vibration / Accelerations	Array of Decimals (G)	Switch - Points - Rail (rail status, crack detection, etc.)	Diagnostic/ Asset-related (triggered)	✓		
Sounds	Array of Decimals (dB)	Switch - Points - Rail (rail status, crack detection, etc.)	Diagnostic / Asset-related (triggered)			
Friction[7]	Decimal (N)	Switch - Points - Rail	Diagnostic (tribometer)			
Number of maintenance interventions	Integer	Switch	Maintenance	✓		
Scheduled maintenance interventions frequency	Time (each "X" days, months, etc.)	Switch	Maintenance	✓		
Maintenance interventions date	Time	Switch	Maintenance	✓	✓	✓
Maintenance intervention start time	Time	Switch	Maintenance	✓	✓	✓
Maintenance intervention end time	Time	Switch	Maintenance	✓	✓	✓
Maintenance intervention type code	String (code)	Switch	Maintenance	✓		
Maintenance	String	Switch	Maintenance	✓	✓	✓

[6] See comment about the nominal displacement of joints in the static data table, which also includes joint type.
[7] This variable refers to the forces that a train can act on the track in order to produce a movement.

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
intervention description			(could be supplied by humans)			
Code for failure that determined a maintenance intervention	String (code)	Switch	Maintenance	✓		
Maintenance team that made the intervention	String (code)	Switch	Maintenance			
Moving switch blades bearings status (lubrication, wear, rust, etc.)[8]	N.D. (non-invasive techniques to be studied in In2Rail)	Switch – Actuation Devices	N.D. (non-invasive techniques to be studied in In2Rail)	✓		
Sand blocking moving parts	Image	Switch – Actuation Devices	Diagnostic	✓		
Ice blocking moving parts	Image	Switch – Actuation Devices	Diagnostic (either camera or thermograph)	✓		
Obstructions	Image	Switch – Actuation Devices	Diagnostic	✓		

Table 3.2: Switch: dynamic data

3.5. Crossing

This section includes the representation of the crossing based on its classification among all the physical railway assets, on the identification of its subcomponents and on the relevant data that should be collected to identify the functional status of the asset.

3.5.1. Asset classification

Crossings are part of the infrastructure of a railway network, and they can be classified as specialized Track Equipment. Figure 3.5 shows the complete classification of the asset with respect to all the physical railway assets.

[8] See Deliverables of Work Package 2 – Innovative S&C solutions of the In2Rail Project for more details

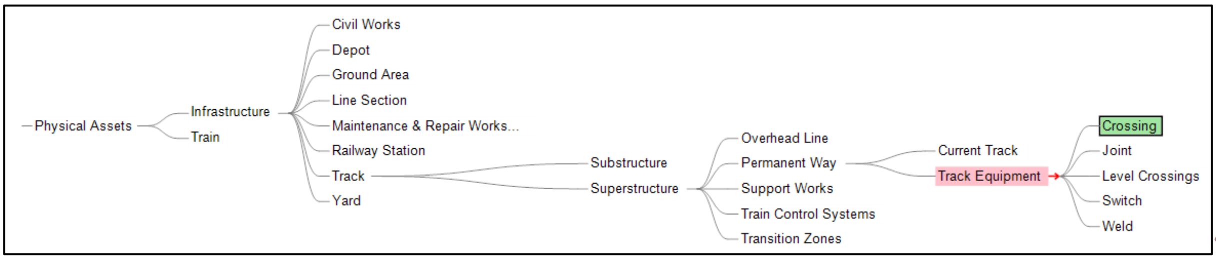


Figure 3.5: Position of Crossing into the railway taxonomy

3.5.2. Asset sub-components

Crossings are simple yet fundamental assets of a railway system that are composed of a few complex sub-components, as shown in Figure 3.6 and Figure 3.7. There are several types of crossings, which can be categorized by:

- Angle:
  - Right,
  - Obtuse,
  - Acute,
- Diversion:
  - Spring,
  - Crossover,
  - Scissors (Double),
  - Gathering lines,
  - Diamond.

Every different type of crossing can be divided in the subcomponents that have been identified. As an example, Figure 3.7 shows a diamond crossing schematically with labels identifying parts.

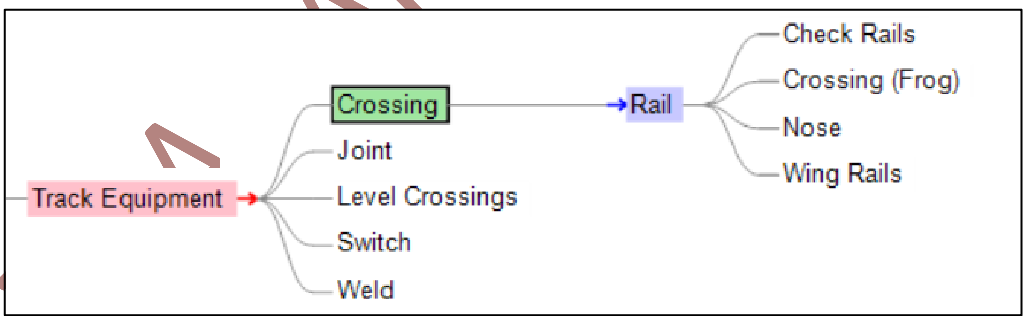


Figure 3.6: Crossings sub-components classification

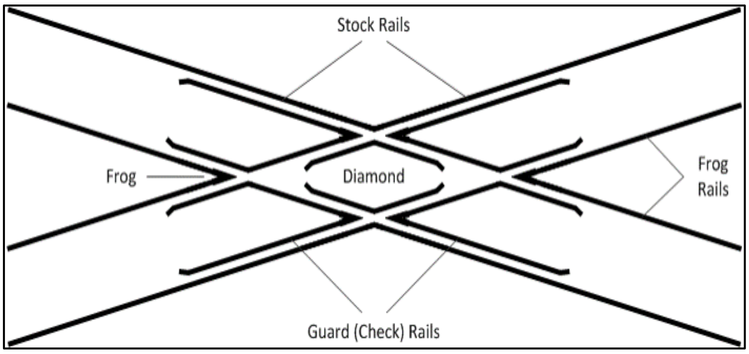


Figure 3.7: Diamond crossing technical picture

3.5.3. Data Tables

All the data mentioned in the following tables (or, at least, most of it) should be stored and collected regularly to create an historical database containing all the possible information related to the functional behaviour of the monitored devices over their entire lifetime.

STATIC Data			**TMS Critical Variables**		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Type - Angle	Enumeration (right, obtuse, acute)	Crossing			
Type - Diversion	Enumeration (Spring, Crossover, Scissors (Double), Gathering lines, Diamond)	Crossing			
Model	String	Crossing			
Absolute Position	Geospatial	Crossing	✓		
Mileage	Decimal (km)	Crossing	✓		
Construction date	Time	Crossing			
Construction series	string	Crossing			
Original Test Date	Time	Crossing			
Installation Date	Time	Crossing			
Length	Decimal (m, mm, etc.)	Crossing	✓		
Altitude	Decimal (m)	Crossing			
Traction System installed	Enumeration (none, AC 25kV 50Hz, ...)	Crossing			
Type of rail	Enumeration (56 E1, 60 E1, ...)	Crossing			
Type of ballast	Enumeration (ballast-less, with ballast)	Crossing			
Type of sleepers	Enumeration (steel, wooden, ...)	Crossing			
Nominal Max Speed (different for each direction)	Decimal (km/h)	Crossing	✓		
Joints Type	Enumeration (welded, insulated, glued, etc.)	Crossing	✓		
Joints Nominal Displacement	Decimal (mm)	Crossing	✓		

Figure 3.8: Static data table for Crossings

DYNAMIC Data				**TMS Critical Variables**		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Environmental Humidity	Decimal (%)	CROSSING (environment)	External			
Environmental Temperature	Decimal (°C)	Track (environment)	External	✓		
Environmental Pressure	Decimal (bars)	CROSSING (environment)	External			
Wind Speed	Decimal (m/s)	CROSSING (environment)	External			
Wind Direction	Decimal	CROSSING	External			

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
	(degrees)	(environment)				
Ballast status	Radar – Image	CROSSING (under-structure)	Diagnostic (Ground penetrating radar, etc.)			
Ballast inclination	Decimal (degrees)	CROSSING (under-structure)	Asset-related (inclinometer)			
Ballast temperature	Decimal (°C)	CROSSING (under-structure)	Asset-related			
Flood alert - water level	Decimal (mm)	CROSSING (under-structure)	External	✓		
Flood alert – digital	Binary	CROSSING (under-structure)	Asset-related (Track circuits)			
Snow/Ice detection	Binary	CROSSING (environment) Actuation devices	External	✓		
Total axle passages (related to each branch) per direction	Integer	CROSSING	Asset-related (Axle counter)			
Total weight transited (related to each branch) per direction	Decimal (kg)	CROSSING	Asset-related (Weight in motion)	✓		
Load per axle (each axle passed or aggregated measure, related to each branch) per direction	Decimal (kg)	CROSSING	Asset-related (Weight in motion)			
Total train passages (related to each branch) per direction	Integer	CROSSING	Asset-related (Traffic Management System)	✓		
Speed of passed trains (related to each branch) per direction	Decimal (km/h)	CROSSING	Asset-related (train odometry)	✓		
Wheel weight transited (related to each rail) per direction	Decimal (kg)	CROSSING	Asset-related (Weight in motion)			

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Gauge	Decimal (m, mm, etc.)	CROSSING	Diagnostic	✓		
Rail profile (height, width, etc.)	Decimal (mm)	CROSSING	Diagnostic	✓		
Rail profile	Image	CROSSING	Diagnostic	✓		
Joint status	Enumeration (OK, KO, creeped, etc.)	CROSSING	Diagnostic	✓		
Joint Displacement	Decimal (mm)	CROSSING	Diagnostic	✓		
Joint visual status	Image	CROSSING	Diagnostic			
Vibration / Accelerations	Array of Decimals (G)	CROSSING (rail status, crack detection, etc.)	Diagnostic/ Asset-related (triggered)	✓		
Sounds	Array of Decimals (dB)	CROSSING (rail status, crack detection, etc.)	Diagnostic / Asset-related (triggered)			
Friction	Decimal (N)	CROSSING	Diagnostic (tribometer)			
Number of maintenance interventions	Integer	CROSSING	Maintenance			
Scheduled maintenance interventions frequency	Time (each "X" days, months, etc.)	CROSSING	Maintenance	✓		
Maintenance interventions date	Time	CROSSING	Maintenance	✓		
Maintenance intervention start time	Time	CROSSING	Maintenance	✓		
Maintenance intervention end time	Time	CROSSING	Maintenance	✓		
Maintenance intervention type code	String (code)	CROSSING	Maintenance	✓		
Maintenance intervention description	String	CROSSING	Maintenance (could be supplied by humans)	✓		
Code for failure that determined a maintenance	String (code)	CROSSING	Maintenance	✓		

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
intervention						
Maintenance team that made the intervention	String (code)	CROSSING	Maintenance			

Figure 3.9: Dynamic data table for Crossings

3.6. Track (Rail)

This section includes the representation of the track based on its classification among all the physical railway assets, on the identification of its subcomponents and on the relevant data that should be collected to identify the functional status of the asset.

3.6.1. Asset classification

Track is part of the infrastructure of a railway network, and it is the structure on which the train runs. Figure 3.10 shows the complete classification of the asset with respect to all the physical railway assets.

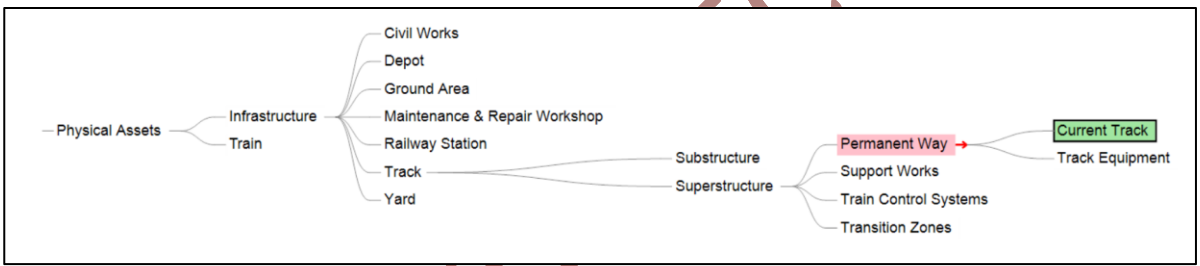


Figure 3.10: Position of Track into the railway taxonomy

3.6.2. Asset sub-components

Track is made by many sub-components, such as rails, fasteners, sleepers and ballast.

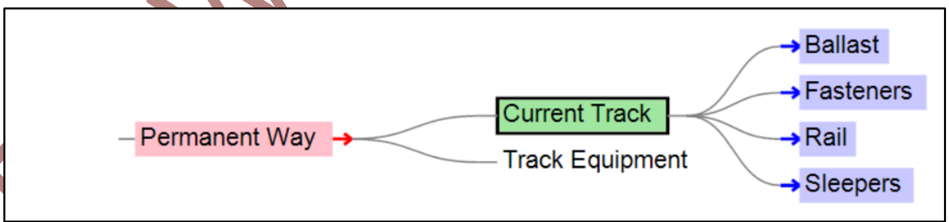


Figure 3.11: Track sub-components classification

The following list recalls also in textual form the most relevant components of a track:

- Ballast;
- Rail;
- Sleepers;
- Fasteners.

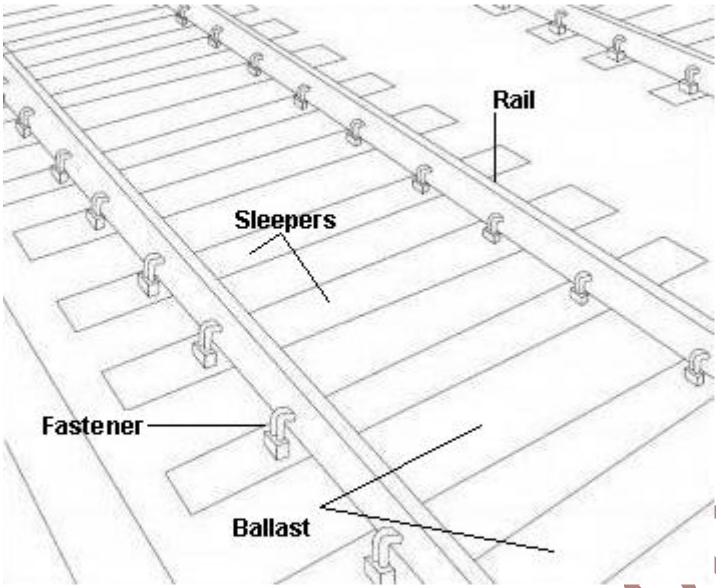


Figure 3.12: Technical picture of a piece of track

3.6.3. Data Tables

All the data mentioned in the following tables (or, at least, most of it) has to be stored and collected regularly in order to create an historical database containing all the possible information related to the functional behaviour of the monitored devices over their entire lifetime.

Please note that, in the track case, all the variables (either static or dynamic) must be specified in relationship with a spatial/geographical value. Indeed, referring to a generic track is ambiguous without this information, since track length can span from 20 m to 200 km.

STATIC Data			TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Type	Enumeration (straight line, curved track, ...)	Track			
Length	Decimal (m, mm, etc.)	Track	✓		
Absolute Position	Geospatial	Track	✓	✓	
Mileage	Decimal (km)	Track	✓		
Altitude	Decimal (m)	Track			
Track ID	Integer/String	Track	✓		
Track elements	Enumeration (switches, level crossings, bridges, tunnels)	Track	✓		
Traction System installed	Enumeration (none, AC 25kV 50Hz, ...)	Track			
Type of rail	Enumeration (56 E1, 60 E1, ...)	Track (under-structure)			
Rail material	String	Track – Rail			
Type of ballast	Enumeration (ballast-less, with ballast)	Track (under-structure)			
Type of sleepers	Enumeration (steel, wooden, ...)	Track (sleepers)			
Type of	Enumeration (k-type, Pandrol,	Track (fasteners)			

STATIC Data			TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
fasteners	...)				
Nominal Max Speed	Decimal (km/h)	Track	✓		
Construction date	Time	Track			
Construction series	string	Track			
Original Test Date	Time	Track			
Installation Date	Time	Track			

Table 3.3: Track: static data

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Current type	Enumeration (straight line, curved track, ...)	Track	Internal			
Environmental Temperature	Decimal (°C)	Track (environment)	External	✓	✓	✓
Environmental Humidity	Decimal (%)	Track (environment)	External			
Environmental Pressure	Decimal (bars)	Track (environment)	External			
Ballast status	Radar – Image	Track (under-structure)	Diagnostic (Ground penetrating radar, etc.)			
Ballast inclination	Decimal (degrees)	Track (under-structure)	Asset-related (inclinometer)			
Ballast temperature	Decimal (°C)	Track (under-structure)	Asset-related			
Flood alert – water level	Decimal (mm)	Track (under-structure)	External	✓	✓	✓
Flood alert – digital	Binary	Track (under-structure)	Asset-related (Track circuits)			
Dust, water, wind and snow detection	Binary/Image	Track (environment)	External	✓	✓	✓
Total axle passages per direction	Integer	Track - Rail	Asset-related (Axle counter)			
Total weight transited per	Decimal (kg)	Track - Rail	Asset-related (Weight in			

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
direction			motion)			
Load per axle per direction	Decimal (kg)	Track - Rail	Asset-related (Weight in motion)			
Total train passages per direction	Integer	Track – Rail	Asset-related (Traffic Management System)	✓		
Quantity of freight traffic per direction	Integer	Track – Rail	Asset-related (Traffic Management System)	✓		
Speed of passed trains per direction	Decimal (km/h)	Track – Rail	Asset-related (train odometry)	✓		
Acceleration	Decimal (m/s^2)	Track – Rail	Asset-related (train odometry)	✓		
Wheel weight transited per direction	Decimal (kg)	Track – Rail	Asset-related (Weight in motion)			
Gauge	Decimal (mm)	Track – Rail	Diagnostic	✓	✓	✓
Alignment	Decimal (m)	Track – Rail	Diagnostic	✓	✓	✓
Cross level	Decimal (mm)	Track – Rail	Diagnostic	✓	✓	✓
Longitudinal level	Decimal (m or mm)	Track – Rail	Diagnostic	✓	✓	✓
Twist	Decimal (% or mm/m)	Track – Rail	Diagnostic	✓	✓	✓
Cant deficiency	Decimal (mm)	Track – Rail	Diagnostic	✓	✓	✓
Cant gradient	Decimal (% or mm/m)	Track – Rail	Diagnostic	✓	✓	✓
Horizontal curvature	Decimal (1/m)	Track – Rail	Diagnostic	✓	✓	✓
Vertical curvature	Decimal (1/m)	Track – Rail	Diagnostic	✓	✓	✓
Gradient	Decimal (% or mm/m)	Track – Rail	Diagnostic	✓	✓	✓
Rail profile (height, width, Curvature, etc.)	Decimal (mm)	Track – Rail	Diagnostic	✓		
Rail profile	Image	Track – Rail	Diagnostic			
Longitudinal Profile (corrugation analysis)	Decimal (mm)	Track – Rail	Diagnostic	✓		

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Vibration / Accelerations	Array of Decimals (G)	Track - Rail (rail status, crack detection, etc.)	Diagnostic/ Asset-related (triggered)	✓	✓	✓
Sounds	Array of Decimals (dB)	Track - Rail (rail status, crack detection, etc.)	Diagnostic / Asset-related (triggered)			
Friction	Decimal (N)	Track - Rail	Diagnostic (tribometer)	✓	✓	✓
Sleeper status	Enumeration (OK, KO, creeped, etc.)	Sleeper	Diagnostic	✓		
Sleeper Mileage[9]	Decimal (mm)	Sleeper	Diagnostic			
Sleeper visual status	Image	Sleeper	Diagnostic			
Fastener status	Enumeration (OK, KO, creeped, etc.)	Fastener	Diagnostic	✓	✓	✓
Fastener Mileage[10]	Decimal (mm)	Fastener	Diagnostic			
Fastener visual status	Image	Fastener	Diagnostic			
Number of maintenance interventions	Integer	Track	Maintenance			
Scheduled maintenance interventions frequency	Time (each "X" days, months, etc.)	Track	Maintenance			
Maintenance interventions date	Time	Track	Maintenance	✓	✓	✓
Maintenance intervention start time	Time	Track	Maintenance	✓	✓	✓
Maintenance intervention end time	Time	Track	Maintenance	✓	✓	✓
Maintenance intervention type code	String (code)	Track	Maintenance			
Maintenance intervention description	String	Track	Maintenance (could be supplied by	✓	✓	✓

[9] Position of the sleeper on the railway line, coded with mileage.
[10] Position of the fastener on the railway line, coded with mileage.

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
			humans)			
Code for failure that determined a maintenance intervention	String (code)	Track	Maintenance	✓		
Maintenance team that made the intervention	String (code)	Track	Maintenance		✓	
Electromagnetic environmental effects	To be defined	Track	Diagnostic			
Scheduled measure frequency	Time (each "X" days, months, etc.)	Track	Diagnostic			
Measure date	Time	Track	Diagnostic			
Measure start time	Time	Track	Diagnostic			
Measure end time	Time	Track	Diagnostic			
Measure duration	Time	Track	Diagnostic			
Measure type code	String (code)	Track	Diagnostic			
Measure description	String	Track	Diagnostic (could be supplied by humans)			
Train direction during measuring	Enumeration	Track	Diagnostic / Asset-related (train odometry)			

Table 3.4: Track: dynamic data

3.7. Catenary

This section includes the representation of the Catenary based on its classification among all the physical railway assets, on the identification of its subcomponents and on the relevant data that should be collected to identify the functional status of the asset.

3.7.1. Asset classification

Catenary (also known as Overhead line) is part of the infrastructure of a railway network, and it is the structure, which provides power supply to electric train. Figure 3.13 shows the complete classification of the asset with respect to all the physical railway assets.

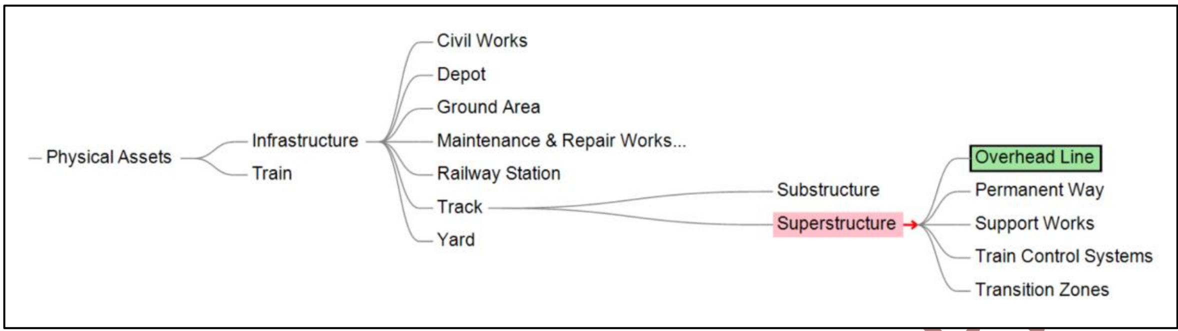


Figure 3.13: Position of Catenary into the railway taxonomy

3.7.2. Asset sub-components

An Overhead Line is composed of many sub-components, which are depicted in Figure 3.14 and Figure 3.15. The first picture shows the classification of the asset sub-components and their categorization in three different classes, depending on the role that they assume in the asset specific operation.

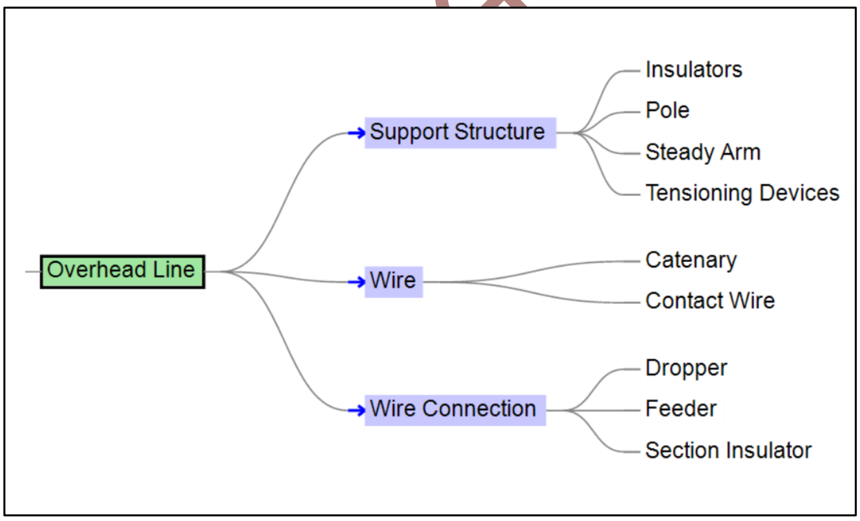


Figure 3.14: Catenary sub-components classification

Figure 3.15 shows a technical representation of some of the Catenary main elements.

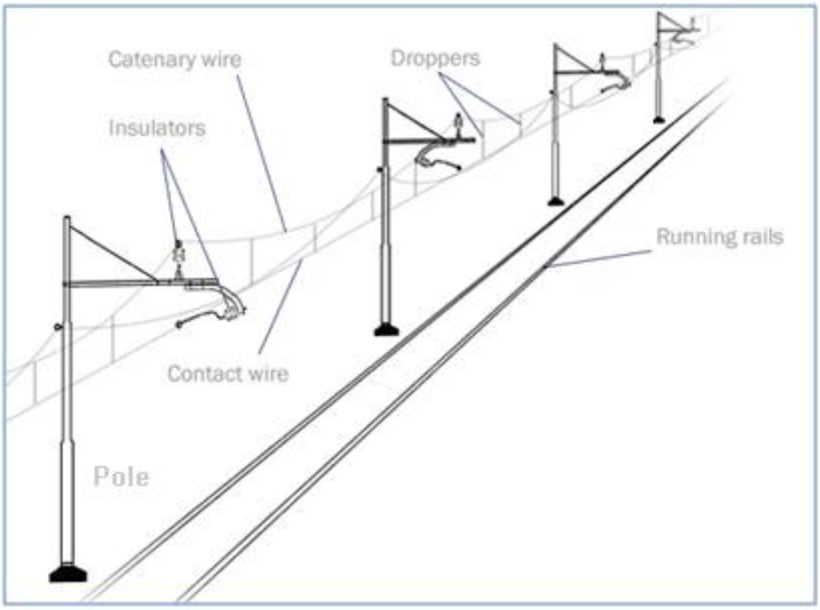


Figure 3.15: Technical picture of a Catenary

The following list recalls in textual form all the most relevant components of an Overhead Line, divided by the functions they implement or grouped by super-component:

- Wires:
  - Catenary,
  - Contact Wire;
- Wire Connections:
  - Dropper,
  - Feeder,
  - Section Insulator;
- Support Structure:
  - Steady arms,
  - Pole,
  - Insulator,
  - Tensioning devices.

3.7.3. Data Tables

All the data mentioned in the following tables (or, at least, most of it) has to be stored and collected regularly in order to create an historical database containing all the possible information related to the functional behaviour of the monitored devices over their entire lifetime.

STATIC Data			**TMS Critical Variables**		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Type	Enumeration (simple catenary, stitched catenary, compound catenary, ...)	Catenary	✓		
Model	String	Catenary			
Absolute Position	Geospatial	Catenary	✓		

STATIC Data			**TMS Critical Variables**		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Mileage	Decimal (km)	Catenary	✓		
Type of Wire	Enumeration (Circular 80 mm^2; 85 mm^2, 100 mm^2; 107 mm^2; 120 mm^2; 150 mm^2 …)	Catenary - Contact wire	✓		
Nominal Wire Section	Decimal (mm)	Catenary - Contact wire	✓		
Number of contact wire	Integer	Catenary - Contact wire			
Wire Length	Decimal (m, mm, etc.)	Catenary - Contact wire			
Nominal Voltage	Decimal (V)	Catenary - Contact wire	✓		
Nominal Max Speed	Decimal (km/h)	Catenary	✓		
Height lower limit	Decimal (m)	Catenary - Contact wire	✓		
Height upper limit	Decimal (m)	Catenary - Contact wire	✓		
Stagger limit straight line	Decimal (mm)	Catenary - Contact wire	✓		
Stagger limit curve	Decimal (mm)	Catenary - Contact wire	✓		
Wear limit	Decimal (mm)	Catenary - Contact wire	✓		
Max allowed slope	Decimal(mm/m)	Catenary	✓		
Steady arm type	Enumeration	Steady arm			
Steady arm nominal vertical inclination	Decimal (degrees)	Steady arm			
Dropper Type	Enumeration (Current carrying dropper, common dropper, …)	Dropper			
Dropper Nominal Distance	Decimal (mm)	Dropper			
Pole Nominal Distance	Decimal (m)	Pole			
Insulator type by function	Enumeration (section insulator, bracket insulator…)	Insulator			
Construction date	Time	Catenary			
Construction series	string	Catenary			
Original Test Date	Time	Catenary			
Installation Date	Time	Catenary			

Table 3.5: Catenary: static data

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Environmental Temperature	Decimal (°C)	Catenary - Contact wire (environment)	External	✓	✓	✓
Environmental Humidity	Decimal (%)	Catenary - Contact wire (environment)	External			
Environmental Pressure	Decimal (bars)	Catenary - Contact wire (environment)	External			
Wind Speed	Decimal (m/s)	Catenary - Contact wire (environment)	External	✓	✓	✓
Wind Direction	Decimal (degrees)	Catenary - Contact wire (environment)	External	✓	✓	✓
Snow/Ice detection	Binary	Catenary - Contact wire (environment)	External	✓	✓	✓
Height	Decimal (m)	Catenary - Contact wire	Diagnostic	✓		✓
Stagger	Decimal (mm)	Catenary - Contact wire	Diagnostic	✓		✓
Wear	Decimal (mm)	Contact wire	Diagnostic	✓		✓
Slope	Decimal (mm/m)	Catenary	Diagnostic	✓		✓
Pole mileage	Decimal (m)	Pole	Diagnostic			
Distance between Poles	Decimal (m)	Pole	Diagnostic			
Steady arm vertical inclination	Decimal (degrees)	Steady arm	Diagnostic			
Steady arm horizontal inclination	Decimal (degrees)	Steady arm	Diagnostic			
Dropper Mileage	Decimal (m)	Dropper	Diagnostic			
Broken Dropper	Binary	Dropper	Diagnostic	✓	✓	✓
Not tensioned dropper	Binary	Dropper	Diagnostic	✓	✓	✓
Insulator mileage	Decimal (m)	Insulator	Diagnostic			
Broken Insulator	Binary	Insulator	Diagnostic	✓	✓	✓
Section Insulator mileage	Decimal (m)	Section Insulator	Diagnostic			
Broken Section Insulator	Binary	Section Insulator	Diagnostic	✓		✓

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Joint Mileage	Decimal (m)	Contact Wire	Diagnostic			
Feeder Mileage	Decimal (m)	Feeder	Diagnostic			
Contact Wire Voltage	Decimal (V)	Contact Wire	Diagnostic	✓		
Contact Wire Uplift	Decimal (mm)	Contact Wire	Diagnostic			
Fixed Point Mileage	Decimal (m)	Tensioning devices	Diagnostic			
Fixed Point Asymmetric Load	Decimal (N)	Tensioning devices	Diagnostic			
Contact Force	Decimal (N)	Contact Wire	Diagnostic	✓	✓	✓
Insufficient stagger	Binary	Catenary - Contact wire	Diagnostic	✓		✓
Overlap Mileage	Decimal (mm)	Contact wire	Diagnostic			
Total train passages per direction	Integer	Catenary	Asset-related (Traffic Management System)			
Speed of passed trains per direction	Decimal (km/h)	Catenary	Asset-related (train odometry)			
Vibration / Accelerations (interaction with pantograph)	Array of Decimals (G)	Contact Wire	Diagnostic/ Asset-related (triggered)	✓		
Sounds (interaction with pantograph)	Array of Decimals (dB)	Contact Wire	Diagnostic / Asset-related (triggered)			
Friction (interaction with pantograph)	Decimal (N)	Contact Wire	Diagnostic (tribometer)			
Number of maintenance interventions	Integer	Catenary	Maintenance			
Scheduled maintenance interventions frequency	Time (each "X" days, months, etc.)	Catenary	Maintenance	✓		
Maintenance interventions date	Time	Catenary	Maintenance	✓	✓	✓
Maintenance intervention start time	Time	Catenary	Maintenance	✓	✓	✓

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Maintenance intervention end time	Time	Catenary	Maintenance	✓	✓	✓
Maintenance intervention type code	String (code)	Catenary	Maintenance	✓		
Maintenance intervention description	String	Catenary	Maintenance (could be supplied by humans)	✓		✓
Code for failure that determined a maintenance intervention	String (code)	Catenary	Maintenance	✓		
Maintenance team that made the intervention	String (code)	Catenary	Maintenance			
Moving parts bearings status (lubrication, wear, rust, etc.)	N.D. (non-invasive techniques to be studied in In2Rail)	Tensioning devices	N.D. (non-invasive techniques to be studied in In2Rail)	✓		
Measure date	Time	Catenary	Diagnostic			
Measure start time	Time	Catenary	Diagnostic			
Measure end time	Time	Catenary	Diagnostic			
Measure duration	Time	Catenary	Diagnostic			
Measure type code	String (code)	Catenary	Diagnostic			
Measure description	String	Catenary	Diagnostic (could be supplied by humans)			
Train direction during measuring	Enumeration	Catenary	Diagnostic / Asset-related (train odometry)			

Table 3.6: Catenary: dynamic data

3.8. Bridge

This section includes the representation of bridge based on its classification among all the physical railway assets, on the identification of its subcomponents and on the relevant data that should be collected to identify the functional status of this asset.

3.8.1. Asset classification

Bridges are infrastructural elements of the railway world that allow overcoming limited discontinuities of a railway line, usually represented by rivers and similar natural obstacles. They are classified as part of the Civil Works together with Embankments and Tunnels. Figure 3.16 shows the complete classification of the asset with respect to all the physical railway assets.

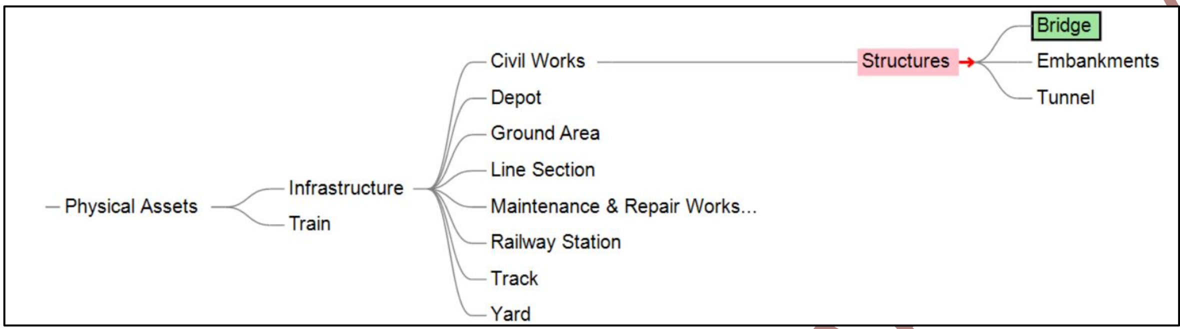


Figure 3.16: Bridge inside railway taxonomy

3.8.2. Asset sub-components

Bridges are composed of many sub-components, which are depicted in Figure 3.17, Figure 3.18 and Figure 3.19. The first picture shows the classification of the asset sub-components and their categorisation in four different classes, while the other two figures show technical diagrams with labels identifying parts.

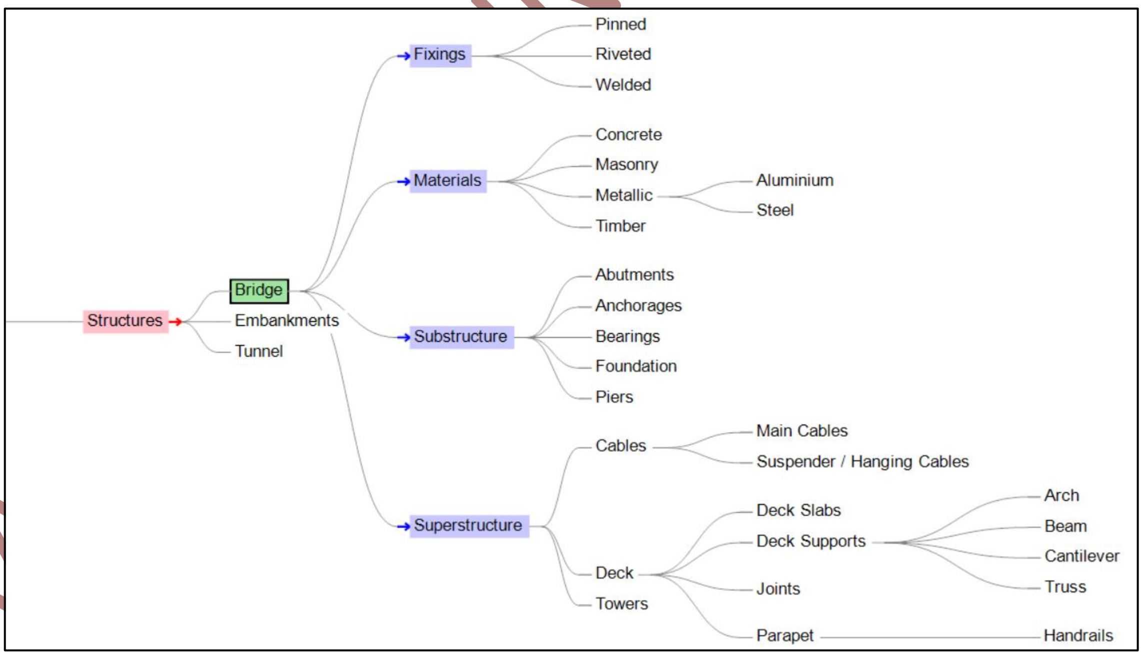


Figure 3.17: Bridge sub-components classification

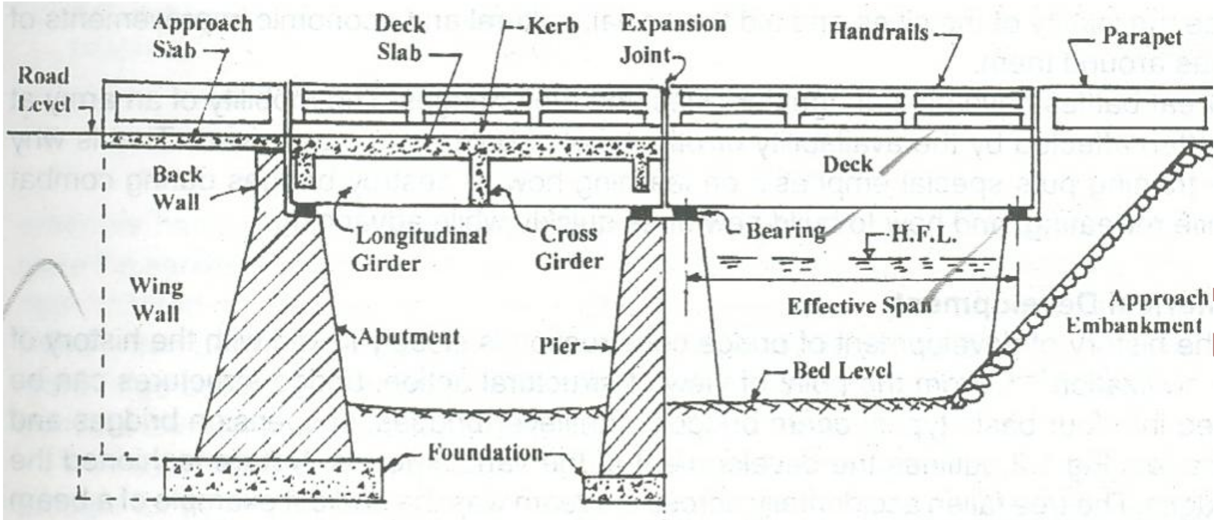


Figure 3.18: Components of a conventional bridge[11]

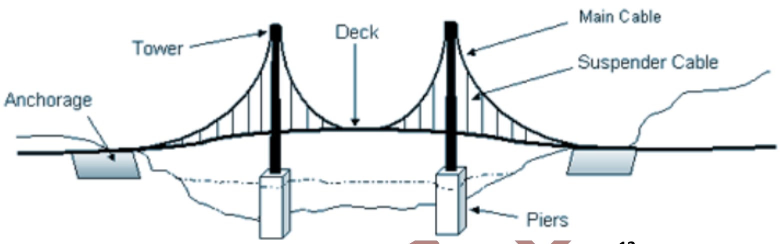


Figure 3.19: Suspension bridge components[12]

3.8.3. Data Tables

All the data mentioned in the following tables (or, at least, most of it) should be stored and collected regularly to create an historical database containing all the possible information related to the functional behaviour of the monitored devices over their entire lifetime.

The content of data tables are partially based on parameters described in [11], [12], [13], [14], [15], [16].

STATIC Data			TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Name	String	Bridge			
Construction date	Date	Bridge			
Design life	Integer (years)	Bridge	✓		
Span length	1 per span, Decimal (m)	Deck slab, pier, tower	✓		
Number of spans	Integer	Deck slab, pier, tower			
Column height	Decimal (m)	Pier, tower			
Total length (inc. approaches)	Decimal (m)	Deck			
Number of decks	Integer	Deck			
Upper design	Integer (Centigrade)	Bridge			

[11] Source: http://www.photonesta.com/bridge-components-parts.html
[12] Source: http://www.wsdot.wa.gov/TNBhistory/Machine/machine1.htm

STATIC Data			TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
temperature					
Lower design temperature	Integer (Centigrade)	Bridge			
Self-weight (dead load)	Decimal (Tons)	Bridge			
Max design traffic load (live load)	Decimal (kN/m^2)	Any load-bearing structure e.g. truss, beam, deck	✓	✓	
Max axle load	Decimal (kN/m^2)	Deck	✓		
Max seismic load	Decimal (kN/m^2)	Bridge	✓		
Max wind load	Decimal (kN/m^2)	Bridge	✓		
Max snow load	Decimal (kN/m^2)	Bridge	✓		
Max tension	Decimal	Main cable	✓		

Table 3.7: Bridge: static data

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Delamination of surfaces	Image	Deck	Thermography			
In-service loading (traffic)	Decimal (tonnes)	Deck, beam, pier etc.	Weigh in motion system	✓		
Rotational movements in support structures	Decimal (inclination, degrees)	Piers, abutments, towers, bearings	Inclinometers	✓	✓	✓
Strain / stress	Decimal (longitudinal strain)	Any load-bearing structure e.g. truss, beam, deck	Strain gauges, fibre optic sensors (Bragg sensors etc.)	✓	✓	✓
Response to vibration	Decimal (m/s^2)	Deck, piers, cables, truss	Accelerometers	✓	✓	
Crack detection (metal structures)	Decimal (time of flight, microseconds)	Beam, truss, deck, cables	Ultrasonic or electromagnetic testing	✓	✓	✓
Ongoing monitoring of existing crack	Decimal (mm)	Any load-bearing structure e.g. tower, pier, deck	Potentiometer or linear variable differential transformer (LVDT)	✓	✓	✓
Bridge scour	River bed profile	Pier, foundation	Sonar or ground penetrating radar	✓		

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
			measurement, visual inspection by diver			
Wind speed	Decimal (m/s)	Bridge	Anemometer, often mounted on tower or associated catenary structure	✓	✓	✓

Table 3.8: Bridge: dynamic data

3.9. Tunnel

This section includes the representation of tunnel based on its classification among all the physical railway assets, on the identification of its subcomponents and on the relevant data that should be collected to identify the functional status of the asset.

3.9.1. Asset classification

Tunnels are infrastructure elements that allow a railway line to be continuous by passing through mountains, underground or underwater. They are classified as part of the Civil Works together with Embankments and Bridges. Figure 3.20 shows the complete classification of the asset with respect to all the physical railway assets.

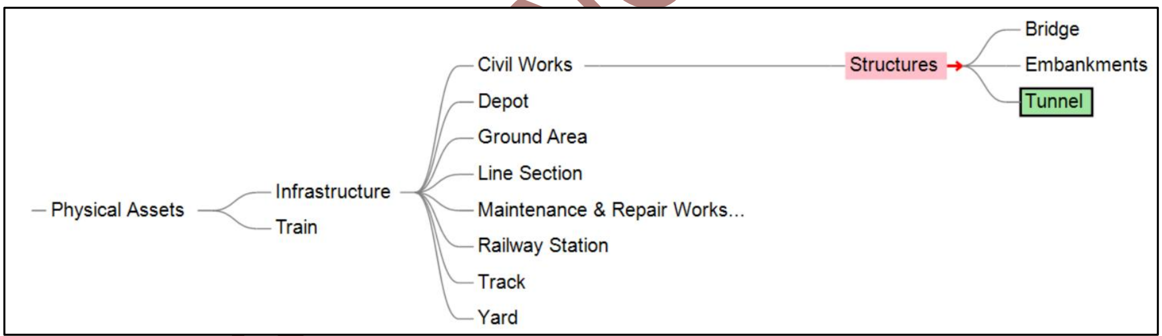


Figure 3.20: Tunnel within railway taxonomy

3.9.2. Asset sub-components

Tunnels are composed of many sub-components, which are depicted in Figure 3.17. This picture shows the classification of the asset sub-components and their categorisation in several different classes.

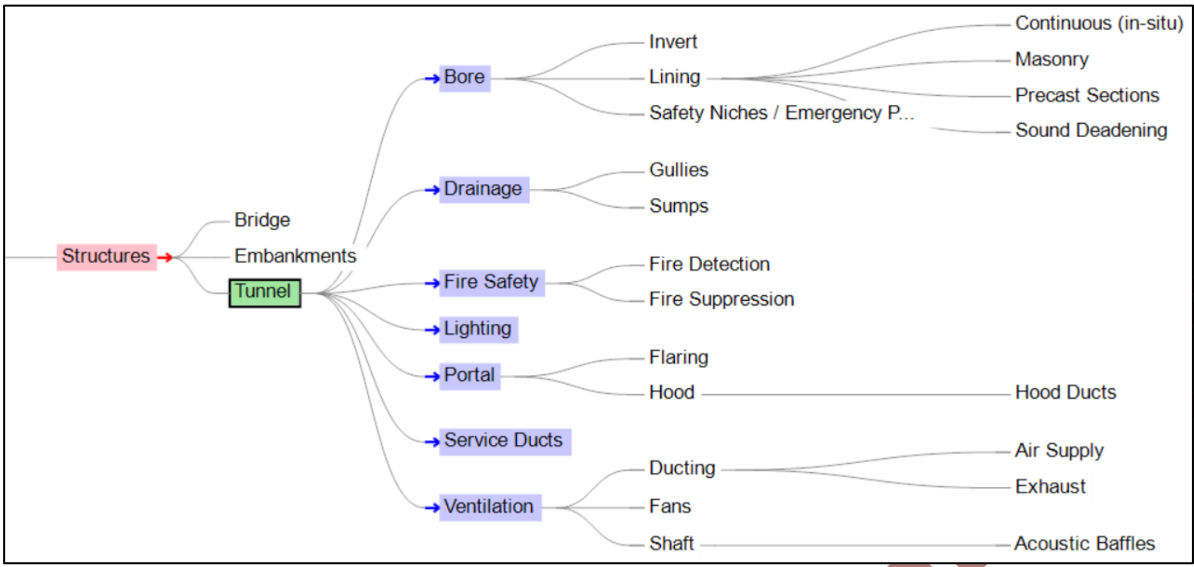


Figure 3.21: Tunnel sub-components classification

3.9.3. Data Tables

All the data mentioned in the following tables (or, at least, most of it) should be stored and collected regularly to create an historical database containing all the possible information related to the functional behaviour of the monitored devices over their entire lifetime.

Regarding tunnels, dynamic data is used to inform maintenance or monitor degradation of asset over extended timeframes. In particular, changes to tunnel profile (appearance of bulges etc.) can affect traffic management through changes to max loading gauge and imposition of speed restrictions. Similar considerations can be drawn for changes to lining condition and consequent need to schedule maintenance.

The content of data tables is partially based on parameters described in [17].

STATIC Data			**TMS Critical Variables**		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Name	String	Tunnel			
Construction date	Date	Tunnel			
Design life	Integer (years)	Tunnel			
Number of bores	Integer	Bore			
Total length	Decimal (m)	Bore	✓		
Portal 1 distance	Decimal (mileage)	Portal	✓	✓	
Portal 1 position	Geoposition	Portal	✓	✓	
Portal 2 distance	Decimal (mileage)	Portal	✓	✓	
Portal 2 position	Geoposition	Portal	✓	✓	
Radius of bore	Decimal (m)	Bore			
Max loading gauge	Text (ERA TSI profile code)	Bore	✓		
Width	Decimal (m)	Bore			
Height	Decimal (m)	Bore, Invert			
Effective height (railhead / invert	Decimal (m)	Bore, Invert, OLE	✓		

STATIC Data			TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
to OLE structures)					
Min Curve Radius	Decimal (m)	Bore			
Reference profile	3D contour	Bore, lining			

Table 3.9: Tunnel: static data

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Profile	3D contour	Bore, lining	Commonly derived from laser-based inspection (point cloud)	✓	✓	
Lining condition (visual)	Video or manual inspection report, may include enumeration of faults (cracking etc.)	Lining	Inspection vehicle or track workers	✓	✓	✓
Lining condition (survey)	Enumeration of condition (good, average, poor) or similar	Lining	Ground penetrating radar or thermography (in particular for crack detection and maintenance issues with pointing/mortar)	✓		
Ongoing monitoring of existing crack	Decimal (mm)	Lining	Potentiometer or linear variable differential transformer (LVDT)			
Lining thickness	Decimal (m)	Lining	Ground penetrating radar			
Ballast depth	Decimal (m)	Invert	Ground penetrating radar or manual pit			
Void / water intrusion detection	Decimal (m) – depth of feature	Bore, lining	Ground penetrating radar or thermography	✓	✓	✓
Air Pressure	Pascal	Tunnel	Air Pressure sensor	✓		
Status of power fans				✓		
Condition of fittings				✓		
Condition of fixtures				✓		
Fire/explosion risk			Hot box detectors, thermal cameras	✓		

Table 3.10: Tunnel: dynamic data

3.10. Embankments

This section includes the representation of embankment based on its classification among all the physical railway assets, on the identification of its subcomponents and on the relevant data that should be collected to identify the functional status of the asset.

3.10.1. Asset classification

Embankments are infrastructural elements classified as part of the Civil Works together with Bridges and Tunnels. Figure 3.22 shows the complete classification of the asset with respect to all the physical railway assets.

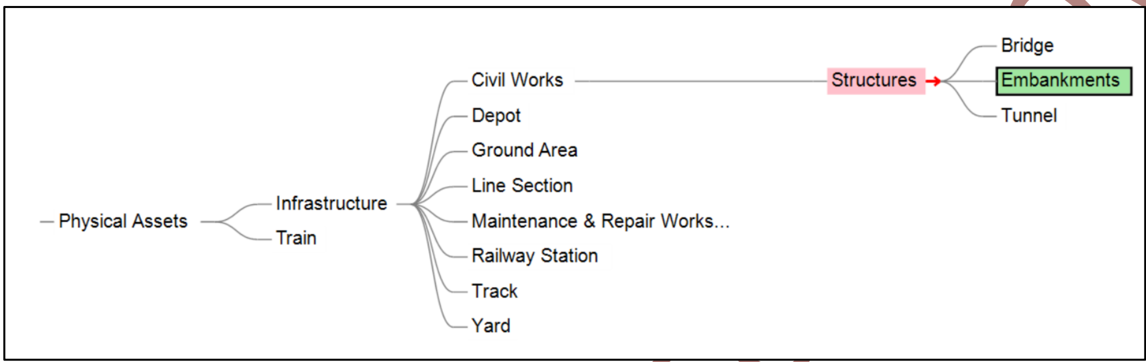


Figure 3.22: Embankments within railway taxonomy

3.10.2. Asset sub-components

Embankments are composed of many sub-components, which are depicted in Figure 3.23. This picture shows the classification of the asset sub-components and their categorisation in several different classes.

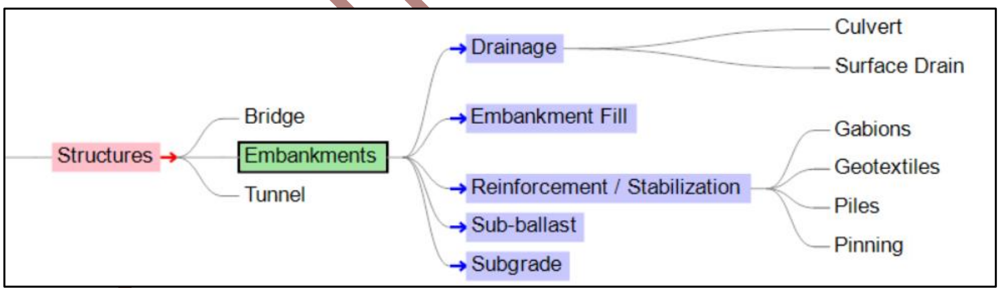


Figure 3.23: Embankments sub-components classification

3.10.3. Data Tables

All the data mentioned in the following tables (or, at least, most of it) should be stored and collected regularly to create an historical database containing all the possible information related to the functional behaviour of the monitored devices over their entire lifetime.

Excessive moisture levels / drying of embankment fill may lead to changes in track geometry (buckling) or in extreme cases slippage. Monitoring of moisture levels can inform on likely swell / shrinkage and allows extra drainage or reinforcement to be installed if needed. Temporary or permanent speed restrictions may need to be applied in cases of extreme change in moisture levels.

The content of data tables are partially based on parameters described in [18], [19], [20].

STATIC Data			TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Slope angle	Decimal (degrees)	Embankment			
Construction date	Date	Embankment			
Design life	Integer (years)	Embankment			
Start distance	Decimal (mileage)	Embankment			
Start position	Geoposition	Embankment	✓		
End distance	Decimal (mileage)	Embankment	✓		
End position	Geoposition	Embankment	✓		
Culvert location	Decimal (mileage)	Culvert	✓		
Culvert diameter	Decimal (m)	Culvert			
Vertical height above feature	Decimal (m)	Embankment			
Length	Decimal (m)	Embankment			
Max live loading	Decimal (kN/m^2)	Embankment	✓		
Fill construction	Enumeration array (sand, gravel, clay, shale etc.)	Embankment fill			
Material reference resistivity	Array, decimal (ohm meter)	Embankment fill (1 per material, as fill construction)			

Table 3.11: Embankment: static data

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Resistivity	Array, decimal (ohm meter) – can be compiled to image	Embankment fill	Installed sensors or periodic survey	✓		
Moisture content	Decimal (%)	Embankment fill	Installed sensors or periodic survey, can be derived from resistivity	✓		
Fill temperature	Decimal (c)	Embankment fill	Installed sensors or periodic survey	✓		
Pore water pressure	Integer (kPa)	Embankment fill	Installed sensors if available	✓		
Vegetation cover	Video or manual inspection record	Embankment	On-train video or track inspection report	✓		

Table 3.12: Embankment: dynamic data

3.11. Line section

This section includes the representation of line section based on its classification among all the physical railway assets, on the identification of its subcomponents and on the relevant data that should be collected to identify the functional status of the asset.

3.11.1. Asset classification

The Line Section is a logical infrastructure element that identifies the characteristics of a certain limited part of a railway line. It is classified as one of the main infrastructure elements, as depicted in Figure 3.24.

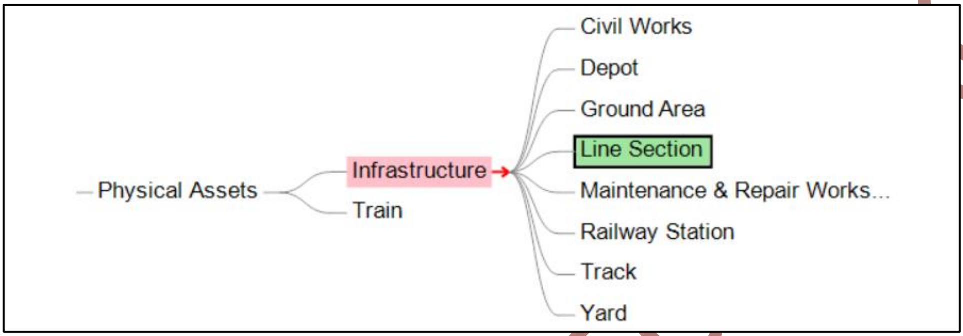


Figure 3.24: Line Section within railway taxonomy

3.11.2. Asset sub-components

Line Sections are composed of many sub-components, which are depicted in Figure 3.25. This picture shows the classification of the asset sub-components and their categorisation in several different classes.

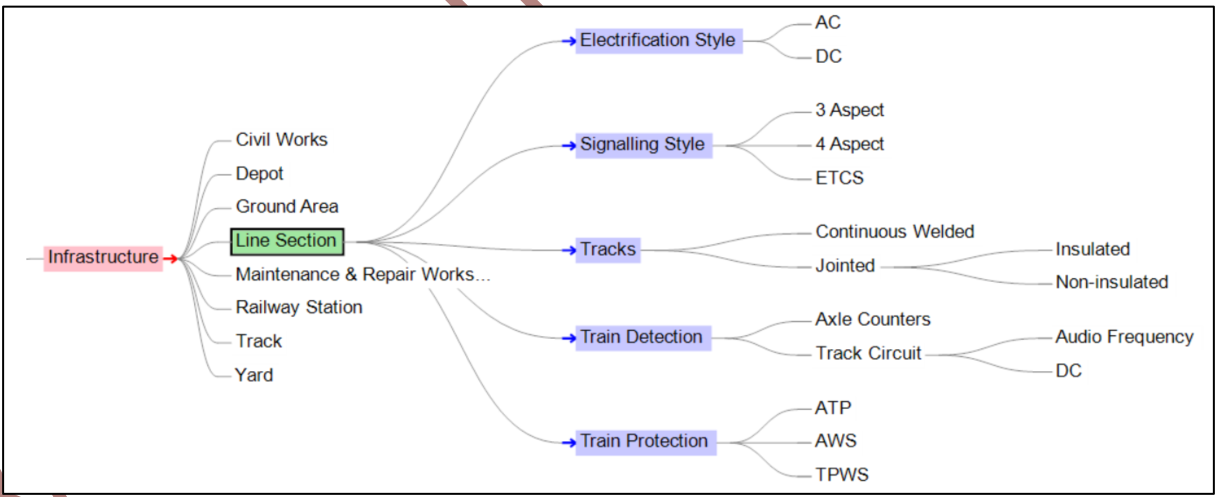


Figure 3.25: Line Section sub-components classification

3.11.3. Data Tables

All the data mentioned in the following tables (or, at least, most of it) should be stored and collected regularly to create an historical database containing all the possible information related to the functional behaviour of the monitored devices over their entire lifetime.

These data tables do not include track-related dynamic data since it is assumed that they are included under the representation of the “track” asset.

STATIC Data			**TMS Critical Variables**		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Length	Decimal (km)	Line section			
Installation date	Date	Tracks			
Design life	Integer (years)	Tracks			
Number of tracks	Integer	Tracks			
Track directions	Array, enumeration (up, down)	Tracks			
Track IDs	String	Tracks			
Loading gauge	String	Tracks			
Start distance	Decimal (mileage)	Tracks	✓	✓	
Start position	Geoposition	Tracks	✓	✓	
End distance	Decimal (mileage)	Tracks	✓	✓	
End position	Geoposition	Tracks	✓	✓	
Installed signalling	Enumeration (See signalling style)	Signalling style	✓		
Installed train detection	Enumeration (See train detection)	Train detection	✓		
Installed electrification	Enumeration (see electrification style)	Electrification style	✓		

Table 3.13: Line section: static data

DYNAMIC Data				**TMS Critical Variables**		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Vegetation coverage	Decimal (%)	Line section	Laser or video survey, manual inspection	✓	✓	✓
Obstruction type	String (obstruction name)	Tracks	Driver report, CCTV (if installed)	✓	✓	✓
Obstruction location	Geoposition	Tracks	Driver report, CCTV (if installed)	✓	✓	✓

Table 3.14: Line section: dynamic data

3.12. Level crossing

This section includes the representation of the level crossing based on its classification among all the physical railway assets, on the identification of its subcomponents and on the relevant data that should be collected to identify the functional status of the asset.

3.12.1. Asset classification

Level Crossings are part of the infrastructure of a railway network, and they can be classified as specialized Track Equipment. Figure 3.26 shows the complete classification of the asset with respect to all the physical railway assets.

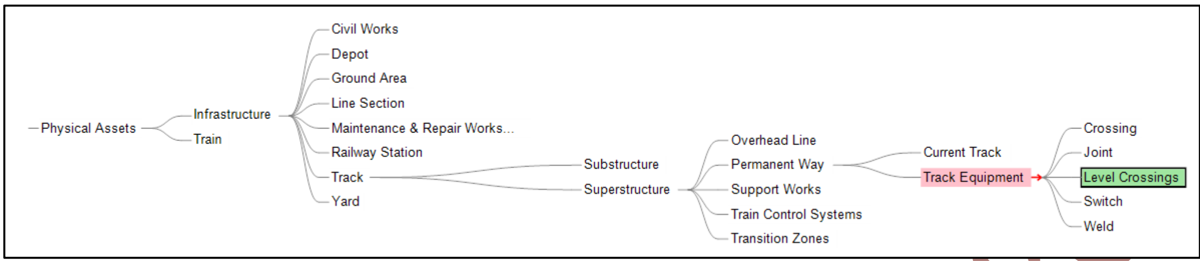


Figure 3.26: Position of Level Crossing into the railway taxonomy

3.12.2. Asset sub-components

Level Crossings are simple yet fundamental assets of a railway system. They are composed of a few complex sub-components, as shown in Figure 3.27, which aim at preventing cars and people from trespassing the railway tracks in case a train is transiting. Although there can be several different types of level crossings (barrier crossings, gated crossings, etc.), they can be all categorized by using the sub-components that have been identified. Figure 3.28 shows a level crossing schematically with labels identifying parts.

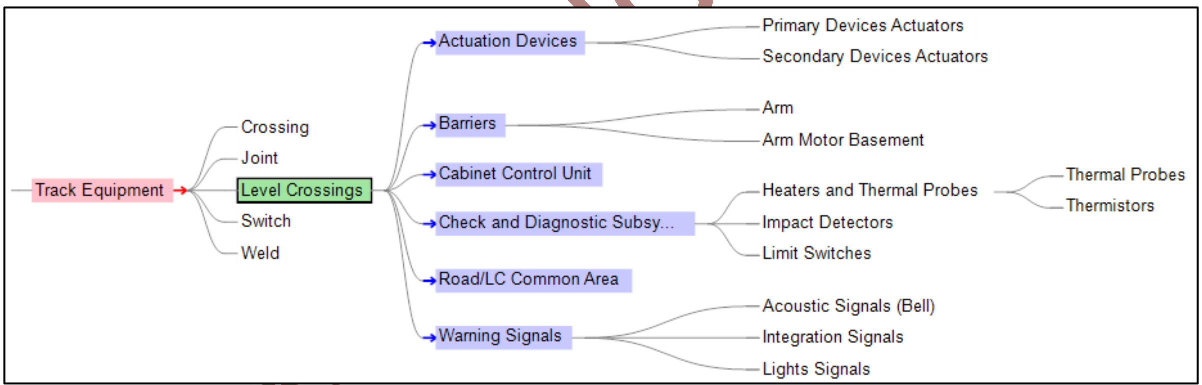


Figure 3.27: Level Crossings sub-components classification

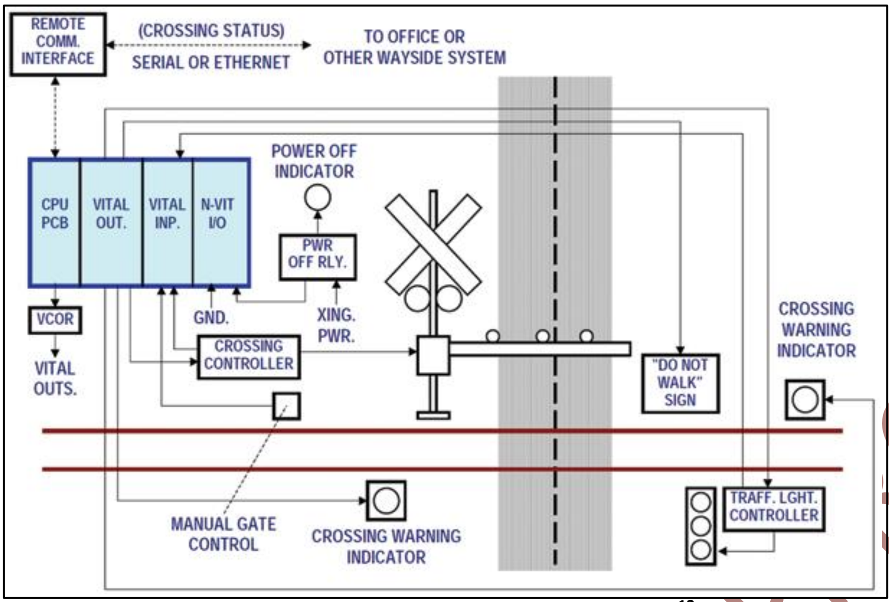


Figure 3.28: Level Crossing technical picture[13]

The following list recalls in textual form all the most relevant components of a level crossing:

- Actuation Devices:
  - Primary Devices Actuators,
  - Secondary Devices Actuators;
- Barriers:
  - Arm,
  - Arm Motor Basement;
- Cabinet Control Unit;
- Check and Diagnostics:
  - Heaters and Thermal Probes:
    - Thermal Probes,
    - Thermistors;
  - Impact Detectors;
  - Limit Switches (or equivalent devices);
- Road/LC Common Area;
- Warning Signals:
  - Acoustic Signals (Bells);
  - Integration Signals;
  - Lights Signals.

3.12.3. Data Tables

All the data mentioned in the following tables (or, at least, most of it) should be stored and collected regularly to create an historical database containing all the possible information related to the functional behaviour of the monitored devices over their entire lifetime.

[13] This picture is taken from an Ansaldo STS public brochure for Level Crossing solutions (*http://www.ansaldo-sts.com/sites/ansaldosts.message-asp.com/files/imce/asts_hitachi_lc_solutions_english_092015_lr.pdf*).
MicroLok® is a registered trademark of Ansaldo STS USA, Inc.

STATIC Data			TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Type	Enumeration (automatic full barrier, automatic half barrier, gated, automatic open with lights, automatic full barrier with obstacle detection, automatic half barrier with obstacle detection, etc.)	Level Crossing (LC)	✓		
Model	String	LC	✓		
Barrier Length	Decimal (m)	LC	✓		
Absolute Position	Geospatial	LC	✓		
Mileage	Decimal (km)	LC	✓		
Nominal Arms Motor Voltage	Decimal (V)	Primary Devices Actuators	✓		
Nominal Arms Motor Current	Decimal (A)	Primary Devices Actuators	✓		
Nominal Arms Motor Power Consumption	Decimal (W)	Primary Devices Actuators	✓		
Nominal Hydraulic Pressure	Decimal (PSI/bar)	Primary Devices Actuators	✓		
Nominal Drop Time	Time	Primary Devices Actuators	✓		
Nominal Rise Time	Time	Primary Devices Actuators	✓		
Nominal Acoustic Power	Decimal (dB)	Acoustic Signals	✓		
Nominal Light Signals Voltage	Decimal (V)	Light Signals	✓		
Nominal Light Signals Current	Decimal (A)	Light Signals	✓		
Nominal Max Speed per track and direction	Decimal (km/h)	LC	✓		
Arms Material (???)	String	Arms			
Construction date	Time	LC			
Construction series	string	LC			
Original Test	Time	LC			

STATIC Data			TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**NR**	**RFI**	**TRV**
Date					
Installation Date	Time	LC			
Striking distance	Decimal (m)	Crossing	✓		
Minimal warning time	Time	Crossing	✓		

Table 3.15: Level crossing: static data

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Current position	Decimal (degrees)	Arm	Internal	✓		
Device Temperature	Decimal (°C)	Thermal Probes	Internal			
Device Status	Enumeration (OK, KO, No-Response, etc.)	LC	Internal	✓	✓	
Environmental Humidity	Decimal (%)	LC (environment)	External			
Environmental Pressure	Decimal (bars)	LC (environment)	External			
Wind Speed	Decimal (m/s)	LC (environment)	External			
Wind Direction	Decimal (degrees)	LC (environment)	External			
Path Status	Image	Road/LC Common Area	Diagnostic	✓		
Flood alert - water level	Decimal (mm)	LC	External	✓		✓
Flood alert – digital	Binary	LC	Asset-related (Track circuits)	✓		
Snow/Ice detection	Binary	LC (environment) Actuation devices	External	✓		✓
Motor Voltage (Primary/Secondary)	Decimal (V)	Actuation Devices	Internal	✓		
Motor Current (Primary/Secondary)	Decimal (A)	Actuation Devices	Internal	✓		
Barrier Drop Time	Time	Actuation Devices	Internal	✓		
Barrier Rise Time	Time	Actuation Devices	Internal	✓		
Hydraulic Pressure	Decimal (PSI/bar)	Primary Devices Actuators	Internal	✓		

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
Open/Close cycle total number	Integer	Actuation Devices / Limit Switch	Internal	✓		
Moving arms bearings and gears status (lubrication, wear, rust, etc.)	N.D.	Actuation Devices	N.D.	✓		
Acoustic Signals Voltage	Decimal (V)	Acoustic Signals	Internal	✓	✓	✓
Acoustic Signals Current	Decimal (A)	Acoustic Signals	Internal	✓	✓	✓
Acoustic Signals Feedback	Sound	Acoustic Signals	External (microphone)	✓	✓	✓
Light Signals Voltage	Decimal (V)	Light Signals	Internal	✓	✓	✓
Light Signals Current	Decimal (A)	Light Signals	Internal	✓	✓	✓
Clearance status (from users point of view)	Binary	Road/LC Common Area	External (clearance sensor e.g. laser, CCTV, etc.)	✓	✓	✓
Clearance status (from train point of view)	Binary	Road/LC Common Area	Asset related (signalling - interlocking)	✓	✓	✓
Road/LC Common Area recording	Image	Road/LC Common Area	External (CCTV)	✓		
Limit Switches status	Binary	Limit Switches	Internal			
Damaged barriers	Binary	Impact detectors	Asset-related	✓	✓	✓
Number of maintenance interventions	Integer	LC	Maintenance	✓		
Scheduled maintenance interventions frequency	Time (each “X” days, months, etc.)	LC	Maintenance	✓		
Maintenance interventions date	Time	LC	Maintenance	✓	✓	✓
Maintenance intervention start time	Time	LC	Maintenance	✓	✓	✓
Maintenance intervention end time	Time	LC	Maintenance	✓	✓	✓
Maintenance intervention type code	String (code)	LC	Maintenance	✓		
Maintenance intervention description	String	LC	Maintenance (could be supplied by	✓		

DYNAMIC Data				TMS Critical Variables		
Name	**Data Type**	**Refers to component**	**Source info**	**NR**	**RFI**	**TRV**
			humans)			
Code for failure that determined a maintenance intervention	String (code)	LC	Maintenance	✓		
Maintenance team that made the intervention	String (code)	LC	Maintenance			
Number of road vehicles transited [14]	Integer	Road/LC Common Area	External (vehicle detection camera, vehicle detection sensors, etc.)	✓		

Table 3.16: Level crossing: dynamic data

[14] This field could be also used to optimize timetable planning correlating road and train traffic

4. Asset status representation

This section presents the representations selected for asset status information by the project team. The asset status representation has been developed to enable the exchange of key asset parameters (as captured from on-asset sensors) and, by extension, the status/availability of the asset, within the context of the nowcasting & forecasting functions of the In2Rail project. It is important to note that, while the nowcasting and forecasting modules are likely, ultimately, to access this data via the In2Rail integration layer, the asset status representation presented here is (at present) purely for use in WP9, and data from the integration layer, as with data from other asset data systems, is expected to be mapped to this representation at the WP9 interface.

4.1. Key elements of asset status

The selection of an appropriate representation for asset status must necessarily start with a consideration of the various aspects of the domain that need to be captured. In section 3.1 of this document the distinction between the static and dynamic elements of the asset representation was introduced as seen below, with the combination of the two contributing to the complete representation (as shown in Figure 4.1).

- **Static data:** related mainly to static characteristics of the asset under examination (e.g. GPS absolute position, ...);
- **Dynamic data:** data coming either from recordings of the usage of the asset (e.g. number of trains passed over the asset, ...) or from external devices/sensors (e.g. environmental temperature, rail profile measurements, ...) and maintenance operations (e.g. number of maintenance operations during lifetime, ...).

While the inter-relationship of static and dynamic data in a single domain is of course comparably commonplace, it is usually avoided in data models. This is primarily because models designed for dynamic data need to result in compact but context-rich representations that can convey a specific sub-set of information rapidly and efficiently (in terms of bandwidth usage), while models intended for use with more static data can afford to be more verbose in exchange for greater flexibility in the range of content that can be represented. In the rail industry, this distinction is most obvious in fields such as operations, where more static data (e.g. seasonal timetables) is represented in XML based formats including railML, but more dynamic data, such as the movements of vehicles between track circuit bays, is frequently streamed as JSON or similar, with the streaming data referencing elements of the more complex static model, but not directly including the detailed information.

STATIC Data		
Name	**Data Type**	**Refers to component**
Type	Enumeration (ordinary, inside curved, outside curved, three way)	Switch
Model	String	Switch
Normal Position	Enumeration (straight, left, right)	Switch (Rail)
Length	Decimal (m, mm, etc.)	Switch
Absolute Position	Geospatial	Switch

DYNAMIC Data			
Name	**Data Type**	**Refers to component**	**Source info**
Current position	Enumeration (straight, left, right)	Switch - Points - Rail	Internal
Device temperature	Decimal (°C)	Switch - Thermal probes and heaters	Internal
Device status	Enumeration (OK, KO, No-Response, Sufferance, etc.)	Switch	Internal
Environmental Humidity	Decimal (%)	Switch (environment)	External

Complete Asset Status Representation

Figure 4.1: Static and dynamic data elements contributing to a complete asset status representation

The need to represent both static and dynamic data within the In2Rail project, suggests that a hybrid approach to the representation of the data will be the most effective, and, on that basis, the main data modelling effort in work package 9 has focussed on the identification of a combination of the models available in the two domains (infrastructure and sensors / observations) that will:

- Complement each other when used in combination;
- Describe the TMS critical variables identified in the asset information specifications in Section 3;
- Allow capture of as many of the none-critical variables as possible.

With that goal in mind, the remainder of Section 4.1 will introduce some the key concepts involved in the description of asset status, before specific models are discussed in Sections 4.2 and 4.3, and finally examples of the usage of the models are presented in Section 4.4.

4.1.1. Infrastructure type

For obvious reasons, the type of infrastructure being studied is of vital importance to the asset status representation – the difference between a current waveform from a switch motor and a level crossing barrier, while similar in format, is huge in terms of interpretation. Fortunately, given the long-established history of the railway industry, the main categories of infrastructure assets are both well-known and common to most railway systems (although local differences exist in terms of precise details such as power supply etc.), meaning that it is safe to assume that the majority of the infrastructure models available on the market will have coverage of the asset categories defined in section 3.

4.1.2. Physical location

"Location" is a concept that underpins much of what must be done, and is certainly one with which humans are all familiar. In the rail industry, however, the description of a specific location has long proved challenging, with several systems being used each based on different references. Generally speaking, these systems can be broken down into two main groupings, absolute and relative positions.

4.1.2.1. Absolute geographical position

Absolute positions are the easiest of the two groups to explain, and represent specific points on the Earth's surface described by a coordinate system (e.g. an OS grid reference, or a WGS-84 position), normally augmented with a specified projection to account for differing height profiles of the ground and the non-spherical shape of the Earth. Absolute geographical positions have, historically, been difficult to calculate, requiring either surveying equipment (such as a sextant) or a map and line-of-sight to several visual references in the surroundings. Both these systems were inconvenient in the early days of the railways, particularly in cuttings or tunnels, and so relative positioning systems (see below) were adopted by the industry.

The arrival of Global Navigation Satellite Systems (GNSS), and the subsequent inclusion of positioning hardware in smartphones and tablets, has made absolute positioning a much more practical system for use by the railways in recent years. Modern infrastructure management tools, such as those provided by Network Rail's ORBIS programme in the UK, now common use absolute positions provided by the United States' GNSS, the Global Positioning System (GPS), to locate maintenance teams on the lineside, and vehicles are increasingly equipped with GPS alongside other detection / positioning technologies. In the coming years, also the EU Galileo positioning system will contribute to increased adoption of positioning systems in the railway.

4.1.2.2. Infrastructure-relative position

Infrastructure-relative positions were adopted by the early rail industry as a convenient means of describing locations on the infrastructure, and are normally given as a distance along a known route or track. In the UK, this type of position is normally reported using a combination of an Engineering Line Reference (a short alphanumeric code assigned to a particular route), a track ID and a distance measured from a major station (e.g. London Euston). On linear assets, such as the railway, relative positioning is a quick way to establish a location that can be easily determined based only on the distance a vehicle has moved along a known track. When working with data from outside the railway however, or data tagged using other positioning systems, relative positions quickly become difficult to work with as complex conversions are needed to switch position references between one system and another.

4.1.3. Dynamic state

The dynamic state of an asset is the key contribution of the asset status representation to the traffic management system, and will rely on well described data from sensors in the field. As seen in section 3, the range of formats for sensor data currently being used within the industry is comparably broad, even for data that are considered “critical” to the traffic management process. This will mean that the data model for the dynamic state data will need to be flexible, capable of specifying a variety of encodings as the specific dataset requires, and ideally be able to handle that data in a decoupled way, to avoid passing large amounts of potentially very sizable data around the traffic management system unnecessarily.

4.1.4. Actionable status

The delivery of actionable status information from asset condition is an important element of the integration of the asset status data with the traffic management system. Actionable data can be thought of as a “go / no go” type message that describes whether the asset is currently available for use. Actionable data will need to be derived based on the combination of sensor data, knowledge of the asset and its behaviour, and the business rules of the owning IM. The delivery of this type of information is easiest when using conceptually rich data models, such as ontology, however the choice of this type of model currently available on the market is limited. If more traditional model families are adopted for use in the project, then a simpler software-driven approach may need to be taken on an asset by asset basis.

4.2. Review of models

4.2.1. Static Infrastructure

The representation of the static infrastructure is a critical component of many ICT systems within the rail industry. Surprisingly however, it is a field where standardisation of formats has been hard to achieve across the various stakeholders in Europe and further afield, and several application-specific formats are in use. These are a mix of mandated formats for specific tasks (e.g. the RINF format discussed in section 4.2.1.3), and community or vendor developed formats serving particular domains. A detailed survey of infrastructure models was developed by the EU-funded Capacity4Rail project, and presented in C4R project deliverable 3.4.1 [1] – the outcomes of that work, and specifically the key models identified in the field, have informed the content of this section.

4.2.1.1. railML

The original railML models (railML, railML 2) are a series of XML schemas produced by a loose consortium of railway companies, academic institutions, and consultancy firms. The models are designed to capture four, largely static, elements of the railway and railway operations (full details may be found in C4R D3.4.1 [1]):

- Common concepts;
- Timetable;
- Rolling stock;
- Infrastructure (at the macroscopic and microscopic levels).

Interlocking design has also been considered for inclusion in the past, however at present this is being held for railML 3 (see section 4.2.1.2).

The traditional railML Infrastructure model handles microscopic and macroscopic representations of the infrastructure separately, with one topology (macroscopic) for lines and operational control points, and another (microscopic) for more detailed constructs such as switches, crossings, and track sections. This mechanism has worked well at an application-level for several years, but risks inconsistencies between the two granularities of model, a problem now being addressed in railML3.

Alongside the network topology, the railML infrastructure model can also contain information on the locations and equipment types of key operational & control assets, e.g. signals, balises, axel counters and level crossings, as well as the presence of linear assets such as electrification equipment. However, while the model can describe the presence of these assets, it does not capture their dynamic state (e.g. the lie of a particular switch, or the presence of an obstruction on a crossing). Operational parameters, such as speed limits, can also be captured.

4.2.1.2. UIC RailTopoModel / railML 3

RailTopoModel[15] is a UIC-endorsed topological model for the representation of railway infrastructure. Recently proposed as an international standard (IRS30100), the model has been developed in close cooperation with the railML consortium, and version 3 of the railML[16] standard, which is currently under development, will serve as the reference "exchange format" implementation for RailTopoModel within the community.

RailTopoModel is designed to address the long-standing problem of aggregation of network topology data from different levels of granularity, as used for a range of applications within the industry. Prior to RailTopoModel infrastructure models including the previous version of railML handled different granularities of infrastructure layout data in separate models (or at least as separate instances of the same model), an arrangement that means it is possible for inconsistencies to arise at different levels of the infrastructure topology abstraction. RailTopoModel addresses this problem by allowing higher level versions of the infrastructure topology to be derived from lower level data, ensuring that the higher-level data is consistent. The ability to abstract infrastructure layouts to different levels of granularity means that RailTopoModel (and by extension railML 3) is well suited to acting as a common

---

[15] http://www.railtopomodel.org/en/
[16] http://www.railml.org/en/

intermediary between different representations of the same infrastructure data, and with the large number of mandated infrastructure formats in use within the rail industry, the railML consortium is hopeful that the new model will provide a common interface between the local infrastructure databases of individual IMs, national databases representing the whole infrastructure of a country, and the mandated requirements for international reporting associated with initiatives such as RINF (see section 4.2.1.3) or INSPIRE (see section 4.2.1.4).

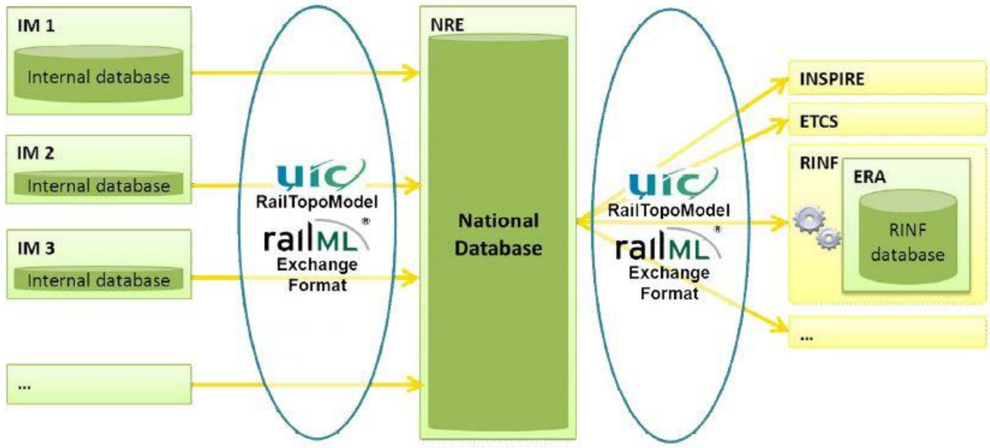


Figure 4.2: A vision of the future of infrastructure data exchange using RailTopoModel and railML [7]

4.2.1.3. RINF

The Register of Infrastructure (RINF) legislation[17] requires that member states provide certain basic information on their railway infrastructure in a common format, to ease the planning and implementation of cross-border passenger and freight services. Under the legislation (which is one of the use-cases described in Figure 4.2) a national authority in each country is responsible for gather the critical infrastructure information, and then reporting it to the European Rail Authority (ERA) via a common interface. The RINF data exchange itself is XML based, and describes the infrastructure in terms of a series of operational points (OPs) connected by sections of line, where:

- A line is a sequence of one or more sections, which may consist of several tracks;
- A section of line is the part of line between adjacent OPs and may consist of several tracks;
- Operational points are locations for train service operations for example where train services can begin and end, change route and where passenger or freight services are provided;
- Stopping points for passengers on plain line are also regarded as OPs;
- Operational points may be locations where the functionality of basic parameters of a subsystem are changing for example: track gauge, voltage and frequency, signalling system;

[17] http://www.era.europa.eu/Core-Activities/Interoperability/Pages/RINF.aspx

- Operational points may be at boundaries between member states or areas of control of different Infrastructure Managers;
- Passing loops and meeting loops on plain line or track connections only required for train operation do not need to be published;
- Sidings are all tracks not used for train service movements.

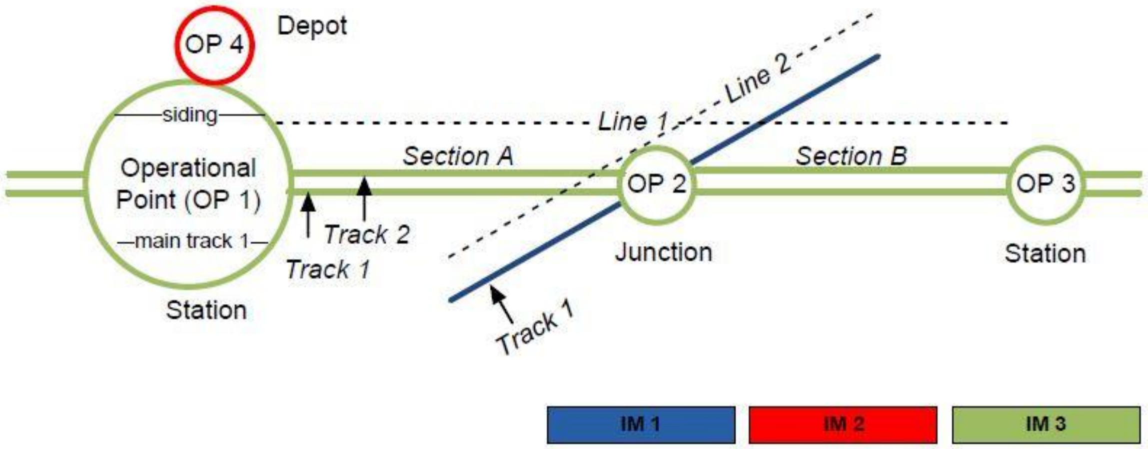


Figure 4.3: The conceptual structure of the RINF data model, elements are coloured according to the IM responsible [2]

4.2.1.4. INSPIRE

The Infrastructure for Spatial Information in Europe (INSPIRE) directive is designed to support the EC's decision making processes around the environment and sustainability by providing data on 34 key themes in common formats, with one of the themes being transportation networks / transport infrastructure. An example of the railway elements found in the INSPIRE transport networks schema can be seen in Figure 4.4. The data would then be aggregated via the INSPIRE geoportal (http://inspire-geoportal.ec.europa.eu) for use by the European Commission and other stakeholders; at the time of writing, the current roadmap for provision of data under INSPIRE set the completion date as October 2017. An interesting aspect of the provision of data under RINF and INSPIRE, is that while they exist for different reasons, the operational data required by RINF is essentially an overlay on a subset of the infrastructure data required under INSPIRE, and therefore there is likely to be a significant overlap between the model coverage for these two initiatives.

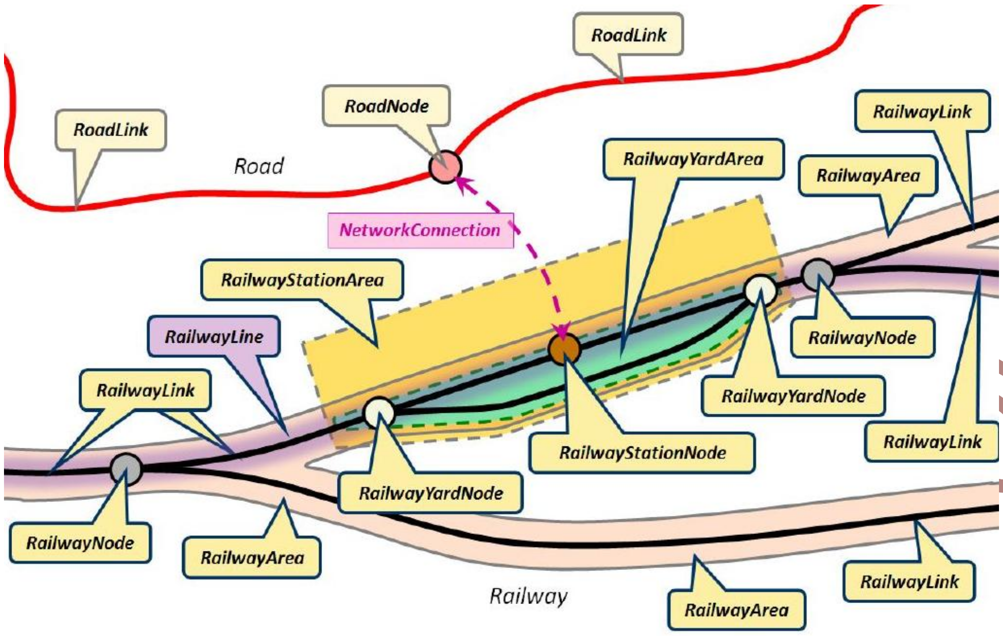


Figure 4.4: An overview of the main railway transport elements in INSPIRE [3]

4.2.2. Sensor data

The use of sensor data, and in its more general sense asset status, within the field of traffic management is still very much in its infancy. At present, the rail industry installs and uses sensors on assets primarily in relation to remote condition monitoring, and the models used in those systems reflect that role (for example the use of the Mimosa OSA-CBM / ISO 13374 standard [5] by Network Rail within Intelligent Infrastructure). As automated assessment of asset status becomes a more important element of traffic management moving forwards, it will be necessary for increasingly context-rich descriptions of sensors to be used, enabling the software systems to use data from a wide range of assets and sensors with the appropriate business logic to derive actionable asset status information. This section of the document will introduce three such context-rich sensor data models that were considered for use as the dynamic data component of the In2Rail asset data representation.

4.2.2.1. Sensor Web Enablement

The Open Geospatial Consortium's (OGC) Sensor Web Enablement (SWE) framework [8] are designed to enable developers to make sensors and sensor data repositories available online. The framework, which is backed by over 300 companies and research organisations worldwide, covers all the main aspects of sensor data collection and delivery, including specifications for open interfaces, sensor service descriptions, feasibility planning for sensor installations, and driver management, however it is the sensor data processing and observation models that are most relevant to the In2Rail asset status representation work. The SWE framework is also compatible with a range of other models that are made available by the same consortium, including the Geography Markup Language (GML) and IndoorGML specifications, the Location Services (OpenLS) specifications for developing location-based software applications, and GeoSPARQL, a query language for accessing geospatial data via the Semantic Web. Within the SWE framework, the description of sensors and sensor data are handled by the SensorML and Observation and Measurement XML models, although the O&M model is of limited applicability if using simple encodings (e.g. comma separated text or similar).

The SensorML model is designed to allow the description of the processes of data collection and transformation, including the description of any sensors and actuators that are involved in the process. From a railway perspective, this can be thought of as a description of a crossing barrier, sensors attached to the asset such as a current clamp on the barrier motor, and the processing performed on the current waveform such as down sampling of the data and baseline adjustment. As with many XML-based standards, SensorML includes a certain amount of flexibility in terms of the information that must be included in a valid file, and as a result can be expressed in a very compact form if required, although this comes at the loss of contextual information about the data.

One very convenient feature of the SensorML model is that it includes native support for describing sensor data that is accessed via resources remote to the file, via web services or similar), this provides a useful way of dividing raw sensor data values from the description of the sensor system that is generating them, and hence (with an appropriate choice of encodings) allows very effective use of bandwidth once the system configuration itself has been described.

4.2.2.2. Semantic Web for Earth and Environmental Terminology

The Semantic Web for Earth and Environmental Terminology (SWEET) ontologies [10] are designed to provide an upper-level ontology model for earth and environmental science work. Developed by the NASA Jet Propulsion Lab in Pasadena, the models enable the representation of natural phenomena, human activities, and most importantly from the perspective of the In2Rail project, the data that is used to describe them (processes, states, and observations).

SWEET is based around ontologies, conceptual models of a world-space that were developed to inherently capture the context of data items for use on the Semantic Web. Ontologies are most easily thought of as data models that can allow a machine to make reasoned inferences in much the same way as a human, in the rail domain early examples of this could be seen in the framework 6 INTEGRAIL project [4], which used ontology as the basis for automatic network statement checking, as well as inference of vehicle status in condition monitoring (e.g. inferring that a vehicle was faulty because one of its axles had a hot box etc. etc.). The additional contextual information also makes ontology-type models popular choices for representing metadata used by simpler, less expressive models – SWE for example uses elements both of its own ontology repository, and SWEET's, for metadata in its XML model.

As with SWE, SWEET is able to capture all the details of sensor configuration etc. that might be needed for an exercise such as the asset status representation for In2Rail. As an ontology, its use would also enable the use of automated reasoning approaches to infer the asset status directly from the sensor data, delivering a very clean architectural model in which the business rules were encapsulated in the data model rather than in the application logic. However, the existing coverage for railway specific topics in ontology models is very limited (one of the most detailed examples at present is the Rail Core Ontology, RaCoOn [6], that has been developed by the University of Birmingham), and the steep learning curve associated with the use of such contextually rich models may prove a problem in the limited time available to the In2Rail project.

4.2.2.3. Semantic Sensor Network

The Semantic Sensor Network (SSN) ontology [9] is the World Wide Web Consortium's (W3C) suggestion of how sensor data should be made available on the web, and as with SWEET is based around the context-rich ontology model family. The SSN model enables developers to describe sensors, the data they produce and how it is measured, and to identify the assets the sensors are installed on, as well as enabling the description of key supporting data, such as the remaining useful battery life of a remote sensor.

The SSN differs from SWEET in that the developers have adopted a more open root to the model, choosing to extend open concepts in other public models (notably the widely used DOLCE Ultralite model) rather than create their own representations of time etc. The use of common "upper level" ontologies means that data exchanged and integrated by the users of different domain models (e.g. ontologies for the rail and highway domains) are more likely to be directly comparable, however this is at the expense of a loss of "control" of the root concepts by the maintainers of the SSN ontology themselves.

The W3C's endorsement of SSN makes it a tempting choice for use within the In2Rail project, but as with SWEET the learning curve associated with effectively using ontology means that it may be impractical given the time available.

4.3. Model choices

As mentioned in the previous sections, the choice of models for use in the asset status representation was driven by two main factors: the need to represent both the static (position, asset type, installed configuration etc.) and dynamic (crossing barrier position, lie of switch) within the model, and the need for the data models chosen to complement each other in terms of structure, implementation and coverage of key topics.

4.3.1. Static data

From the perspective of the static infrastructure three main options were considered – the railML family of models, and the infrastructure models presented in response to the RINF and INSPIRE directives from the EU. Other infrastructure models are of course available and in use, however the project team was of the firm opinion that selecting a model that was officially sanctioned, either by the EU in the case of RINF and INSPIRE, or by the UIC in the case of RailTopoModel / railML 3, would add a stability to the formats that was a desirable characteristic where more than one model would be used in tandem to deliver the final asset data representation. This is often not the case in independent open source models, as the developers are often making continual adjustments to support new features.

When comparing the models in more detail, all three models specify XML implementations/serialisation, which are capable of supporting the type of extended use that is being proposed for the asset status representation. From the perspective of the level of detail in the representations of the infrastructure, both RailTopoModel / railML 3 and INSPIRE cover a much a larger set of railway concepts than RINF, which is primarily focussed on the description of the infrastructure for the planning of services on cross-border corridors. This limited the usefulness of RINF in the domain of interest of the In2Rail asset representation, and so the model was ruled out at this stage.

The choice between RailTopoModel / railML 3 and the INSPIRE models was less clear cut, with both able to represent the basic layout of the infrastructure, albeit at differing levels of detail. The INSPIRE model offered a couple of potentially useful fringe benefits over RailTopoModel / railML 3; first and foremost amongst these was the fact that data appropriate to the format is already being collected to support the INSPIRE activities, and therefore would be available to Infrastructure Managers within Europe without dedicated data collection activities needing to take place. Secondly, the wider focus of the INSPIRE directive on 34 themes, rather than just on rail, would mean that the model was better suited to linking to non-rail assets; although non-rail assets are currently out of scope for the In2Rail data representation, it is easy to see how in contexts such as tramways or non-segregated light rail, or in the context of traffic management within a Smart City, it may be useful to include non-rail infrastructure assets in the future.

Alongside the fringe benefits of adopting INSPIRE, there is at present one major issue with the use of railML 3, in that the model itself is still only available in an alpha release. This is obviously an issue, however, the alignment of the railML 3 infrastructure development with RailTopoModel, which has already been released as IRS 30100, means that the core concepts being addressed by the model are already clear, and this is sufficient for use at this stage of the asset data representation development process.

Ultimately the choice of the infrastructure component of the asset status model has been made based on the fact that In2Rail is at this stage a rail industry project, and although interactions with external modes will be a key developmental area for the industry going forwards the far greater depth of coverage of rail infrastructure assets associated with the RailTopoModel / railML 3 models, coupled with the railML developers' vision for the model to serve as a conduit for creation of the INSPIRE content from national rail industry databases (and hence for railML to include all rail concepts needed by INSPIRE as a minimum), meant that, at present, railML 3 is the appropriate infrastructure representation for use within the asset status model.

4.3.1.1. Integration of In2Rail infrastructure model requirements in railML®

A central outcome of this In2Rail project deliverable is the decision that the XML based format railML® shall be used for the exchange of infrastructure related static data between the stakeholders and IT applications of In2Rail. As the asset status representation requires also modelling and exchange of dynamic information, e.g. the status of an infrastructure element, SensorML has been selected for these dynamic data. railML® and SensorML together provide the selected solution for a data modelling format required by the In2Rail application. The following section answers the question how to get from the infrastructure model requirements formulated in chapter 3 to their integration in the railway data exchange format railML®.

As a first step, **a railML® use case** has been set up based on the content of this deliverable. This use case has been submitted to railML.org being the institution that coordinates the development of the railML® data exchange format. The ideas and requirements for the railML® schema improvement are formulated by the railML® community using web based communication technologies, i.e. a forum, a Wiki documentation and a Trac ticket system. IN2Rail followed this community approach by writing a use case wrapping up the infrastructure related requirements of IN2Rail application. This use case is publicly available in the railML® Wiki [20]. Figure 4.5 shows a screenshot of this website.

Based on the description of the railML® infrastructure scheme use case "Asset Status Representation" it was concluded that the existing railML® version (2.3) as well as its predecessor versions are not sufficient for matching all the requirements of IN2Rail application. Fortunately, railML.org is currently coordinating and pushing the development of a new baseline of its data exchange format, which will be named **railML® version 3.** According to railML.org this new version of railML® will be able to handle many more use cases than baseline 2. Therefore, the railML® infrastructure scheme use case "Asset Status Representation" has been set up as a railML® v3 use case.

Figure 4.5: The railML® infrastructure scheme use case "Asset Status Representation"

The new version of the railway infrastructure data exchange format, railML® v3, is being developed by railML.org. Its primary focus is on the fundamental re-structuring of the infrastructure schema. The roadmap set up by railML.org proposes the release of a first usable version of railML® v3 by February 2017 (see [21]). To have a more direct influence on the development of railML® v3, DLR decided to join the "railML® v3 early users group" that has been set up to test the evolving railML® v3 alpha versions and to provide valuable feedback. This feedback is more or less directly incorporated in the ongoing development of railML® v3 alpha. By joining this group DLR representing the IN2Rail project consortium want to make sure that its infrastructure model requirements are discussed within the railML® community and most likely considered for implementation in the first railML® v3 release in February 2017. In particular, DLR is going to challenge the current railML® v3 alpha schema files (XSD) against their requirements and provide feedback on possible gaps. This feedback will be communicated also via the railML.org forum so that the railML® community has the chance to comment on it and ideally support the "Asset Status Representation" use case and make it more generic.

With the formulation of the railML® v3 infrastructure scheme use case "Asset Status Representation", the railML® related activities of In2Rail WP9.1 are finished. The results of this work will be used as input for Shift2Rail initiatives. As the new railML® v3 data exchange format will evolve step by step over the upcoming years, the main task of Shift2Rail railML® activities shall be the constant challenge of the model against the asset status representation use case. Further, Shift2Rail is going to put its focus on the dissemination of the described use case on European level in order to support use case generalization among different European railway infrastructure managers. However, it must be stressed that the aforementioned goals can only be reached if a constant contribution towards railML® development and railML.org user community can be guaranteed.

4.3.2. Dynamic data

4.3.2.1. Serialisation

The SensorML 2 standard provides several different options for the encoding of the sensor data themselves. In the case of the In2Rail asset data representation, the decision taken was to enable access to these data via RESTful web service interfaces (thus enabling large items of data, such as images of level crossings, to be handled without needed to pass the data around at all times), a modelling choice that has the additional benefit of making it very easy to represent the majority of the dynamic data values as simple text serialisations.

Examples of the text serialisation can be seen in section 4.4, however in brief a simple representation of two timestamped values from a single sensor could look as follows (the exact format of the string can be defined in the XML description on a sensor-by-sensor basis). Each <timestamp, value> pair is split by a comma, with multiple pairs then differentiated by a space (below this has been exaggerated for ease of reading).

2016-09-06T05:30:00Z,10.0   2016-09-06T05:32:00Z,12.0   2016-09-06T05:34:00Z,15.0

4.3.3. Architecture

The proposed model, based on railML, RailTopoModel and sensorML, is aimed at representing the status of the assets in a data layer, like the one used in WP9, in which the real-time constraints are relatively weak.

The logic behind can be synthesised as follows:

- The proposed model provides an interface between the WP9 big data architecture (see deliverable D9.2 [22]) and the external models for all the variables that are collected or generated (by the nowcasting and forecasting modules) within or published by WP9;
- The proposed model does NOT prescribe TMS in which format to publish data and does NOT specify any part of data grid in Integration Layer inside of TMS.

From the architectural perspective, the representation makes deliberate use of web service endpoints as the primary means of accessing the underlying values of the various parameters described. While many of the parameters represented could have been included directly in the XML, with whole XML messages being transmitted when updates occurred, the WP9 team decided that, particularly for larger packets of data (such as images) this would place an unacceptable load on the communications layer – particularly given there was no guarantee the nowcasting / forecasting tasks in progress would need that information. The need to provide web service endpoints for the complex data items, meant that from the perspective of consistence, it was desirable to use the same model for all values, and hence accessing all data in this way is recommended.

The examples presented in section 4.4 all used a polled endpoint for this purpose, as it was anticipated that nowcast & forecast modules would not be running continuously and hence would only require fresh data when beginning a new processing cycle. However, if polling is not felt to be an appropriate method for accessing data at scale in a deployed system, it should be noted that the SensorML model also provides for streaming endpoints, and these could easily be used in place of the polled variant described below.

4.4. Worked examples

This section presents two simple worked examples showing how the railML and SensorML fragments for selected TMS critical variables from a level crossing and from a switch can be combined to produce a usable asset status representation. railML 2 fragments are used for the infrastructure, as at this stage railML 3 is still not formally available. railML 3 is expected to be available in beta before the end of 2016, and if this target is achieved, it is anticipated that railML 3 will be used in place of railML 2 in the "production" version of the asset status representation. Examples are based on the official documentation from the respective consortia, see railML and SensorML websites for further details:

- railML (https://www.railml.org/en/);
- SensorML (http://www.opengeospatial.org/standards/sensorml).

4.4.1. Level crossing

The first example chosen for the asset status data representation is the level crossing, with a particular focus placed on one TMS critical variable, the flood water level (this is used of clarity and simplicity of the example, and the extension to other critical variables is trivial). The full description of the TMS critical and other variables for this asset can be found in section 3.12.

4.4.1.1. Level crossing – railML 2 representation of static infrastructure

Beginning with the static data, a (fragment of a) railML 2 representation of a simple crossing can be found below. The wider representation of the infrastructure is left out for simplicity.

```
<track id="t26" name="track_26">
    <trackElements>
        <levelCrossings>
            <levelCrossing id="lc211" name="level_crossing_melton_mowbray_211" pos="25.2"
                dir="none">
                <geoCoord coord="52.761955 -0.819494" epsgCode="urn:ogc:def:crs:EPSG::4326"/>
            </levelCrossing>
        </levelCrossings>
    </trackElements>
    <trackTopology>
        <trackBegin id="tt2001" pos="0"/>
        <trackEnd id="tt2002" pos="38.0"/>
    </trackTopology>
</track>
```

Graphically, this fragment can be summarised as follows:

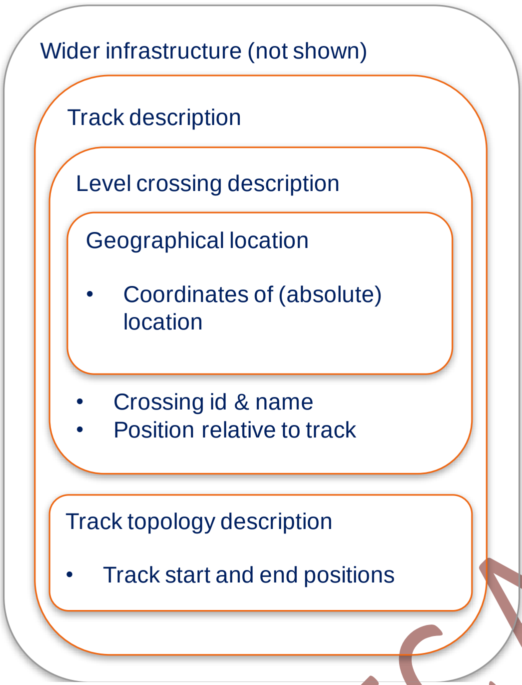


Figure 4.6: Simplified graphical view of railML 2 fragment for level crossing

The fragment shows a single track, with starting and ending mileages given by the pos attributes of the topology elements. Alongside these is the description of the crossing itself, defined with a unique id and short name, a relative position (defined as a distance along the tracks), and a geoCoord giving an absolute position and projection system (in this case WGS84, but referred to by its EPSG code).

4.4.1.2. SensorML 2 representation of flood water depth sensor

A simple SensorML representation of a sensor (in this case a floodwater depth sensor) is shown below, in this case the sensor is only described in terms of a human readable description of its function, a quantity that it measures (to which a value could be attached as an extra element if you wanted to transmit sensor value data embedded in the XML – this is not the design choice made in In2Rail), and a description of the physical location in which the sensor is installed. The physical installation location may be useful on a large site, or in the cases where a detailed frame of reference (e.g. specifics of rotation of sensor) are required, however in most cases most the location data in the railML file is likely to be sufficient.

```
<sml:PhysicalComponent gml:id="FLOODWATER_DEPTH_SENSOR"
    xmlns:sml="http://www.opengis.net/sensorML/2.0"
    xmlns:swe="http://www.opengis.net/swe/2.0"
    xmlns:gml="http://www.opengis.net/gml/3.2"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xsi:schemaLocation="http://www.opengis.net/sensorML/2.0
                            http://schemas.opengis.net/sensorml/2.0/sensorML.xsd">
    <!-- ================================================ -->
    <!-- System Description -->
    <!-- ================================================ -->
    <gml:description>
```

```xml
        Fixed location floodwater depth sensor
    </gml:description>
    <!-- ================================================= -->
    <!-- Observed Property = Output -->
    <!-- ================================================= -->
    <sml:outputs>
        <sml:OutputList>
            <sml:output name="waterLevel">
                <swe:Quantity definition="http://in2rail.eu/ont/waterLevel"/>
            </sml:output>
        </sml:OutputList>
    </sml:outputs>
    <!-- ================================================= -->
    <!-- Sensor Location and Orientation -->
    <!-- ================================================= -->
    <sml:position>
        <swe:DataRecord>
            <swe:field name="location">
                <swe:Vector
                    definition="http://sensorml.com/ont/swe/property/SensorLocation"
                    referenceFrame="http://www.opengis.net/def/crs/EPSG/6.7/4979"
                    localFrame="#SENSOR_FRAME">
                    <swe:coordinate name="Lat">
                        <swe:Quantity definition="http://sensorml.com/ont/swe/property/Latitude" axisID="Lat">
                            <swe:uom code="deg"/>
                            <swe:value>52.761955</swe:value>
                        </swe:Quantity>
                    </swe:coordinate>
                    <swe:coordinate name="Lon">
                        <swe:Quantity definition="http://sensorml.com/ont/swe/property/Longitude" axisID="Long">
                            <swe:uom code="deg"/>
                            <swe:value>-0.819494</swe:value>
                        </swe:Quantity>
                    </swe:coordinate>
                    <swe:coordinate name="Alt">
                        <swe:Quantity definition="http://sensorml.com/ont/swe/property/Altitude" axisID="Alt">
                            <swe:uom code="m"/>
                            <swe:value>83</swe:value>
                        </swe:Quantity>
                    </swe:coordinate>
                </swe:Vector>
            </swe:field>
        </swe:DataRecord>
    </sml:position>
</sml:PhysicalComponent>
```

Graphically, this fragment can be summarised as follows:

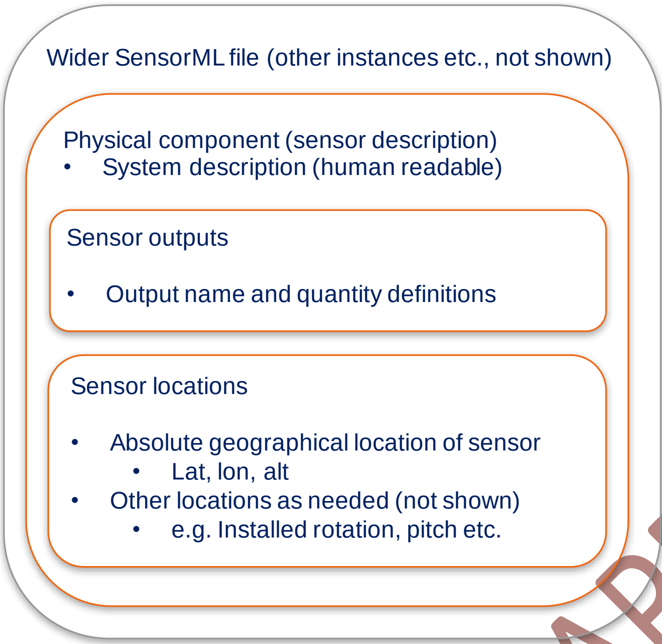


Figure 4.7: Simplified graphical representation for SensorML representation of sensor and location

4.4.1.3. SensorML 2 representation of physical system

A more detailed SensorML 2 representation of a sensor as a physical system is shown below, it is recommended that this type of representation be used for sensors within the In2Rail project, as the additional context captured in this file will make later reuse of the data much more reliable. This description includes details of the physical inputs being monitored, the outputs being produced, the position of the sensor installation, and the physical sensors being used to record the data.

```
<sml:PhysicalSystem gml:id="level_crossing_melton_mowbray_211"
    xmlns:sml="http://www.opengis.net/sensorML/2.0"
    xmlns:swe="http://www.opengis.net/swe/2.0"
    xmlns:gml="http://www.opengis.net/gml/3.2"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xsi:schemaLocation="http://www.opengis.net/sensorML/2.0
                    http://schemas.opengis.net/sensorml/2.0/sensorML.xsd">
    <!-- ================================================= -->
    <!-- System Description -->
    <!-- ================================================= -->
    <gml:description> Melton Mowbray level crossing </gml:description>
    <gml:identifier codeSpace="uid">urn:x-in2rail:level_crossing_melton_mowbray_211
    </gml:identifier>
    <!-- ================================================= -->
    <!-- Inputs = Observed Properties -->
    <!-- ================================================= -->
    <sml:inputs>
        <sml:InputList>
            <sml:input name="depth">
                <sml:ObservableProperty
                    definition="http://sweet.jpl.nasa.gov/2.3/propSpaceHeight.owl#Depth"/>
            </sml:input>
        </sml:InputList>
    </sml:inputs>
    <!-- ================================================= -->
    <!-- Outputs = Quantities -->
    <!-- ================================================= -->
    <sml:outputs>
        <sml:OutputList>
            <sml:output name="waterLevel">
                <swe:DataRecord>
                    <swe:field name="depth">
```

```xml
                        <swe:Quantity definition="http://sensorml.com/ont/swe/property/WaterDepth">
                            <swe:label>Depth of water</swe:label>
                            <swe:uom code="mm"/>
                        </swe:Quantity>
                    </swe:field>
                </swe:DataRecord>
            </sml:output>
        </sml:OutputList>
    </sml:outputs>
    <!-- ================================================= -->
    <!-- System Location -->
    <!-- ================================================= -->
    <sml:position>
        <gml:Point gml:id="stationLocation"
            srsName="http://www.opengis.net/def/crs/EPSG/0/4326">
            <gml:coordinates>52.761955 -0.819494</gml:coordinates>
        </gml:Point>
    </sml:position>
    <!-- ================================================= -->
    <!-- System Components -->
    <!-- ================================================= -->
    <sml:components>
        <sml:ComponentList>
            <sml:component name="depthGauge" xlink:title="urn:aquaread:sensors:ap_5000"
                xlink:href="http://in2rail.eu/xml/sensors/aquaread_complete.xml"/>
        </sml:ComponentList>
    </sml:components>
    <!-- =================================================== -->
    <!-- Connections between components and system output -->
    <!-- =================================================== -->
    <sml:connections>
        <sml:ConnectionList>
            <!-- connection between depth gauge's output and system's waterLevel output -->
            <sml:connection>
                <sml:Link>
                    <sml:source ref="components/depthGauge/outputs/depth"/>
                    <sml:destination ref="outputs/waterLevel/depth"/>
                </sml:Link>
            </sml:connection>
        </sml:ConnectionList>
    </sml:connections>
</sml:PhysicalSystem>
```

Graphically, this fragment can be summarised as follows:

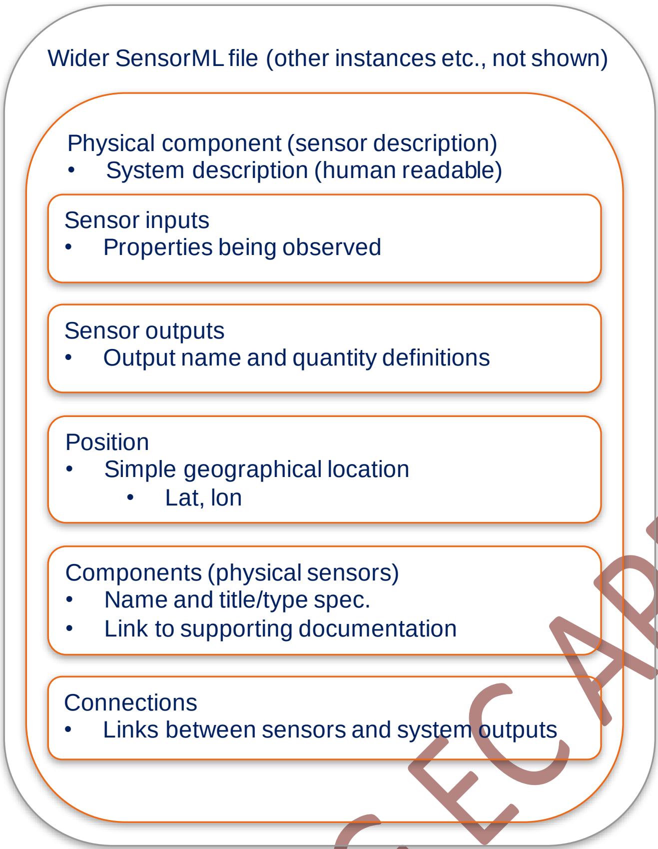


Figure 4.8: Simplified representation of SensorML fragment describing relationship of sensors within physical system

4.4.1.4. SensorML 2 dynamic data

The fragment below is intentionally similar to that described in section 4.4.1.3, however this time the details of the endpoints for the RESTful web services where the sensor data can be obtained are included in the file. The records are defined as pairs of sampling time, water depth values, delivered as a time series, and encoded as a string with a provided format (see section 4.4.1.4.1 below).

```
<sml:outputs>
    <sml:OutputList>
        <sml:output name="waterLevel">
            <sml:DataInterface>
                <!-- data description -->
                <sml:data>
                    <swe:DataStream>
                        <swe:elementType name="waterLevelStream">
                            <swe:DataRecord
                                definition="http://sensorml.com/ont/swe/property/TimeSeries">
                                <swe:label>Floodwater Depth Measurement</swe:label>
                                <swe:field name="time">
                                    <swe:Time
                                        definition="http://sensorml.com/ont/swe/property/SamplingTime">
                                        <swe:uom
                                            xlink:href="http://www.opengis.net/def/uom/ISO-
                                                    8601/0/Gregorian"/>
                                    </swe:Time>
                                </swe:field>
                                <swe:field name="depth">
                                    <swe:Quantity
                                        definition="http://sensorml.com/ont/swe/property/WaterDepth">
                                        <swe:uom code="mm"/>
                                    </swe:Quantity>
                                </swe:field>
                            </swe:DataRecord>
```

```
                        </swe:elementType>
                        <!-- encoding description -->
                        <swe:encoding>
                            <swe:TextEncoding tokenSeparator="," blockSeparator=" "/>
                        </swe:encoding>
                        <!-- reference the values at a RESTful resource -->
                        <!-- e.g. returns latest measurement(s) -->
                        <swe:values xlink:href="http://in2rail.eu:4563/sensor/02080"/>
                    </swe:DataStream>
                </sml:data>
            </sml:DataInterface>
        </sml:output>
    </sml:OutputList>
</sml:outputs>
```

Graphically, this fragment can be summarised as follows:

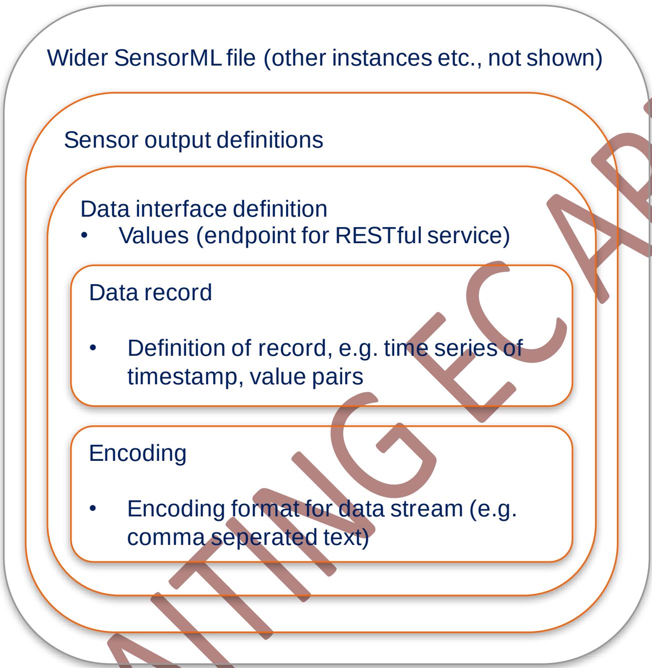


Figure 4.9: Simplified representation of SensorML fragment showing endpoint for access to data values, and specification for data encoding

4.4.1.4.1. A note on dynamic data values

SensorML provides many possible routes for dynamic data from sensors to be accessed. In the example above, the data is accessed via a RESTful webservice interface defined in the sensor description – this decouple method for accessing the data has the dual benefits of enabling the “static” configuration / installation details of the sensors themselves to be described in the XML representation (which itself will be included in the railML representation, see below), and the dynamic data (the values from the sensor) to be encoded differently – in this case the XML specifies a text serialisation – allowing a much less bandwidth-intensive representation to be used in the dynamic context. An example “value” for this text serialisation, as returned by the web service interface, would be:

2016-09-06T05:32:15Z,10.0

Additional values, if returned by the same request at the same time, would be separated by spaces according to the serialisation defined in the example xml, although this can be configured to suit.

4.4.1.5. Combined asset representation

The combined asset status representation for the level crossing is made up from the two sets of xml fragments, describing the structure and configuration of the infrastructure and sensor hardware, and the set of RESTful web services that provide access to the dynamic data values. As before, it is suggested that the two static components of the data (the infrastructure data, and the configuration / installation data for the sensor), which are both stored as XML, are presented as a single instance with the SensorML data included in the railML representation via the xs:anyAttribute element of the levelCrossing element, **note that the use of anyAttribute should be confirmed formally with the railML consortium when railML 3 is released to ensure there is not an alternative mechanism already defined in the new standard**.

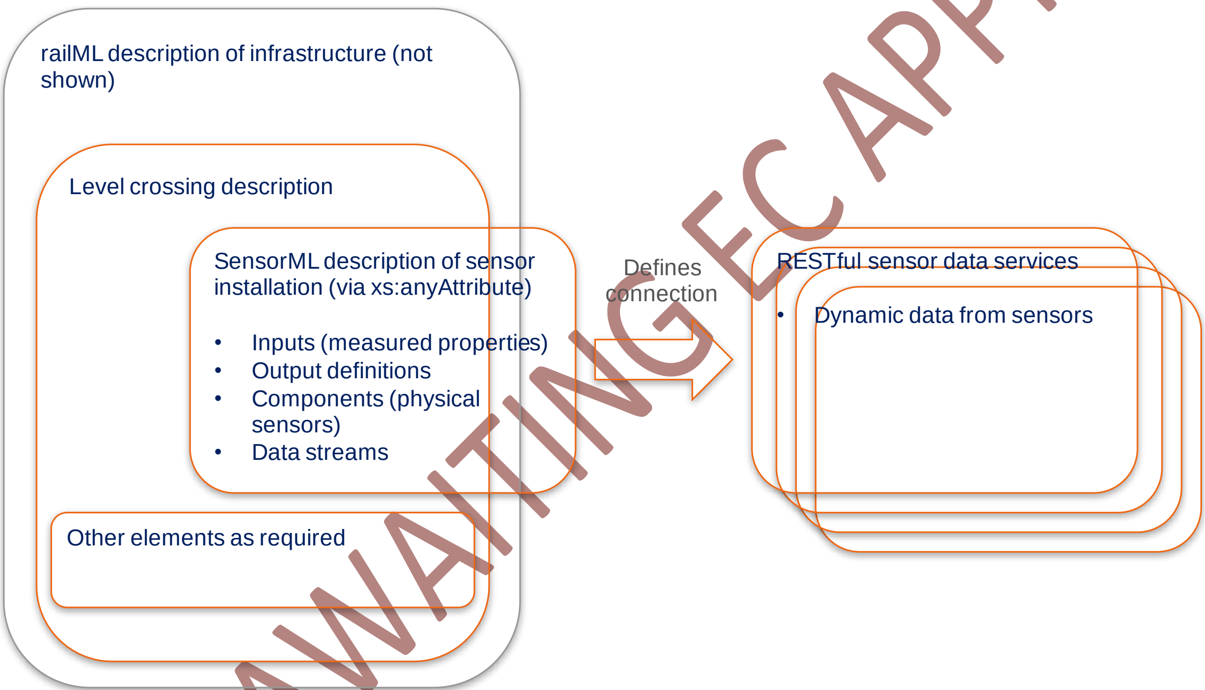


Figure 4.10: Completed level crossing representation with static infrastructure and configuration as XML, and serialised data via RESTful interface

4.4.2. Switch

The switch example is based on the same concepts as the level crossing, and as a result only the infrastructure description and dynamic data description are shown in detail. The example for the switch is more complex than the level crossing as multiple sensors are considered to be monitoring the device, this means the overall monitoring property of the "switch movement" also contains a detailed record of the "motor current".

4.4.2.1. Switch – railML 2 representation of static infrastructure

For the switch, a simple track containing the switch has been defined (t10), with a straight-through connection (t11) and an incoming branch (t12). The switch can be thought of as being in either a "normal" (straight-through from t11 to t10), or a "reverse" (t12 to t10) lie. The layout can be seen in Figure 4.11.

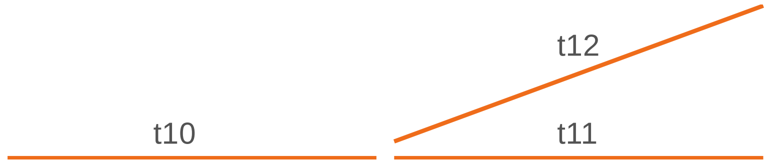


Figure 4.11: Layout of switch

In railML, the layout can be defined as shown below (for further details on the syntax etc. consult the railML wiki – https://wiki.railml.org), with the simplified graphical representation shown in Figure 4.12.

```
<track id="t10">
    <trackTopology>
        <trackBegin id="tt1000">
            <connection id="c1" ref="c2"/>
        </trackBegin>
        <trackEnd id="tt1001"> ... </trackEnd>
        <connections>
            <switch id="s1" pos="0">
                <connection id="c4" ref="c3" orientation="incoming" course="left"/>
                <geoCoord coord="52.520223, -1.679334" epsgCode="urn:ogc:def:crs:EPSG::4326"/>
            </switch>
        </connections>
    </trackTopology>
</track>
<track id="t11">
    <trackTopology>
        <trackBegin id="tt1002">
            <connection id="c2" ref="c1"/>
        </trackBegin>
        <trackEnd id="tt1003"> ... </trackEnd>
    </trackTopology>
</track>
<track id="t12">
    <trackTopology>
        <trackBegin id="tt1004">
            <connection id="c3" ref="c4"/>
        </trackBegin>
        <trackEnd id="tt1005"> ... </trackEnd>
    </trackTopology>
</track>
```

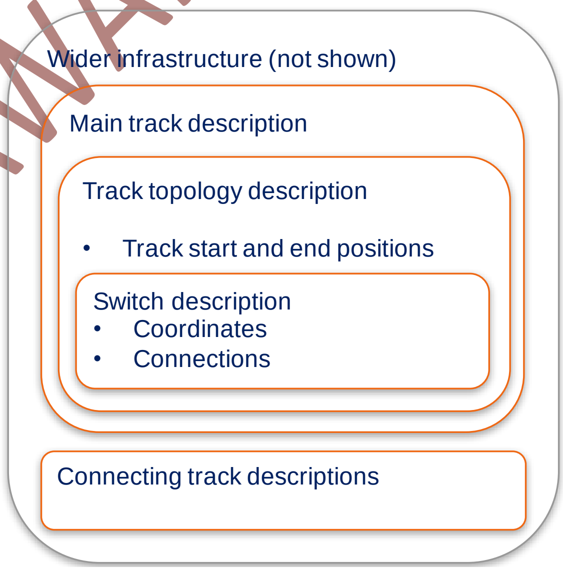


Figure 4.12: Simplified representation of switch

4.4.2.2. SensorML 2 dynamic data

Here, as with the level crossing, the dynamic data are delivered via RESTful web service endpoints, however this time a record containing three fields is defined (to represent the output description in the previous section) – a sampling time, a current waveform representing the motor current during a movement of the switch (accessed via a separate endpoint for that sensor / quantity, and hence with its own encoding), and the final lock direction expressed as a boolean. Like the level crossing, the full SensorML definition will be included with the static infrastructure data and transmitted infrequently – only the values accessed via the web service endpoints are updated on a frequent basis, and these use a simple comma separated text encoding. A graphical representation of the fragment can be seen in Figure 4.13.

```
<sml:outputs>
  <sml:OutputList>
    <sml:output name="switchMovement">
      <sml:DataInterface>
        <!-- data description -->
        <sml:data>
          <swe:DataStream>
            <swe:elementType name="switchMovementStream">
              <swe:DataRecord
                definition="http://sensorml.com/ont/swe/property/TimeSeries">
                <swe:label>Measurement of Switch Movement</swe:label>
                <swe:field name="time">
                  <swe:Time
                    definition="http://sensorml.com/ont/swe/property/SamplingTime">
                    <swe:uom
                      xlink:href="http://www.opengis.net/def/uom/ISO-
                        8601/0/Gregorian"/>
                  </swe:Time>
                </swe:field>
                <swe:field name="currentWaveform">
                  <swe:DataArray
                    definition="http://sensorml.com/ont/swe/property/SeriesData">
                    <swe:elementCount>
                      <swe:Count>
                        <swe:value>6000</swe:value>
                      </swe:Count>
                    </swe:elementCount>
                    <swe:elementType name="motorCurrentStream">
                      <swe:DataRecord
                        definition="http://sensorml.com/ont/swe/property/TimeSeries">
                        <swe:label>Motor current measurement</swe:label>
                        <swe:field name="time">
                          <swe:Time
                            definition="http://sensorml.com/ont/swe/property/SamplingTime">
                            <swe:uom
                              xlink:href="http://www.opengis.net/def/uom/ISO-
                                8601/0/Gregorian"/>
                          </swe:Time>
                        </swe:field>
                        <swe:field name="current">
                          <swe:Quantity
                            definition="
                              http://sweet.jpl.nasa.gov/2.3/procPhysical.owl#ElectricCurrent">
                            <swe:uom code="A"/>
                          </swe:Quantity>
                        </swe:field>
                      </swe:DataRecord>
                    </swe:elementType>
                    <swe:EncodedValuesGroup>
                      <swe:encoding>
                        <swe:TextEncoding tokenSeparator="," blockSeparator=" "/>
                      </swe:encoding>
                      <swe:values xlink:href="http://in2rail.eu:4563/sensor/023"/>
                    </swe:EncodedValuesGroup>
```

```
            </swe:field>
            <swe:field name="lockDirection">
              <swe:Boolean
                definition="http://in2rail.eu/ont/swe/property/PointLie">
              </swe:Boolean>
            </swe:field>
          </swe:DataRecord>
        </swe:elementType>
        <!-- encoding description -->
        <swe:encoding>
          <swe:TextEncoding tokenSeparator="," blockSeparator=" "/>
        </swe:encoding>
        <swe:values xlink:href="http://in2rail.eu:4563/sensor/02080"/>
      </swe:DataStream>
    </sml:data>
  </sml:DataInterface>
 </sml:output>
</sml:OutputList>
</sml:outputs>
```

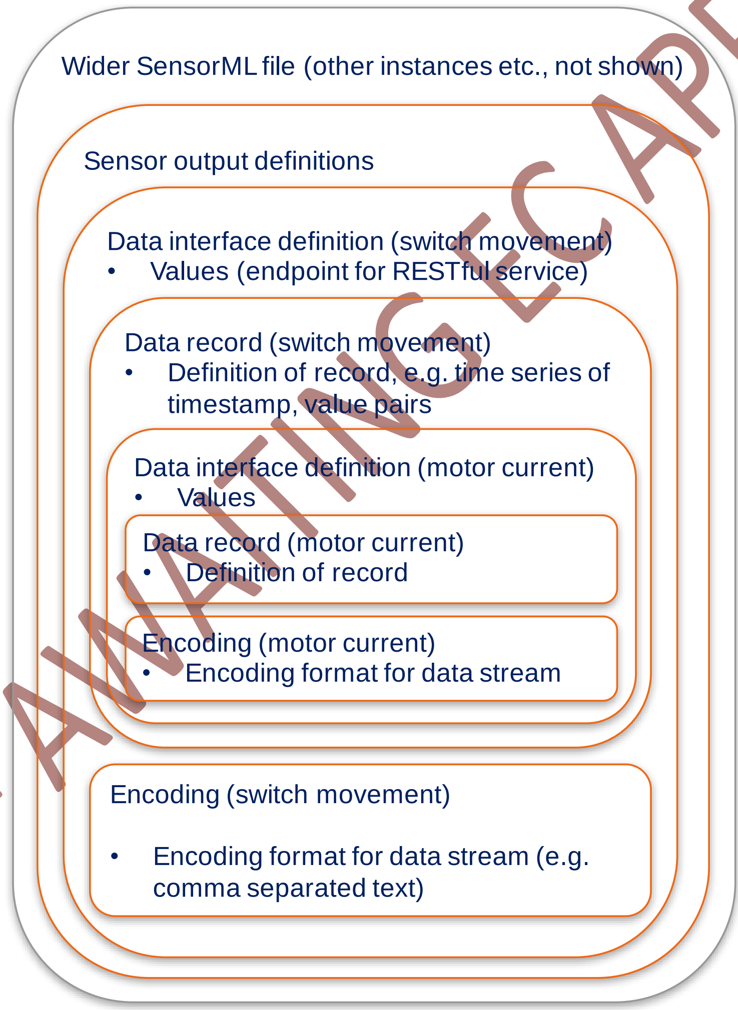


Figure 4.13: Simplified representation of the SensorML data definition for the switch

5. Conclusions

The aim of this report has been to identify the possible attributes used to represent the operational status of a set of railway assets relevant to the TMS, as defined in other work packages within the In2Rail project. For the nine assets described, attributes were classified as either static (attributes with values that never change or change extremely infrequently, and are often related to the type of asset or its installed configuration) or dynamic (attributes with values that change frequently, and are related to the operational state of the asset e.g. the lie of a switch).

Existing models capable of representing the static and dynamic data elements were then reviewed, and with none found to adequately represent both classifications of data independently a hybrid approach is proposed, under which the static elements of the data are described using railML, while the dynamic elements are described using the Open Geospatial Consortium's SensorML model, part of the Sensor Web Enablement suite of standards.

Worked examples were then provided related to two of the infrastructure assets, the level crossing and the switch. These examples show how the details of sensors and sensor configurations can be included in the static infrastructure data using the pre-defined extension element in the railML standard, while the dynamic component of the sensor data is accessed via a web service endpoint (also defined in the static description of the asset).

Obviously, the work done needs further consolidated in Shift2Rail initiatives and to be widely disseminated to all other railway stakeholders to reach consensus and start a possible standardisation activity.

6. References

[1] Capacity 4 Rail Consortium (2016). Deliverable 3.4.1 - Data notation and modelling. Available from http://www.capacity4rail.eu/IMG/pdf/c4r_-_d341_-_data_notation_and_modelling_-_public.pdf, last accessed September 5th, 2016.

[2] European Railway Agency (2010). Rail System Register of Infrastructure. Available from http://www.era.europa.eu/Document-Register/Documents/IU-Recommendation%20on%20specification%20of%20RINF-Final%20Report.pdf, last accessed September 5th, 2016.

[3] Thematic Working Group on Transport Networks (2014). D2.8.I.7 Data Specification on Transport Networks – Technical (v3.2). INSPIRE Thematic Working Group Transport Networks

[4] InteGRail Consortium (2011) InteGRail - Intelligent Integration of Railway Systems, Final Report. Available from http://www.integrail.info/documenti/InteGRail_Final_Project_Report.pdf, last accessed September 5th, 2016.

[5] IS0 (2003). ISO 13374: Condition monitoring and diagnostics of machines — Data processing, communication and presentation. Available online from https://www.iso.org/obp/ui/#iso:std:iso:13374:-1:ed-1:v1:en, last accessed September 5th, 2016.

[6] Tutcher, J., Easton, J. M., & Roberts, C. (2015). Enabling Data Integration in the Rail Industry Using RDF and OWL - the RaCoOn Ontology. ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering.

[7] Kolmorgen, V. P. (2014). 2nd UIC RailTopoModel and railML® Conference - Session C : The usage of railML® for the RINF project of ERA. Retrieved September 5th, 2016 from http://documents.railml.org/events/slides/2014-04-08_uic-railtopomodel+railml-conference_railml4rinf.pdf

[8] Botts, M., Percivall, G., Reed, C., & Davidson, J. (October 2006). OGC® sensor web enablement: Overview and high level architecture. In International conference on GeoSensor Networks (pp. 175-190). Springer Berlin Heidelberg.

[9] Compton, M., Barnaghi, P., Bermudez, L., GarcíA-Castro, R., Corcho, O., Cox, S., ... & Huang, V. (2012). The SSN ontology of the W3C semantic sensor network incubator group. Web Semantics: Science, Services and Agents on the World Wide Web, 17, 25-32.

[10] Raskin, R., & Pan, M. (2003, October). Semantic web for earth and environmental terminology (sweet). In Proc. of the Workshop on Semantic Web Technologies for Searching and Retrieving Scientific Data.

[11] http://peer.berkeley.edu/publications/peer_reports/reports_2003/0316_appCD/papers/NCEE7psdm3.pdf

[12] http://civildigital.com/things-you-didnt-know-chaotianmen-bridge-world-record-steel-arch-bridge/

[13] McCann, D.M. et al. "Radar measurement of bridge scour" 1st International Conference on Railway Engineering, London, 1998.

[14] UIC "Assessment, Reliability and Maintenance of Masonry Arch Bridges: Summary Report" Project I/03/U/285, October 2003.

[15] EU FP6 Sustainable Bridges Project "Condition assessment and inspection of steel railway bridges, including stress measurements in riveted, bolted and welded structures: Background document SB3.4", November 2007.

[16] Gale, J. "An extensive structural monitoring system for Blackfriars railway bridge, London" 9th International Conference on Railway Engineering, London, June 2007.

[17] Fletcher, G. / British Rail Research "Tunnel Integrity Monitoring Systems: Interim Report", October 1992.

[18] Gunn et al. "moisture monitoring in clay embankments using electrical resistivity tomography", 12th International Conference on Railway Engineering, London, July 2013
[19] Higgins, M., and Gallop, M. "Railway embankment instability, inspection, remediation and asset management", 1st International Conference on Railway Engineering, London, 1998.
[20] Briggs, K. "Charing Embankment: Climate change impacts on embankment hydrology". Ground Engineering, June 2010.railML.org Wiki: The asset status representation use case. In: https://wiki.railml.org/index.php?title=User:Ferri_Leberl/IS:UC:Asset_status_representation; last access: 12.10.2016
[21] railML.org: News July 2016. In: https://www.railml.org/en/public-relations/news/news-reader/railml-3-roadmap.html; last access: 12.10.2016
[22] In2Rail Consortium (2016). Deliverable D9.2 "Data/Information Management System architecture design".