TCCS SD2 Summary on General Diagnostic Concept

Author(s)	Karl-Albrecht Klinge , Ralph R Müller

Abstract	A summary document on the progress and discussion issues on the General Diagnosis Concept

Classification	public
Status	Open
Version	1.0
Revision	352574
Last Change Date	09.09.2024

INFO: History table is not displayed, because this document is in status **doc_open**.
RULE: History table is not displayed, in statuses: { draft doc_open doc_inprogress doc_contentApproval doc_contentDecision }
CONTACT: For more information contact Administrator

Content

Contents

Generic diagnostics concept – summary document on data categorisation

1 Introduction

Generally, diagnostics do not contribute to the safety of technical systems. They contribute to RAM, and there is a relation between reliability and safety. Nevertheless, functional safety proof for technical systems shall not be based on diagnostics.
Diagnostics in the railway system can be understood in various different perspectives:

- Physical scope:
  - Insight into technical systems and their components
  - Perception of a technical system in its environment
  - Interaction between technical systems resulting in awareness about the technical state

For diagnostics we can differentiate four key areas:

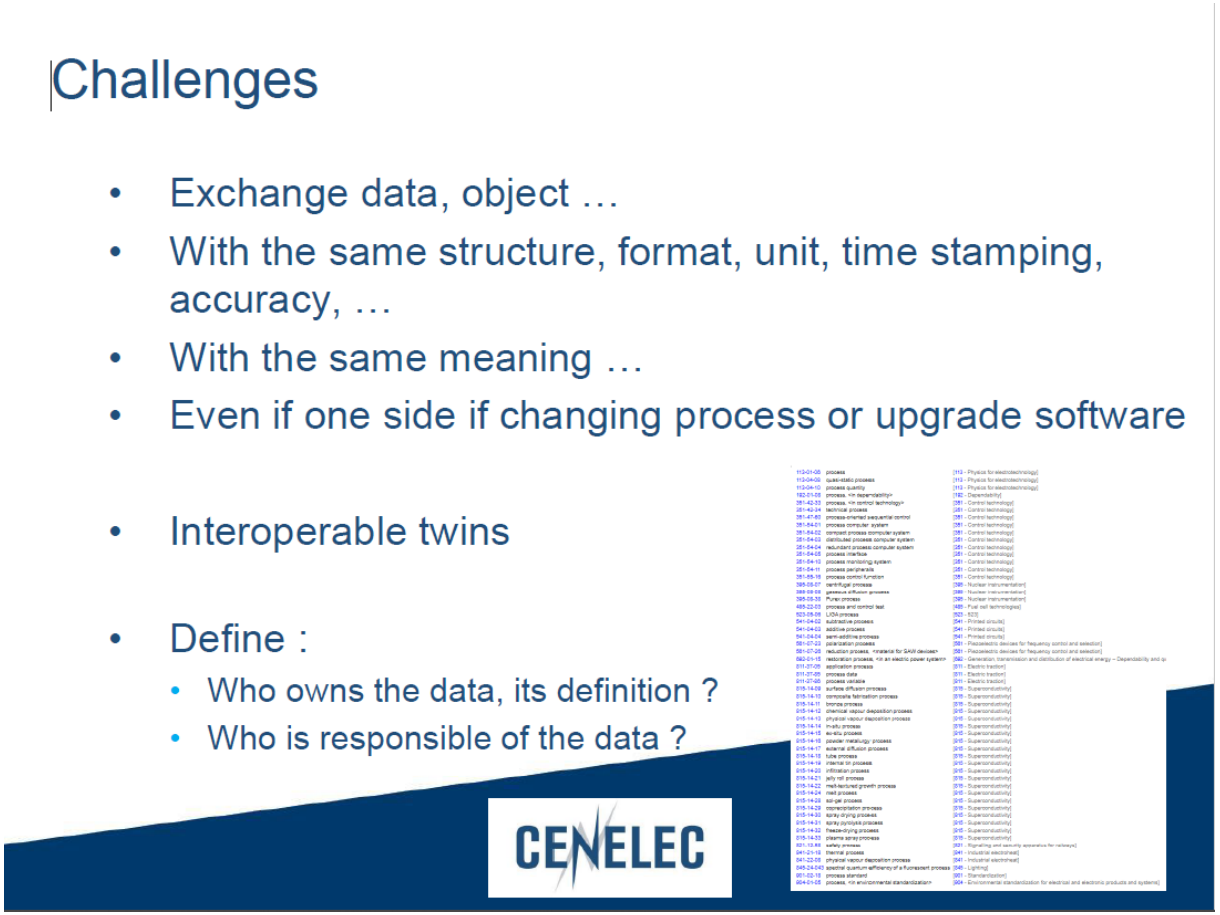
  i. Trackside > is monitoring > Trackside
  ii. Trackside > is monitoring > Vehicle (==> Checkpoints)
  iii. Vehicle > is monitoring > Vehicle
  iv. Vehicle > is monitoring > Trackside (==> Continuous Track Monitoring - CTM)

- Data ownership scope:
  - open data
  - protected/ proprietary data
- Data quality scope
  - dynamic (live) data
  - historic (stored) data
  - data requirements (confidence, accuracy, repeatability, ...)
- Data user scope
  - operations

    - maintenance/ asset management
    - product development
- Data analytics scope
    - forecasting / predictability opportunities
    - dependabilities (system vs components)

All these scopes are - at least partially - interconnected and represent various perspectives on diagnostics.**[**Open**]**

Request from System Pillar Steering Group: In its meeting on 29th of May 2024 the EU-RAIL System Pillar Steering Group gave the following task for delivery to the Transverse CCS Domain (TCCS): **"A summary document on the progress and discussion issues on the General Diagnosis Concept to be circulated to the members as a basis for further discussion on the data categorisation."**
The SP STG minutes state: "Members should decide based on the documents provided, after which feedback can be given and focused discussions and workshops can be organised before written approval is sought. If no approval can be obtained by then, the issue will be discussed further at the next SP STG meeting. Keir FITCH elaborated that the discussion should not be about maintaining the three categories, but about which data should be included in which of them." **[**Open**]**



Standardisation bodies view on data exchange needs:
[Content to be approved**]**

Figure: CENELEC TC 9X (Electrical and electronic applications for railways), Survey Group 34 (Digitalization for railways), presentation on 26th of February 2022 (see: https://rails-project.eu/wp-content/uploads/sites/73/2022/03/Cenelec_TC_9X_SG34.pdf **[**Open**]**

This summary document provides an introduction to the fundamentals of Diagnostics and Monitoring in **Chapter 2**. The Operational Analysis and Operational Epics for the railway system, as outlined in the [TCCS SD 2 Deliverable on ORS], have been broken down for implementation. To expedite this process, the implementation principles

described in subsequent chapters have been tested in collaboration with the TACS Domain (EULYNX). **Chapter 3** defines data categorization, focusing on semantics and metadata across the dimensions of Modeling Rule, Data Type, and Data Category.

In **Chapter 4**, the types of references are explained. OPC-UA is discussed in **Chapters 5** (Data Syntax) and **Chapter 6** (Protocols). OPC UA (Open Platform Communications Unified Architecture) is a data exchange standard widely used in industrial automation and communication. It is an Ethernet-based, service-oriented messaging specification that defines various transport, encoding, security, and data models for command, control, and data exchange. OPC UA is an independent standard, not tied to any specific system or manufacturer, and facilitates both PC-to-machine and machine-to-machine communication.

Chapter 7 presents Use Cases for diagnostic data, and **Chapter 8** describes the first reference implementation, specifically highlighting the initial deployment of an MDM at Innotrans 2024. **Chapter 9** delves into the primary use case of Condition-Based Maintenance (CBM), while **Chapter 10** elaborates on Data Access, following the “need-to-know principle.” This guiding principle for all diagnostic purposes is further illustrated in this chapter.

What has been standardized thus far forms the foundation for identifying further needs. Therefore, **Chapter 11** provides an outlook on trackside, vehicle, and general requirements for achieving significant railway system optimization. Optimizing the railway system at a comprehensive level requires cooperation, which is explained and justified in **Chapter 12**. Additionally, the TCCS Domain contributes to ongoing discussions on data ownership in the same chapter, along with an outlook on the next steps to be taken.

Finally, Annex 1 (**Chapter 13**) lists the standardized trackside data points, showcasing the railway sector’s commitment to standardization while also highlighting the challenges and motivation to progress further. This effort extends beyond just vehicles and trains, aiming for intelligent, comprehensive coverage of the Single EU Railway Area. **Chapter 12** concludes with the ambition to evolve through diagnostics and digital modeling towards a digital twin of the railway system and its subsystems. Once a digital twin is available, railway system operations can shift from a reactive approach to a proactive one. This means that decision-making will increasingly rely on simulated scenarios, rather than responding to unexpected events, allowing for well-informed forecasting and the consideration of alternative options.
[Open**]**

2 Diagnostics and Monitoring

System Pillar TCCS provides a generic CCS system solution (including on-board and trackside) for diagnostics and monitoring services. This solution covers the continuous collection and observation of data, at least related to operation, performance, and health of the CCS system. It includes the detection and reporting of failures to provide information that guide the root causes analysis, to support maintenance teams in resolving issues timely. When operationally relevant, operations (TMS) is supported in

taking mitigation measures and update operational planning.

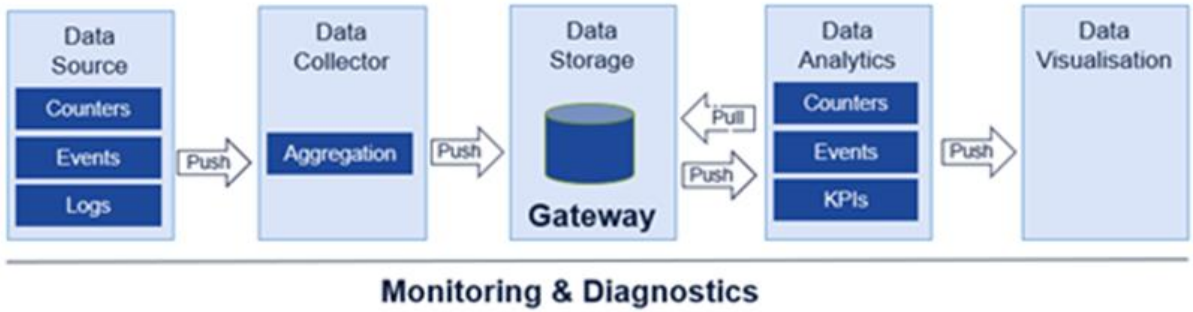


Figure: Generic Diagnostic and Monitoring Data Flow

The collection of the data is ensured over standardized interfaces to enable exchangeability. The collected data of different data sources are stored, and access provided over standard interfaces to different consumers like authorized personnel, maintenance teams, or other technical systems. The diagnostics and monitoring solution provides technical mechanisms considering data sharing capabilities, data confidentiality, data integrity and data availability.**[**Open**]**

All data shall comply with the **ERA-ontology and the derived CCS/TMS Data Model**, including its extensions for diagnostics use cases.**[**Open**]**

3 Data categorization

3.1 Standardization of semantics

For all standardized data it is forbidden to create the same data point using a different attribute name (synonym) and metadata (e.g. unit) or to use an already standardized attribute name with a different semantics any (homonym).
[Open**]**

Non standardized manufacturer specific data points can exist. In case these data points are standardized later the manufacturer must adapt to the standardized data point in the next release. The meaning of datapoints must be disclosed to the operator (magic numbers are forbidden).**[**Open**]**

Already existing standards might be taken into account to define data points, such as **IRS 50405** for rolling stock only.**[**Open**]**

3.2 Metadata

For each datapoint the following metadata must be provided:**[**Open**]**

Modeling Rule	
Mandatory	Mandatory data must be provided by the (sub-)system.
Optional	Optional data can be provided by the (sub-)system. It may be subject to specific contract

[Open**]**

Data Type	**Description**
raw data	Data that is measured. Raw data must include a corresponding unit. It is essential to use SI (International System of Units) units to ensure consistency across all measurements. Raw data is the basis to create prognostic models.
diagnosis	Discrete Values as an enumerator that describe a state of the system, in most cases derived from raw data.
status	Certain diagnostic values are linked to a specific status. The most critical diagnostic value will determine the overall status of the system. The status of individual components is aggregated at higher system levels, taking into account any redundancies within the system.
prognosis	The result of a prognostic model is a prognosis. It is a a prediction of a future status of a system. This prediction helps anticipate potential issues or failures, allowing for proactive maintenance and decision-making to ensure system

	reliability.
restriction	A restriction is the effect of a failure on operation. A point may be fixed in left position - no routes using it in right position can be set. Or a light signal can only show certain signal aspects. The restrictions are provided to the TMS (Traffic Management System).
configuration parameter	A data point that shows the parametrization of the system.

[Open**]**

Data Category	**Description**
Operation	Data point for getting insights in operation. E.g. Turn counters or activeCurrent for points
Equipment	We have developed a model based on a modular principle that enables suppliers to represent their specific architecture. This approach allows for the creation of an online inventory from the field, serving as a “single source of truth.” This model has been already adopted by EULYNX.
Network	We have provided a model to describe the network including e.g. redundant transport channels for the rasta protocol. This model has been already adopted by EULYNX.
Environment	Data describing the environmental conditions, such as air temperature and humidity, can be aggregated from different elements to provide a holistic overview. This approach reduces the need for multiple sensors while still offering comprehensive environmental insights. By correlating environmental conditions with other data, valuable insights can be gained, enhancing the overall analysis and understanding of the system.

[Open**]**

4 References

The railway system contains numerous references. Setting a route requires many field elements to work together, connected by a logical “and.” These references—such as which route requires specific elements, which power supply connects to which object controllers, and which object controller manages particular logical points—are crucial for understanding the causes and propagation of errors.**[**Open**]**

We differentiate the following types of references:

Type	Description
hierarchical references	"HasComponent" reference.
functional references	"provider - consumer" relationships "implements" relationships
redundancy	a model including RedundancyStatus for a component being aggregated in a RedundancyGroup that has a number of RedundancyStatus elements to be available.

[Open**]**

5 Data syntax

In the system pillar, we primarily use JSON syntax for most cases. For diagnostics, we have decided to adopt the OPC UA protocol. This protocol includes a standardized binary syntax that is already optimized for performance.**[**Open**]**

6 Protocols

OPC UA protocol is used for diagnostics.
The system pillar has provided a document for generic OPC UA requirements that have been adopted by TACS Domain (EULYNX).**[**Open**]**

7 Use Cases for diagnostic data

TCCS domain focuses on system-to-collector-interfaces (component to server) for fully open and standardized data (see ch. 3, category 1).

Use case	Description	Interface capability
Status Monitoring and Diagnosis of the CCS system components	Automatically identifying and diagnosing issues within the CCS system components such as switch malfunctions, ETCS-OB incorrect speed command, traction issues or communication interruptions.	Standard format, data structure and protocol for data transfer
Condition Based Maintenance (CBM)	This strategy involves monitoring the actual condition of an asset to determine the appropriate time for maintenance (more details see below).	Standard format, data structure and protocol for data transfer
Predictive Maintenance	Historical data and predictive analytics to forecast potential system failures or issues before they occur, allowing for pre-emptive maintenance actions to avoid service disruption, including equipment gradual deterioration, environmental conditions, operational loads, maintenance records, and failure history	Standard format, data structure and protocol for data transfer
Performance Monitoring	Continuously assessing the performance of the CCS system against predefined benchmarks or KPIs (Key Performance Indicators) to ensure optimal operation	Standard format, data structure and protocol for data transfer
Configuration and Version Control	Monitoring the configuration of the CCS system components, including hardware, software, firmware and parametrisation versions, to ensure compatibility and facilitate seamless updates or modifications	Standard format, data structure and protocol for data transfer
Environmental Condition Monitoring	Collecting and analysing data related to various environmental factors such as temperature, humidity, precipitation, wind speed, and other weather-related conditions along the railway network. The objective is to	Standard format, data structure and protocol for data transfer

	identify conditions that might affect the operation of the CCS system	
Network Interface Monitoring	Monitoring of the data flows across the CCS components in particular application data of the communications interfaces (e.g. SCI interfaces traffic or ETCS-OB messages)	Standard format, data structure and protocol for data transfer
Operational Performance Monitoring to Support TMS	Operational Performance Monitoring to support the Traffic management system focuses on the provision of data related to the performance of trains, track infrastructure, and environmental conditions. The aim is to identify any degradation in performance that could necessitate adjustments to train schedules (timetable) or speed restrictions for example	Standard format, data structure and protocol for data transfer
Support Statistical Analytics	Statistical analysis supported by the diagnostic and Monitoring service based on the collected data from various components of the CCS system	Standard format, data structure and protocol for data transfer
Support Remote Collection of Log Files: only transfer (not content of log files yet)	Remotely collect log files from various components of the CCS system, enabling analysis and troubleshooting without the need for physical access to each component	Data transfer, without standardization of the content (logging mechanism)

Table: Data Types related with use case

[Open**]**

8 Reference implementation: EULYNX BL4 R3

The generic models, data categorizations and generic OPC UA requirements have been taken over by EULYNX in the current baseline.

In the TACS Domain (EULYNX) BL4 R3 the interfaces are described in accordance with the operational requirements, system, and logical concepts. By aligning our efforts, we can create a cohesive framework that supports robust diagnostics, efficient maintenance planning, and optimal operational performance.

This cooperation has already proven to be successful due to the mutual involvement of experts from both the TCCS and TACS Domains in the System Pillar, supported by EULYNX working groups. Their collaboration has facilitated seamless integration and effective communication, ensuring that the development of interfaces is both comprehensive and aligned with industry needs.

Continued cooperation will include work for the TPS/EIL, the MDM and the TMS/restriction model. We will also continue to work on the current releases for trackside assets that are published under trackside-assets-specifications and include a total of 631 data points on the level of CCS field elements - see annex 1.

A first reference implementation of the System Pillar diagnostics concept is provided on Innotrans 2024 by the TCCS Domain, see https://rail-research.europa.eu/calendar/innotrans-2024/ - System Pillar - Transversal.**[**Open**]**

9 Condition Based Maintenance (CBM)

In addition to improving operational performance (TMS), the aim is to enable railways to use **Condition-Based Maintenance (CBM)** as a maintenance strategy. CBM is a data-driven approach to maintenance, allowing organizations to maintain equipment more effectively, improve operational efficiency, and reduce costs.

This strategy involves monitoring the actual condition of an asset to determine the appropriate time for maintenance. Unlike time-based maintenance, which is performed at scheduled intervals regardless of the equipment's condition, CBM focuses on performing maintenance only when it is needed. This approach relies on the real-time diagnostic data gathered.

Key Aspects of CBM:

- **Real-Time Monitoring:** Continuous or periodic monitoring of equipment conditions using sensors and other diagnostic tools.
- **Data Analysis:** Analyzing the collected data to assess the health of the equipment and predict potential failures.
- **Proactive Maintenance:** Maintenance actions are taken based on the actual condition of the equipment, which can help prevent unexpected breakdowns.
- **Reduced Downtime:** By addressing issues only when necessary, CBM can reduce downtime and extend the lifespan of equipment.
- **Cost Efficiency:** CBM can lower maintenance costs by preventing unnecessary maintenance and reducing the likelihood of catastrophic failures.

[Open**]**

10 Data Access - The Need to Know Principle

Ingested data should be made available to every stakeholder that "needs to know" the data.**[**Open**]**

RU/IM data access requirements

The RU/IM would like to operate reliable and safe components.

1. For predictability of possible upcoming operational restrictions in any foreseeable degraded situation to anticipate potential disruptions in service, thus minimizing their impact and improving the overall reliability of the railway system. This data furthermore will support the improvement of maintenance programs on components down to LRU level (spare parts). This will be facilitated with diagnostics information on LRU level (physical view)
2. It would be best if the maintaining entity would have the information to exchange a component before failure (prediction). The RU/IM would like to use different prediction algorithms that are offered on the market. The RU/IM needs to have the possibility to share the data with 3rd party prediction model providers.
3. If a component fails the maintaining entity needs to know its material to replace it.
4. The non-hierarchical provider-consumer relationships must be transparent for the operator to know what effects a failure of one component (provider) might have on other components (consuming a function of the providing component.
5. For transparency of operational restrictions resulting from technical failures to ensure that all relevant stakeholders, including operators, maintenance teams, freight forwarders and passengers, are aware of any limitations or changes in the operational capabilities of the railway system due to technical failures. This should be possible through real-time data feeds.

[Open**]**

Supplier/Integrator data access requirements

The suppliers would like to deliver reliable and safe components.

Along with the product a supplier/integrator should deliver algorithms that are able to predict the behaviour of the product.**[**Open**]**

Data access requirements for the creation of Prediction Models

To create a prediction model we need:

(1) the diagnostics data (including raw data)

and

(2) repair data including the findings and root causes from the field

The diagnostics data (1) is beginning to be standardized. There is currently no standardized distribution mechanism, so that a supplier/integrator could get the data for the components it has delivered.

For (2) there is not yet a standard interface to share the repair data from the maintaining entity to the creators of prediction models.**[**Open**]**

Data ownership
The data is shared data based on the need-to-know described above.
[Open**]**

11 Further needs - an outlook

11.1 Trackside

In TCCS, we have defined the Operational Epics for diagnostics and maintenance from a top-down perspective. Meanwhile, EULYNX is progressing from the bottom up, building on existing concepts. To enhance efficiency, we need to accelerate the integration of these two workstreams. We can begin by testing this approach on the following work items:**[**Open**]**

TPS - Interlocking (SDI-TPS)
We need to gather the operational requirements for TPS. For instance, if a point fails, we must identify the routes that become unavailable so that TMS can calculate the operational performance restrictions. More broadly, we need to obtain the topology of all trackside elements (e.g., which points are in which routes) and derive a restriction model from this information.
[Open**]**

MDM (SDI-DS)
The service function diagnostic of the MDM acts as a namespace aggregator and is a COTS product functioning as an OPC UA gateway server. We need to incorporate some AddIns, including method calls at the MDM level (these method calls will not be propagated to the field elements). These AddIns should support the following:

1. Location of the field element: Reference to the ERA-ontology derived CCS/TMS data model.
2. Maintenance Mode: Allows the maintainer to set the maintenance mode, ensuring that during this time, any technical status will not be propagated to maintenance planning. Reason is to avoid "wrong" diagnostic data that is not caused by real failures but by maintenance inspection and test actions.
3. Estimated Repair Time: Maintenance planning can set the estimated repair time.
4. Foreign Keys to National ERP Systems.
5. Material: Type of Equipment reference.

By implementing these AddIns, we can enhance the functionality and integration of the MDM with various asset management and maintenance processes.
[Open**]**

Interface to TMS - the restriction model

A failure in any component of the railway system can result in an operational restriction. Any operational restriction affecting CCS and TMS information and decisions falls within the System Pillar's scope. Diagnostic systems are used to monitor these failures, while the maintenance planning system (usually part of the Enterprise Resource Planning (ERP) system) contains information on the duration of these failures. By combining diagnostic information with maintenance planning data, we can accurately calculate restrictions that can be utilized in the TMS.

For example, the TMS may be informed that a point is fixed in the left position for the next 6 hours until repair. This situation may lead to an operational restriction, such as a reduced number of available routes and therefore less capacity at an operating location (OP). As a result, trains that do not need to stop at the operating location (OP) may be diverted to an alternative route.

These restrictions should be modeled, preferably considering the results of the shift-to-rail model, to ensure accurate and efficient operational planning and decision-making.

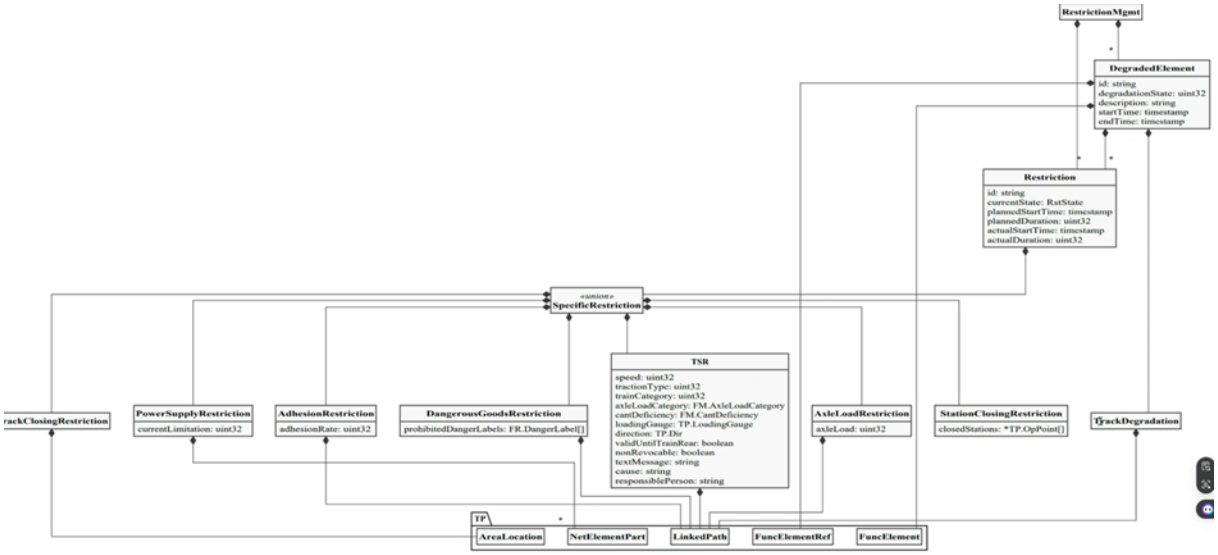


Figure: Restriction model (derived from ERA-ontology and its CCS/TMS Data Model)

The proposed use of the maintenance planning can extend the UC3.1 “Wayside and Infrastructure IAMS for TMS optimization” in EU-RAIL FP3 (IAM4RAIL). Thus, the System Pillar diagnostics (TCCS and HERD) shall be coherent with the standard as produced by FP3, and FP3 shall respect the constraints and prescriptions (e.g. ERA-ontology with its derived CCS/TMS Data Model with diagnostics model extension) provided by System Pillar. This should enable exploiting the full potential of seamless digitization in SERA.**[**Open**]**

11.2 Vehicles

As currently defined, the Subset 149 for the on-board covers only the data defined in Subset 140 (ATO data) and Subset 027 (JRU data) to be collected by the diagnostic and monitoring collector (MDCM). Diagnostic data shall be collected for the following purposes: maintenance of the asset in operation (relevant to RU), planning of train operations at TMS level (relevant to IM).
The extension of the subset 149 is necessary to cover diagnostics data points from the CCS-OB components and additional TCMS data that is relevant for the operational status of the vehicle (e.g., subset 119, subset 139, GPS data, traction system state, brake system state, etc.).
Another important aspect to be covered is the air gap interface (collector OB to collector OFF-B) as defined in the TCCS remit.
[Open**]**

Operational Product Group Set Model
We need to partition a generic model based on functionality of vehicles along with its diagnoses and raw data to monitor effectively. This involves categorizing the model and data into distinct product groups to ensure precise and targeted monitoring.

A good basis to start this work could be the EN 15380-5.
[Open**]**

11.3 Generics

Status Quo of the Physical View - online Inventory - Equipment Model
Diagnostic models have a physical and a logical views.
The physical view is for the maintainer and the asset management:

- maintainer: which replaceable unit has failed, what do I have to take from stock to replace it
- asset management: live online inventory of the "as build" from the field

The logical model (also known as the operational model) examines whether the subsystem can fullfil its operational functions.
The generic model contains parts that are reused in specific models and reduces redundant model definitions that would be difficult to maintain.

The physical view is shown by the equipment model. The equipment model allows the identification of the material (i.e., the type), as well as the serial number and installation location of a failed component remotely. This enables the maintenance technician to bring new equipment of the same type as the failed material to resolve the issue as part of an optimized value chain.
The current challenge lies in the fact that, despite standardized partitioning of the overall system, the specific architecture of subsystems in the field varies by manufacturer. Nevertheless, an operator wants to detect potential deviations, errors, or malfunctions and understand the underlying diagnoses and the significance of hierarchical and functional fault propagations.

The solution is a standardized modular system, referred to here as the equipment model, which allows the manufacturer to map the specific hierarchical and functional architecture of their system down to the smallest replaceable unit.

The material is uniquely specified with the manufacturer and model information. There might also be a foreign key to the Enterprise Resource Planing System (ERP) of the operator. Additionally, for reordering equipment of this material type, the orderCode and orderUrl can be provided. The generic model could be shared between train and trackside.
See Annex A for detailed information.**[**Open**]**

Equipment Model for vehicles
As a quick win we can examine to take over the Equipment Model in vehicles also to know what are the exchangeable parts. This could be based on TSI CCS (listed interoperability constituents) or subset 026.
A cooperation with the TrainCS domain could be useful.**[**Open**]**

New Operational Epics for the online Inventory
We will collect new operational epics that needs to be supported by the diagnostic model, e.g. that we know who is supplier and integrator in order to know which supplier/integrator is getting the data from which component.
[Open**]**

Standardized Interfaces for:
- Data distribution from RU/IM to suppliers/integrators and 3rd party prediction algorithm providers.
- At Runtime: generic interface to include prediction algorithms from different vendors in the live diagnostics data stream. **[**Open**]**

12 Cooperations

On diagnostics for **operational purposes** (see CCS/TMS Data Model, Restriction Domain), TCCS Domain cooperates in the System Pillar with the PRAMS Domain and Task 3 (CMS/ TMS), and in the Innovation Pillar with FP 1 (TMS/ TT).

On diagnostics for **CBM purposes** (see CCS/TMS Data Model, Equipment Models), TCCS Domain cooperates in the System Pillar with Task 1, Task 2 (PRAMS, CE, basically all application defining Domains), Task 5 (HERD), and in the Innovation Pillar with FP 2 and 3.

[Open**]**

The following figure demonstrates that cooperation on diagnostics beyond system boarders has already become a focal point in industry discussions. The optimisation of the entire railway system shall prevail. There is no competitive advantage if one sub-system optimises itself to the disadvantage of other sub-systems.

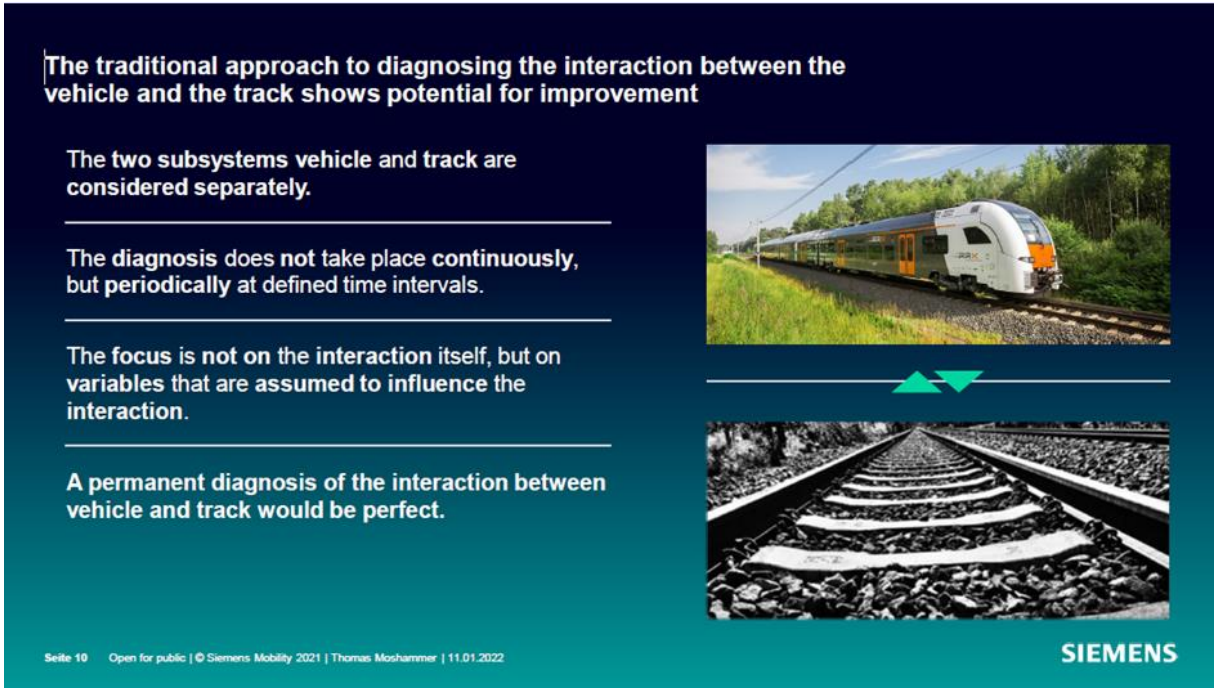


Figure: Request for a holistic approach on diagnostics [source: Siemens]

[Open**]**

Conclusion

There is clear ambition to evolve thanks to diagnostics and digital modelling to a digital twin of the railway system and its sub-systems. Once a digital twin is available, the railway system operation has the benefit to change from "re-act" to "pro-act". Meaning, that the decision making more and more will be based on simulated scenarios rather than mitigating, if not trouble-shooting, to what happened far too often as a surprise rather than a well substantiated forecast with alternative options for considering and selecting.**[**Open**]**

13 Annex 1: Standardised Trackside Data Points

EU-RAIL TCCS Diagnostics for TACS field elements						
Name Specifies the name of the node or variable, providing a unique identifier for each item.	NodeClass Indicates the type of node, which classifies the nature of the item in the model.	DataType Describes the type of data , which determines the format and nature of the data.	ModellingRule Defines rules or constraints for the node, guiding its usage in the model	AttributeType Identifies the type of attribute, categorizing the purpose or function of the data.	Reference Type Specifies the type of relationship or connection the node has with other nodes,	Target Indicates the target node or type, providing context on how different nodes interact or connect.
1. SubsystemType	ObjectType	-	-	-	IsImplemententedBy	EquipmentType
SubsystemIdentification	Variable	String	Mandatory	configuration	-	-
StatusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
StatusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
isTimeSynchronised	Variable	Boolean	Optional	diagnosis	-	-
2. FieldElementType	ObjectType	-	-	-	-	-
OperationStatus	Variable	FieldElementOperationStatus (Enumeration)	Mandatory	diagnosis	-	-
FieldElementSpecificationRevision	Variable	String	Mandatory	configuration	-	-

BasicDataReadable	Variable	BasicDataReadable (Enumeration)	Mandatory	diagnosis	-	-
TrainDetectionSystemType	ObjectType	-	-	-	GeneratesEvent ConnectsTo ConnectsTo	LogEventSubsystemType SCI_TDS_PDI_Type SDI_TDS_Type
GenericIOType	ObjectType	-	-	-	GeneratesEvent ConnectsTo ConnectsTo	LogEventSubsystemType SCI_IO_PDI_Type SDI_IO_Type
LightSignalType	ObjectType	-	-	-	GeneratesEvent ConnectsTo ConnectsTo	LogEventSubsystemType SCI_LS_PDI_Type SDI_LS_Type
PointType	ObjectType	-	-	-	GeneratesEvent GeneratesEvent ConnectsTo ConnectsTo	LogEventSubsystemType PointTurnEventType SCI_P_PDI_Type SDI_P_Type
LevelCrossingType	ObjectType	-	-	-	GeneratesEvent ConnectsTo ConnectsTo	LogEventSubsystemType SCI_LX_PDI_Type SDI_LX_Type

[Open**]**

<AuxillaryInput>	Placeholder Object	-	-	-	-	-
3. InterfaceType	ObjectType	-	-	-	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
connectionStatus	Variable	ConnectionStatus (Enumeration)	Mandatory	diagnosis	-	-
SCI_PDI_Type	ObjectType	-	-	-	-	-
SDI_Type	ObjectType	-	-	-	-	-
SSI_Type	ObjectType	-	-	-	-	-
SMI_Type	ObjectType	-	-	-	-	-
4. SDI_Type	ObjectType	-	-	-	-	-
InterfaceRevisionSDISubsystem	Variable	String	Mandatory	configuration	-	-
InterfaceRevisionSDIGeneric	Variable	String	Mandatory	configuration	-	-

SDI_IO_Type	ObjectType	-	-	-	-	-
SDI_LS_Type	ObjectType	-	-	-	-	-
SDI_IC_Type	ObjectType	-	-	-	-	-
SDI_P_Type	ObjectType	-	-	-	-	-
SDI_TDS_Type	ObjectType	-	-	-	-	-
5. SDI_IO_Type	ObjectType	-	-	-	GeneratesEvent	LogEventInterface

DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
6. SDI_LS_Type	ObjectType	-	-	-	GeneratesEvent	LogEventInterface
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
7. SDI_IC_Type	ObjectType	-	-	-	GeneratesEvent	LogEventInterface
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
8. SDI_P_Type	ObjectType	-	-	-	GeneratesEvent	LogEventInterface
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
9. SDI_TDS_Type	ObjectType	-	-	-	GeneratesEvent	LogEventInterface
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
10. SCI_PDI_Type	ObjectType	-	-	-	-	-
version	Variable	String	Mandatory	configuration	-	-
connectionStatus	Variable	PdiConnectionStatus (Enumeration)	Mandatory	diagnosis	-	-

interfaceRevisionSCISubsystem	Variable	String	Mandatory	configuration	-	-
transportLayer	Variable	TransportLayer (Enumeration)	Mandatory	configuration	-	-
interfaceRevisionSCIGeneric	Variable	String	Mandatory	configuration	-	-
SCI_IO_PDI_Type	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventInterfacePdiEventType LogEventInterfaceType
SCI_LS_PDI_Type	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventInterfacePdiEventType LogEventInterfaceType
SCI_IC_PDI_Type	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventInterfacePdiEventType LogEventInterfaceType

SCI_P_PDI_Type	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventInterfacePdiEventType LogEventInterfaceType
SCI_TDS_PDI_Type	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventInterfacePdiEventType LogEventInterfaceType
11. SCP_Type	ObjectType	-	-	-	-	-
scpConnectionStatus	Variable	ScpConnectionStatus (Enumeration)	Mandatory	diagnosis	-	-
rastaId	Variable	String	Mandatory	configuration	-	-
EC_safety	Variable	Integer (Int32)	Mandatory	counter	-	-
EC_address	Variable	Integer (Int32)	Mandatory	counter	-	-
EC_SN	Variable	Integer (Int32)	Mandatory	counter	-	-
EC_CS	Variable	Integer (Int32)	Mandatory	counter	-	-
EC_type	Variable	Integer (Int32)	Mandatory	counter	-	-
T_seq	Variable	Double	Mandatory	configuration	-	-
N_diagnosis	Variable	Integer (Int32)	Mandatory	configuration	-	-

DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<SCI_PDI>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<TransportChannelRasta>	Placeholder Object	-	OptionalPlaceholder	-	-	-
12. SCI_IO_PDI_Type	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventInterfacePdiEventType LogEventInterfaceType
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
13. SCI_LS_PDI_Type	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventInterfacePdiEventType LogEventInterfaceType

DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
14. SCI_IC_PDI_Type	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventInterfacePdiEventType LogEventInterfaceType
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
15. SCI_P_PDI_Type	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventInterfacePdiEventType LogEventInterfaceType
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
16. SCI_TDS_PDI_Type	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventInterfacePdiEventType LogEventInterfaceType
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
17. SSI_Type	ObjectType	-	-	-	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
InterfaceRevisionSSIGeneric	Variable	String	Mandatory	configuration	-	-
18. SMI_Type	ObjectType	-	-	-	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-

InterfaceRevisionSMIGeneric	Variable	String	Mandatory	configuration	-	-
<ConfigurationItem>	Placeholder Object	-	-	-	-	-
19. ConfigurationItemType	ObjectType	-	-	-	-	-
ActivationState	Variable	ActivationState (Enumeration)	Mandatory	diagnosis	-	-
PreloadState	Variable	PreloadState (Enumeration)	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-

20. TransportChannelType	ObjectType	-	-	-	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
TransportChannelRastaType	ObjectType	-	-	-	-	-
TransportChannelOpcuaType	ObjectType	-	-	-	-	-
<NetworkConfiguration>	Placeholder Object	-	-	-	-	-
21 .TransportChannelRastaType	ObjectType	-	-	-	-	-
status	Variable	TransportChannelRastaStatus (Enumeration)	Mandatory	diagnosis	-	-
N_missed	Variable	Integer (Int32)	Mandatory	raw data	-	-
T_drift	Variable	Double	Mandatory	raw data	-	-
T_drift2	Variable	Double	Mandatory	raw data	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
22. TransportChannelOpcuaType	ObjectType	-	-	-	-	-
		QualifiedName	Mandatory	-	-	

DefaultInstanceBrowseName	Variable					-
23. OpcUaServerType	ObjectType	-	-	-	ServesInterface ServesInterface	SDI_Type SMI_Type
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<SDI>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<TransportChannelOpcua>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<SMI>	Placeholder Object	-	OptionalPlaceholder	-	-	-

24. TlsCertificateType	ObjectType	-	-	-	IsTlsCertificateOf	TransportChannelType
status	Variable	TlsStatus (Enumeration)	Mandatory	diagnosis	-	-
caName	Variable	String	Mandatory	configuration	-	-
errorMessage	Variable	String	Mandatory	diagnosis	-	-
validationMessage	Variable	String	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
25. EquipmentType	ObjectType	-	-	-	IsPartOfRedundancyGroup	RedundancyStatus
isTimeSynchronised	Variable	Boolean	Optional	diagnosis	-	-
hardwareRevision	Variable	String	Mandatory	configuration	-	-
manufacturer	Variable	String	Mandatory	configuration	-	-
manufacturerModel	Variable	String	Mandatory	configuration	-	-
manufacturingDateTime	Variable	DateTime	Mandatory	configuration	-	-

replacementIndication	Variable	EquipmentReplaceabilityStatus (Enumeration)	Mandatory	diagnosis	-	-
serialNumber	Variable	String	Mandatory	configuration	-	-
softwareRevision	Variable	String	Mandatory	configuration	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-

label	Variable	String	Optional	configuration	-	-
<Controller>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<PhysicalAnalogInput>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<PhysicalAnalogOutput>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<PhysicalDigitalInput>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<PhysicalDigitalOutput>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<PhysicalNetworkInterface>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<PowerSupply>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<StorageMediumFlashMemory>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<SubEquipment>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<PhysicalSeparatedOutput>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<PhysicalSeparatedInput>	Placeholder Object	-	OptionalPlaceholder	-	-	-
		-	OptionalPlaceholder	-	-	

<InputSwitch>	Placeholder Object					-
<InputButton>	Placeholder Object	-	OptionalPlaceholder	-	-	-
25. ControllerType	ObjectType	-	-	-	IsPartOfRedundancyGroup	RedundancyStatus
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
systemDescription	Variable	String	Optional	configuration	-	-

operatingSystem	Variable	String	Optional	configuration	-	-
operationStatus	Variable	ControllerOperationStatus (Enumeration)	Mandatory	diagnosis	-	-
coolingFanStatus	Variable	CoolingFanStatus (Enumeration)	Optional	diagnosis	-	-
TemperatureStatus	Variable	TemperatureStatus (Enumeration)	Mandatory	diagnosis	-	-
cpuHealthStatus	Variable	CpuHealthStatus (Enumeration)	Mandatory	diagnosis	-	-
cpuLoadStatus	Variable	CpuLoadStatus (Enumeration)	Mandatory	diagnosis	-	-
label	Variable	String	Optional	configuration	-	-
ramSize	Variable	Integer (UInt64)	Mandatory	configuration	-	-
ramHealthStatus	Variable	RamHealthStatus (Enumeration)	Mandatory	diagnosis	-	-
bootingLastDateTime	Variable	DateTime	Optional	raw data	-	-
bootingLastReason	Variable	ControllerResetReason (Enumeration)	Optional	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-

26. PhysicalNetworkInterfaceType	ObjectType	-	-	-	IsPartOfRedundancyGroup	RedundancyStatus
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
macAddress	Variable	String	Mandatory	configuration	-	-
operationStatus	Variable	PhysicalNetworkInterfaceOperationalStatus (Enumeration)	Mandatory	diagnosis	-	-
nominalBandwidth	Variable	Long (UInt64)	Mandatory	configuration	-	-

DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
label	Variable	String	Optional	configuration	-	-
<NetworkConfiguration>	Placeholder Object	-	OptionalPlaceholder	-	-	-
27. NetworkConfigurationType	ObjectType	-	-	-	-	-
ipv4DefaultGateway	Variable	String	Optional	configuration	-	-
description	Variable	String	Optional	configuration	-	-
hostName	Variable	String	Optional	configuration	-	-
ipv4Address	Variable	String	Mandatory	configuration	-	-
ipv4SubnetMask	Variable	String	Mandatory	configuration	-	-
ipv6Address	Variable	String	Optional	configuration	-	-
ipv6SubnetMask	Variable	String	Optional	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-

ipv6DefaultGateway	Variable	String	Optional	configuration	-	-
PhysicalNetworkInterface	Object	-	Mandatory	-	-	-
28. StorageMediumType	ObjectType	-	-	-	IsPartOfRedundancyGroup	RedundancyStatus
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
memorySize	Variable	Integer (UInt64)	Mandatory	configuration	-	-

temperatureStatus	Variable	TemperatureStatus (Enumeration)	Optional	diagnosis	-	-
label	Variable	String	Optional	configuration	-	-
StorageMediumFlashMemoryType	ObjectType	-	-	-	-	-
29. StorageMediumFlashMemoryType	ObjectType	-	-	-	-	-
wearStatus	Variable	WearStatus (Enumeration)	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
30. PowerSupplyType	ObjectType	-	-	-	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
inputVoltageStatus	Variable	VoltageStatus (Enumeration)	Mandatory	diagnosis	-	-
outputVoltageStatus	Variable	VoltageStatus (Enumeration)	Mandatory	diagnosis	-	-
outputVoltageNominal	Variable	Integer (UInt16)	Mandatory	configuration	-	-
		Integer (UInt16)	Mandatory	diagnosis	-	

outputPower	Variable					-
outputPowerLimit	Variable	Integer (UInt16)	Mandatory	configuration	-	-
label	Variable	String	Optional	configuration	-	-
temperatureStatus	Variable	TemperatureStatus (Enumeration)	Optional	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
inputVoltage	Variable	Real	Mandatory	raw data	-	-

inputCurrent	Variable	Real	Mandatory	raw data	-	-
31. PhysicalOutputType	ObjectType	-	-	-	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
label	Variable	String	Optional	configuration	-	-
PhysicalDigitalOutputType	ObjectType	-	-	-	-	-
PhysicalAnalogOutputType	ObjectType	-	-	-	-	-
PhysicalSeparatedOutputType	ObjectType	-	-	-	-	-
32. PhysicalDigitalOutputType	ObjectType	-	-	-	-	-
physicalOutputValue	Variable	HighLow (Enumeration)	Mandatory	diagnosis	-	-
outputVoltage	Variable	Real	Optional	raw data	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-

outputCurrent	Variable	Real	Optional	raw data	-	-
33. PhysicalAnalogOutputType	ObjectType	-	-	-	-	-
voltage	Variable	Real	Mandatory	raw data	-	-
current	Variable	Real	Mandatory	raw data	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
33. PhysicalSeparatedOutputType	ObjectType	-	-	-	-	-

outputVoltage	Variable	OutputValue (Enumeration)	Mandatory	raw data	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
34. PhysicalInputType	ObjectType	-	-	-	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
label	Variable	String	Optional	configuration	-	-
PhysicalDigitalInputType	ObjectType	-	-	-	-	-
PhysicalAnalogInputType	ObjectType	-	-	-	-	-
PhysicalSeparatedInputType	ObjectType	-	-	-	-	-
35. PhysicalDigitalInputType	ObjectType	-	-	-	-	-
physicalInputValue	Variable	HighLow (Enumeration)	Mandatory	diagnosis	-	-
inputVoltage	Variable	Real	Optional	raw data	-	-

DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
inputCurrent	Variable	Real	Optional	raw data	-	-
36. PhysicalAnalogInputType	ObjectType	-	-	-	-	-
voltage	Variable	Real	Mandatory	raw data	-	-
current	Variable	Real	Mandatory	raw data	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-

37. PhysicalSeparatedInputType	ObjectType	-	-	-	-	-
inputVoltage	Variable	InputValue (Enumeration)	Mandatory	raw data	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
38. InputDeviceType	ObjectType	-	-	-	-	-
label	Variable	String	Optional	configuration	-	-
InputButtonType	ObjectType	-	-	-	-	-
InputSwitchType	ObjectType	-	-	-	-	-
39. InputButtonType	ObjectType	-	-	-	-	-
IsPressed	Variable	Boolean	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
40. InputSwitchType	ObjectType	-	-	-	-	-
InputSwitchPosition	Variable	InputSwitchPosition (Enumeration)	Mandatory	diagnosis	-	-
		QualifiedName	Mandatory	-	-	

DefaultInstanceBrowseName	Variable					-
41. AuxillaryInputType	ObjectType	-	-	-	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<InputButton>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<InputSwitch>	Placeholder Object	-	OptionalPlaceholder	-	-	-
42. RedundancyGroupType	ObjectType	-	-	-	-	-

label	Variable	String	Mandatory	configuration	-	-
isAvailable	Variable	Boolean	Mandatory	diagnosis	-	-
minimumAvailable	Variable	Integer (UInt16)	Mandatory	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
43. RedundancyStatusTypes	ObjectType	-	-	-	IsPartOfRedundancyGroup	RedundancyGroupType
isAvailable	Variable	Boolean	Mandatory	diagnosis	-	-
isExcludedFromRedundancyGroup	Variable	Boolean	Mandatory	diagnosis	-	-
isActive	Variable	Boolean	Mandatory	raw data	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
42. BaseEventType	ObjectType	-	-	-	-	-
LogEventType	ObjectType	-	-	-	-	-
43. LogEventType	ObjectType	-	-	-	-	-

messageId	Variable	Long (UInt64)	Optional	-	-	-
LogEventInterfaceType	ObjectType	-	-	-	-	-
LogEventSubsystemType	ObjectType	-	-	-	-	-
44. LogEventInterfaceType	ObjectType	-	-	-	-	-
LogEventInterfacePdiEventType	ObjectType	-	-	-	-	-
45. LogEventInterfacePdiEventType	ObjectType	-	-	-	-	-

error	Variable	PdiError (Enumeration)	Optional	diagnosis	-	-
eventNotification	Variable	PdiEventNotification (Enumeration)	Mandatory	diagnosis	-	-
46. LogEventSubsystemType	ObjectType	-	-	-	-	-
PointTurnEventType	ObjectType	-	-	-	-	-
LogEventTvpSectionAxleCounterCommandType	ObjectType	-	-	-	-	-
LogEventTvpSectionAxleCounterSweepingFailsType	ObjectType	-	-	-	-	-
LogEventDetectionDeviceLinearTrackCircuitOccupancyType	ObjectType	-	-	-	-	-
LogEventLocalRequestType	ObjectType	-	-	-	-	-
LogEventLocalHandoverType	ObjectType	-	-	-	-	-
BarrierTurnEventType	ObjectType	-	-	-	-	-
47. MotorTurnDataType	ObjectType	BaseDataType	-	-	-	-
index	Variable	String	Mandatory	configuration	-	-

48. MotorTurnData_1AC_Type	ObjectType	BaseDataType	-	-	-	-
voltage	Variable	Real	Mandatory	raw data	-	-
current	Variable	Real	Mandatory	raw data	-	-
cosPhi	Variable	Real	Mandatory	raw data	-	-
49. MotorTurnData_1AC_ActiveCurrentInductiveCompensation_Type	ObjectType	BaseDataType	-	-	-	-
activeCurrent	Variable	Real	Mandatory	raw data	-	-

50. MotorTurnData_1AC_ActiveCurrentPhaseAngleCompensation_Type	ObjectType	BaseDataType	-	-	-	-
activeCurrent	Variable	Real	Mandatory	raw data	-	-
51. MotorTurnData_1AC_Power_Type	ObjectType	BaseDataType	-	-	-	-
power	Variable	Real	Mandatory	raw data	-	-
52. MotorTurnData_3AC_Type	ObjectType	BaseDataType	-	-	-	-
cosPhi_L1	Variable	Real	Mandatory	raw data	-	-
cosPhi_L2	Variable	Real	Mandatory	raw data	-	-
cosPhi_L3	Variable	Real	Mandatory	raw data	-	-
current_L1	Variable	Real	Mandatory	raw data	-	-
current_L2	Variable	Real	Mandatory	raw data	-	-
current_L3	Variable	Real	Mandatory	raw data	-	-
voltage_L1	Variable	Real	Mandatory	raw data	-	-

voltage_L2	Variable	Real	Mandatory	raw data	-	-
voltage_L3	Variable	Real	Mandatory	raw data	-	-
53. MotorTurnData_3AC_Power_Type	ObjectType	BaseDataType	-	-	-	-
power_L1	Variable	Real	Mandatory	raw data	-	-
power_L2	Variable	Real	Mandatory	raw data	-	-
power_L3	Variable	Real	Mandatory	raw data	-	-

54. MotorTurnData_Hydraulic_Type	ObjectType	BaseDataType	-	-	-	-
fluidPressure_left	Variable	Real	Mandatory	raw data	-	-
fluidPressure_right	Variable	Real	Mandatory	raw data	-	-
fluidLevelStatus	Variable	FluidLevelStatus (Enumeration)	Mandatory	diagnosis	-	-
voltageOnEndposition	Variable	Real	Mandatory	raw data	-	-
55. PointType	ObjectType	-	-	-	GeneratesEvent GeneratesEvent ConnectsTo ConnectsTo	LogEventSubsystemType PointTurnEventType SCI_P_PDI_Type SDI_P_Type
aggregateAbleToMoveStatus	Variable	PointAbleToMoveStatus (Enumeration)	Optional	diagnosis	-	-
driveCutoffPrinciple	Variable	PointDriveCutOffPrinciple (Enumeration)	Optional	configuration	-	-
isUsingRedrive	Variable	Boolean	Mandatory	configuration	-	-
lastCommandedPosition	Variable	PointCommandedPosition (Enumeration)	Mandatory	diagnosis	-	-
movementStatus	Variable	PointMovementStatus (Enumeration)	Mandatory	raw data	-	-
pointAbleToMoveStatus	Variable	PointAbleToMoveStatus (Enumeration)	Optional	diagnosis	-	-

position	Variable	PointPosition (Enumeration)	Mandatory	diagnosis	-	-
positionDegraded	Variable	PointPositionDegraded (Enumeration)	Mandatory	diagnosis	-	-
pSamplingInterval	Variable	Real	Mandatory	configuration	-	-
turnCounter	Variable	Long (UInt64)	Optional	counter	-	-
pointOperationTimer	Variable	Real	Mandatory	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-

<PoinMachine>	Placeholder Object	-	MandatoryPlaceholder	-	-	-
56. PointMachineType	ObjectType	-	-	-	-	-
ableToMoveStatus	Variable	PointAbleToMoveStatus (Enumeration)	Optional	diagnosis	-	-
index	Variable	String	Mandatory	configuration	-	-
isCrucial	Variable	Boolean	Mandatory	configuration	-	-
machineType	Variable	PointMachine_Type (Enumeration)	Mandatory	configuration	-	-
position	Variable	PointPosition (Enumeration)	Mandatory	raw data	-	-
timeOffsetStartLeft	Variable	Real	Optional	configuration	-	-
timeOffsetStartRight	Variable	Real	Optional	configuration	-	-
turnCounter	Variable	Long (UInt64)	Mandatory	counter	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<DetectionCircuitLeft>	Placeholder Object	-	OptionalPlaceholder	-	-	-
		-	OptionalPlaceholder	-	-	

<DetectionCircuitRight>	Placeholder Object					-
57. PointTurnEventType	ObjectType	-	-	-	-	-
commandedPosition	Variable	PointCommandedPosition (Enumeration)	Mandatory	diagnosis	-	-
failureReason	Variable	PointTurnFailureReason (Enumeration)	Mandatory	diagnosis	-	-
humidity	Variable	Real	Optional	raw data	-	-
isEndpositionReached	Variable	Boolean	Mandatory	diagnosis	-	-

temperatureAir	Variable	Real	Optional	raw data	-	-
turnTime	Variable	Real	Mandatory	raw data	-	-
MotorTurnData_1AC	Variable	Boolean	Optional	-	-	-
MotorTurnData_1AC_ActiveCurrentInductiveCompensation	Variable	Boolean	Optional	-	-	-
MotorTurnData_1AC_ActiveCurrentPhaseAngleCompensation	Variable	Boolean	Optional	-	-	-
MotorTurnData_1AC_Power	Variable	Boolean	Optional	-	-	-
MotorTurnData_3AC	Variable	Boolean	Optional	-	-	-
MotorTurnData_3AC_Power	Variable	Boolean	Optional	-	-	-
MotorTurnData_Hydraulic	Variable	Boolean	Optional	-	-	-
58. DetectionCircuitLeftType	ObjectType	-	-	-	HasPhysicalReferenceChannel	PhysicalAnalogInputType
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
59. DetectionCircuitRightType	ObjectType	-	-	-	HasPhysicalReferenceChannel	PhysicalAnalogInputType

DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
60. GenericIOType	ObjectType	-	-	-	GeneratesEvent ConnectsTo ConnectsTo	LogEventSubsystemType SCI_IO_PDI_Type SDI_IO_Type
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
label	Variable	String	Optional	configuration	-	-
<LogicalInputChannelSingleChannelType>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<LogicalInputChannelTwoChannelsType>	Placeholder Object	-	OptionalPlaceholder	-	-	-

<LogicalOutputChannelSingleChannelType>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<LogicalOutputChannelSingleChannelType>	Placeholder Object	-	OptionalPlaceholder	-	-	-
61. PhysicalChannelConnectionType	ObjectType	-	-	-	-	-
channelType	Variable	PhysicalChannelTwoChannelsType (Enumeration)	Mandatory	-	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
62. LogicalChannelType	ObjectType	-	-	-	-	-
index	Variable	Integer (UInt64)	Mandatory	configuration	-	-
operationalIdentifierAdjacentSystem	Variable	Byte [20]	Mandatory	configuration	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
LogicalInputChannelType	ObjectType	-	-	-	-	-
LogicalOutputChannelType	ObjectType	-	-	-	-	-

63. LogicalInputChannelType	ObjectType	-	-	-	-	-
logicalInputValue	Variable	LogicalInputValue (Enumeration)	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
LogicalInputChannelSingleChannelType	ObjectType	-	-	-	HasPhysicalReferenceChannel	PhysicalInputType
LogicalInputChannelTwoChannelsType	ObjectType	-	-	-	-	-
64. LogicalInputChannelSingleChannelType	ObjectType	-	-	-	HasPhysicalReferenceChannel	PhysicalInputType

65. LogicalInputChannelTwoChannelsType	ObjectType	-	-	-	-	-
isValenceFailure	Variable	Boolean	Mandatory	diagnosis	-	-
valenceType	Variable	ValenceType (Enumeration)	Mandatory	configuration	-	-
<PhysicalChannelConnection>	Placeholder Object	-	OptionalPlaceholder	-	HasPhysicalReferenceChannel	PhysicalInputType
63. LogicalOutputChannelType	ObjectType	-	-	-	-	-
dutyRatioFixedConfuration	Variable	DutyRatioFixedConfiguration (Enumeration)	Mandatory	configuration	-	-
flashingPeriodFixedConfiguration	Variable	FlashingPeriodFixedConfiguration (Enumeration)	Mandatory	configuration	-	-
logicalOutputValueCommanded	Variable	LogicalOutputValue (Enumeration)	Mandatory	diagnosis	-	-
outputStatus	Variable	OutputDisturbanceStatus (Enumeration)	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
LogicalOutputChannelSingleChannelType	ObjectType	-	-	-	HasPhysicalReferenceChannel	PhysicalOutputType
LogicalOutputChannelTwoChannelsType	ObjectType	-	-	-	-	-

64. LogicalOutputChannelSingleChannelType	ObjectType	-	-	-	HasPhysicalReferenceChannel	PhysicalOutputType
65. LogicalOutputChannelTwoChannelsType	ObjectType	-	-	-	-	-
isValenceFailure	Variable	Boolean	Mandatory	diagnosis	-	-
valenceType	Variable	ValenceType (Enumeration)	Mandatory	configuration	-	-
<PhysicalChannelConnection>	Placeholder Object	-	OptionalPlaceholder	-	HasPhysicalReferenceChannel	PhysicalOutputType
66. LightSignalType	ObjectType	-	-	-	GeneratesEvent ConnectsTo ConnectsTo	LogEventSubsystemType SCI_LS_PDI_Type SDI_LS_Type

isLuminosityChangeable	Variable	Boolean	Mandatory	configuration	-	-
luminosityCommanded	Variable	NightDay (Enumeration)	Mandatory	diagnosis	-	-
luminosityCurrent	Variable	NightDay (Enumeration)	Mandatory	diagnosis	-	-
operationalIdentifier	Variable	Byte [20]	Mandatory	configuration	-	-
signalInformationAdditionalCommanded	Variable	Byte [12]	Mandatory	diagnosis	-	-
signalInformationAdditionalCurrent	Variable	Byte [12]	Mandatory	diagnosis	-	-
signalVectorCommanded	Variable	Byte [6]	Mandatory	diagnosis	-	-
signalVectorCurrent	Variable	Byte [6]	Mandatory	diagnosis	-	-
label	Variable	String	Optional	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<LightPointsMatrix>	Placeholder Object	-	OptionalPlaceholder	-	IsDrivenBy	EquipmentType
<LightPointsSingleLamp>	Placeholder Object	-	OptionalPlaceholder	-	-	-

<LightPointsSingleLEDs>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<LightSignalAdjacentTrainControlElements>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<Indicator>	Placeholder Object	-	OptionalPlaceholder	-	-	-
67. LightSignalAdjacentOutputChannelType	ObjectType	-	-	-	HasPhysicalReferenceChannel	PhysicalOutputType
interfaceConnectionStatus	Variable	InterfaceConnectionStatus (Enumeration)	Mandatory	diagnosis	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-

statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
label	Variable	String	Optional	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
68. LightSignalAdjacentTrainControlElementsType	ObjectType	-	-	-	-	-
trainProtectionName	Variable	String	Optional	configuration	-	-
type	Variable	LightSignalAdjacentTrainControlSystem (Enumeration)	Mandatory	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<LightSignalAdjacentOutputChannel>	Placeholder Object	-	OptionalPlaceholder	-	-	-
69. LightPointType	ObjectType	-	-	-	-	-
counterLightDurationDay	Variable	Long (UInt64)	Mandatory	counter	-	-
counterLightDurationNight	Variable	Long (UInt64)	Mandatory	counter	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-

statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
logicalLightPointStatus	Variable	LogicalLightPointStatus (Enumeration)	Mandatory	diagnosis	-	-
label	Variable	String	Optional	configuration	-	-
LightPointMatrixType	ObjectType	-	-	-	IsDrivenBy	EquipmentType
LightPointSingleLampType	ObjectType	-	-	-	-	-
LightPointSingleLEDType	ObjectType	-	-	-	-	-

70. LightPointMatrixType	ObjectType	-	-	-	-	-
valueCurrent	Variable	String	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
71. LightPointSingleFilamentType	ObjectType	-	-	-	-	-
filamentStatus	Variable	FilamentStatus (Enumeration)	Mandatory	diagnosis	-	-
role	Variable	LampWireRole (Enumeration)	Mandatory	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
72. LightPointSingleLampType	ObjectType	-	-	-	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<LightPointSingleFilament>	Object	LightPointSingleFilamentType	MandatoryPlaceholder	-	-	-
73. LightPointSingleLEDType	ObjectType	-	-	-	-	-
degenerationGrade	Variable	Real	Mandatory	diagnosis	-	-

isElectronicFailure	Variable	Boolean	Mandatory	diagnosis	-	-
numberOfDefectLEDs	Variable	Integer (UInt16)	Optional	raw data	-	-
pNumberOfLEDs	Variable	Integer (UInt16)	Optional	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
74. IndicatorType	ObjectType	-	-	-	HasPhysicalReferenceChannel	PhysicalOutputType
indicatorStatus	Variable	IndicatorStatus (Enumeration)	Mandatory	diagnosis	-	-

label	Variable	String	Optional	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
75. DetectionDeviceType	ObjectType	-	-	-	-	-
isActive	Variable	Boolean	Mandatory	diagnosis	-	-
isInputSignalFailure	Variable	Boolean	Mandatory	diagnosis	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
DetectionDevicePunctiformType	ObjectType	-	-	-	-	-
DetectionDeviceLinearType	ObjectType	-	-	-	-	-
76. DetectionDevicePunctiformType	ObjectType	-	-	-	-	-
index	Variable	Integer (Int16)	Mandatory	configuration	-	-
DetectionDeviceCountingHeadType	ObjectType	-	-	-	-	-

DetectionDevicePunctiformPresenceSensorType	ObjectType	-	-	-	-	-
77. DetectionDeviceCountingHeadType	ObjectType	-	-	-	-	-
counterAxlesAgainstReferenceDirection	Variable	Long (UInt64)	Mandatory	counter	-	-
counterAxlesReferenceDirection	Variable	Long (UInt64)	Mandatory	counter	-	-
counterUndefinedPattern	Variable	Long (UInt64)	Mandatory	counter	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-

DetectionDevicePunctiformCountingHeadMagneticType	ObjectType	-	-	-	-	-
78. DetectionDevicePunctiformCountingHeadMagneticType	ObjectType	-	-	-	-	-
driftStatus	Variable	DriftStatus (Enumeration)	Optional	diagnosis	-	-
isMagneticFieldFailure	Variable	Boolean	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
79. DetectionDevicePunctiformPresenceSensorType	ObjectType	-	-	-	-	-
counterSensorActivation	Variable	Long (UInt64)	Mandatory	counter	-	-
DetectionDevicePunctiformPresenceDetectorMagneticType	ObjectType	-	-	-	-	-
DetectionDevicePunctiformPresenceDetectorLightBarrierType	ObjectType	-	-	-	-	-
80. DetectionDevicePunctiformPresenceDetectorMagneticType	ObjectType	-	-	-	-	-
driftStatus	Variable	DriftStatus (Enumeration)	Mandatory	diagnosis	-	-
isMagneticBuildUpFailure	Variable	Boolean	Mandatory	diagnosis	-	-
		QualifiedName	Mandatory	-	-	

DefaultInstanceBrowseName	Variable					-
81. DetectionDevicePunctiformPresenceDetectorLightBarrierType	ObjectType	-	-	-	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
82. DetectionDeviceLinearType	ObjectType	-	-	-	-	-
isVehicleDetected	Variable	Boolean	Mandatory	diagnosis	-	-
DetectionDeviceLinearLoopType	ObjectType	-	-	-	-	-

DetectionDeviceLinearTrackCircuitType	ObjectType	-	-	-	-	-
83. DetectionDeviceLinearLoopType	ObjectType	-	-	-	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
84. DetectionDeviceLinearTrackCircuitType	ObjectType	-	-	-	-	-
circuitConnectionStatus	Variable	CircuitConnectionStatus (Enumeration)	Mandatory	diagnosis	-	-
qualityOfTransmission	Variable	QualityOfTransmission (Enumeration)	Mandatory	diagnosis	-	-
receiverThreshold	Variable	Real	Mandatory	configuration	-	-
statusReceiver	Variable	StatusReceiver (Enumeration)	Mandatory	diagnosis	-	-
statusTransmitter	Variable	StatusTransmitter (Enumeration)	Mandatory	diagnosis	-	-
unitReceiverThreshold	Variable	UnitReceiverThreshold (Enumeration)	Mandatory	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
85. TrainDetectionSystemType	ObjectType	-	-	-	GeneratesEvent LogEventSubsystemType ConnectsTo SCI_TDS_PDI_Type ConnectsTo SDI_TDS_Type	

DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<TrainDetectionPoint>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<TvpSectionAxleCounter>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<TvpSectionTrackCircuit>	Placeholder Object	-	OptionalPlaceholder	-	-	-
85. TvpSectionType	ObjectType	-	-	-	-	-
isFailureOperational	Variable	Boolean	Mandatory	diagnosis	-	-

isFailureTechnical	Variable	Boolean	Mandatory	diagnosis	-	-
operationalidentifier	Variable	Byte [20]	Mandatory	configuration	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
TvpSectionAxleCounterType	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventTvpSectionAxleCounterCommandType LogEventTvpSectionAxleCounterSweepingFailsType
TvpSectionTrackCircuitType	ObjectType	-	-	-	GeneratesEvent	LogEventDetectionDeviceLinearTrackCircuitOccupancyType
86. TvpSectionAxleCounterType	ObjectType	-	-	-	GeneratesEvent GeneratesEvent	LogEventTvpSectionAxleCounterCommandType LogEventTvpSectionAxleCounterSweepingFailsType
fillingLevelBeforeDrfcOrFc	Variable	Integer (Int16)	Mandatory	diagnosis	-	-
counterDrfcFc	Variable	Long (UInt64)	Mandatory	counter	-	-
fillingLevel	Variable	Integer (Int16)	Mandatory	diagnosis	-	-

abilitytoFCStatus	Variable	AbilityToFCStatus (Enumeration)	Mandatory	diagnosis	-	-
changeTrigger	Variable	ChangeTrigger (Enumeration)	Mandatory	diagnosis	-	-
occupancyStatus	Variable	OccupancyStatus (Enumeration)	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<DetectionDevicePunctiformPresenceDetectorLightBarrier>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<DetectionDevicePunctiformPresenceDetectorMagnetic>	Placeholder Object	-	OptionalPlaceholder	-	-	-

<DetectionDevicePunctiformCountingHeadMagnetic>	Placeholder Object	-	OptionalPlaceholder	-	-	-
87. TvpSectionTrackCircuitType	ObjectType	-	-	-	GeneratesEvent	LogEventDetectionDeviceLinearTrackCircuitOccupancyType
POMFailureStatus	Variable	TvpSectionPomStatus (Enumeration)	Optional	diagnosis	-	-
tvpSectionStatusTrackCircuit	Variable	TvpSectionStatusTrackCircuit (Enumeration)	Mandatory	diagnosis	-	-
powerSupplyStatus	Variable	TvpSectionPowerSupplyStatus (Enumeration)	Mandatory	diagnosis	-	-
sweepingTrainRequired	Variable	SweepingStatus (Enumeration)	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<DetectionDeviceLinearLoop>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<DetectionDeviceLinearTrackCircuit>	Placeholder Object	-	OptionalPlaceholder	-	-	-
88. LogEventTvpSectionAxleCounterCommandType	ObjectType	-	-	-	-	-
commandReceived	Variable	TvpSectionCommandType (Enumeration)	Mandatory	diagnosis	-	-
commandTriggeredByRole	Variable	CommandTriggeredByRole (Enumeration)	Mandatory	diagnosis	-	-

fillingLevel	Variable	Integer (Int16)	Mandatory	diagnosis	-	-
isRejected	Variable	Boolean	Mandatory	diagnosis	-	-
88. LogEventTvpSectionAxleCounterSweepingFailsType	ObjectType	-	-	-	-	-
SweepingFailureReason	Variable	SweepingFailureReason (Enumeration)	Mandatory	diagnosis		
89. LogEventDetectionDeviceLinearTrackCircuitOccupancyType	ObjectType	-	-	-	-	-
measuredReceiverSignal	Variable	Real [*]	Mandatory	raw data	-	-

measuredTransmitterSignal	Variable	Real [*]	Mandatory	raw data	-	-
samplingInterval	Variable	Real	Mandatory	configuration	-	-
unitReceiverSignal	Variable	UnitReceiverSignal (Enumeration)	Mandatory	configuration	-	-
unitTransmitterSignal	Variable	UnitTransmitterSignal (Enumeration)	Mandatory	configuration	-	-
90. TrainDetectionPointType	ObjectType	-	-	-	-	-
operationalidentifier	Variable	Byte [20]	Mandatory	configuration	-	-
directionOfPassing	Variable	DirectionOfPassing (Enumeration)	Mandatory	diagnosis	-	-
passingStatus	Variable	PassingStatus (Enumeration)	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<DetectionDevicePunctiformPresenceDetectorLightBarrier>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<DetectionDevicePunctiformPresenceDetectorMagnetic>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<DetectionDevicePunctiformCountingHeadMagnetic>	Placeholder Object	-	OptionalPlaceholder	-	-	-

91. LevelCrossingType	ObjectType	-	-	-	GeneratesEvent GeneratesEvent GeneratesEvent ConnectsTo ConnectsTo	LogEventSubsystemType LogEventLcalRequestType LogEventLocalHandoverType SCI_IC_PDI_Type SDI_IC_Type
functionalStatus	Variable	FunctionalStatus (Enumeration)	Mandatory	diagnosis	-	-
label	Variable	String	Mandatory	configuration	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-

isClosureTimerOverrun	Variable	Boolean	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<DetectionElement>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<LevelCrossingProtectionFacility>	Placeholder Object	-	OptionalPlaceholder	-	-	-
92. LevelCrossingProtectionFacilityType	ObjectType	-	-	-	-	-
activationCounter	Variable	Long (UInt64)	Mandatory	raw data	-	-
label	Variable	String	Optional	configuration	-	-
protectionfacilityStatus	Variable	ProtectionFacilityStatus (Enumeration)	Mandatory	raw data	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<Barrier>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<ObstacleDetectorOther>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<ObstacleDetectorRadar>	Placeholder Object	-	OptionalPlaceholder	-	-	-

<RoadSignals>	Placeholder Object	-	OptionalPlaceholder	-	-	-
93. BarrierType	ObjectType	-	-	-	GeneratesEvent	BarrierTurnEventType
barrierBoomLightStatus	Variable	BarrierBoomLightsStatus (Enumeration)	Optional	diagnosis	-	-
barrierStatus	Variable	BarrierStatus (Enumeration)	Mandatory	diagnosis	-	-
expectedPosition	Variable	BarrierExpectedPosition (Enumeration)	Mandatory	diagnosis	-	-
isUnexpectedPosition	Variable	Boolean	Mandatory	diagnosis	-	-

label	Variable	String	Optional	configuration	-	-
turnCounter	Variable	Long (UInt64)	Mandatory	counter	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
94. ObstacleDetectorType	ObjectType	-	-	-	-	-
label	Variable	String	Optional	configuration	-	-
obstacleDetectionStatus	Variable	ObstacleDetectorStatus (Enumeration)	Mandatory	diagnosis	-	-
statusTechnical	Variable	StatusTechnical (Enumeration)	Mandatory	diagnosis	-	-
statusTechnicalManufacturerSpecificMessage	Variable	MultiStateDiscreteTypeSupplier (UInt16)	Optional	diagnosis	-	-
ObstacleDetectorRadarType	ObjectType	-	-	-	-	-
ObstacleDetectorOtherType	ObjectType	-	-	-	-	-
95. ObstacleDetectorRadarType	ObjectType	-	-	-	-	-
isHeatingFailure	Variable	Boolean	Mandatory	diagnosis	-	-

isMotorFailure	Variable	Boolean	Mandatory	diagnosis	-	-
isOccupancyDetectionFailure	Variable	Boolean	Mandatory	diagnosis	-	-
isScannerBasicPosition	Variable	Boolean	Mandatory	diagnosis	-	-
isScannerFailure	Variable	Boolean	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
96. ObstacleDetectorOtherType	ObjectType	-	-	-	-	-

DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
97. DetectionElementType	ObjectType	-	-	-	-	-
label	Variable	String	Optional	configuration	-	-
statusDetectionElement	Variable	StatusDetectionElement (Enumeration)	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
98. RoadSignalsType	ObjectType	-	-	-	-	-
label	Variable	String	Optional	configuration	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
<WarningBell>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<WarningLamp>	Placeholder Object	-	OptionalPlaceholder	-	-	-
99. WarningBellType	ObjectType	-	-	-	-	-
label	Variable	String	Optional	configuration	-	-

warningBellStatus	Variable	WarningBellStatus (Enumeration)	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-
100. WarningLampType	ObjectType	-	-	-	-	-
label	Variable	String	Optional	configuration	-	-
warningLampStatus	Variable	WarningLampStatus (Enumeration)	Mandatory	diagnosis	-	-
DefaultInstanceBrowseName	Variable	QualifiedName	Mandatory	-	-	-

<LightPointMatrix>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<LightPointSingleLamp>	Placeholder Object	-	OptionalPlaceholder	-	-	-
<LightPointSingleLED>	Placeholder Object	-	OptionalPlaceholder	-	-	-
101. BarrierTurnEventType	ObjectType	-	-	-	-	-
barrierTurnFailureReason	Variable	BarrierTurnFailureReason (Enumeration)	Mandatory	diagnosis	-	-
expectedPosition	Variable	BarrierExpectedPosition (Enumeration)	Mandatory	diagnosis	-	-
humidity	Variable	Real	Optional	raw data	-	-
isEndpositionReached	Variable	Boolean	Mandatory	diagnosis	-	-
temperatureAir	Variable	Real	Optional	raw data	-	-
turnTime	Variable	Real	Mandatory	raw data	-	-
MotorTurnData_1AC	Variable	Boolean	Optional	-	-	-
MotorTurnData_1AC_ActiveCurrentInductiveCompensation	Variable	Boolean	Optional	-	-	-
MotorTurnData_1AC_ActiveCurrentPhaseAngleCompensation		Boolean	Optional	-	-	

	Variable					-
MotorTurnData_1AC_Power	Variable	Boolean	Optional	-	-	-
MotorTurnData_3AC	Variable	Boolean	Optional	-	-	-
MotorTurnData_3AC_Power	Variable	Boolean	Optional	-	-	-
MotorTurnData_Hydraulic	Variable	Boolean	Optional	-	-	-
102. LogEventLocalRequestType	ObjectType	-	-	-	-	-

isRequestToActivate	Variable	Boolean	Mandatory	diagnosis	-
isRequestToDeactivate	Variable	Boolean	Mandatory	diagnosis	-
handoverIndex	Variable	Integer (Int16)	Mandatory	configuration	-
103. LogEventLocalHandoverType	ObjectType	-	-	-	-
handoverCommandStatus	Variable	HandoverCommandStatus (Enumeration)	Mandatory	diagnosis	-
handoverReactionStatus	Variable	HandoverReactionStatus (Enumeration)	Mandatory	diagnosis	-
requestIndex	Variable	Integer (Int16)	Mandatory	configuration	-

Table 1 : Standardised Data Points

14 Tables

Please update the table of figures.

Please update the table of figures.