EULYNX Initiative

Europe's Rail Joint Undertaking

Interface definition and specification SMI

Document number: Eu.Doc.76
Version: 2.3 (1.A)

Contents

ID	Type	Requirements
Eu.SMI.5	Head	**1 Introduction**
Eu.SMI.155	Head	**1.1 Release information**
Eu.SMI.194	Info	[Eu.Doc.76] Interface definition and specification SMI CENELEC Phase: 5 Version: 2.3 (1.A) Approval date: 02.06.2025
Eu.SMI.206	Info	**Version history**
Eu.SMI.255	Info	version number: 2.0 (0.A) date: 18.05.2022 author: Andreas Strahm, Nico Huurman review: CCB changes: EUAR-508, EUAR-520, EUAR-526, EUAR-528, EUAR-529, EUAR-537
Eu.SMI.259	Info	version number: 2.0 (1.A) date: 27.06.2023 author: Nico Huurman review: TCCS+TACS Mirror Group changes: EUAR-564, EUAR-589, EUAR-594, EUAR-610, EUAR-612, EUAR-613
Eu.SMI.261	Info	version number: 2.1 (0.A) date: 29.04.2024 author: Nico Huurman, Ibtihel Cherif review: cluster changes: EUAR-641, EUAR-645, EUAR-680, EUAR-681, EUAR-687, EUAR-688, EUAR-697, EUAR-712, EUAR-713, EUAR-714, EUAR-715
Eu.SMI.289	Info	version number: 2.2 (0.A) date: 18.06.2024 author: Nico Huurman, Ibtihel Cherif review: TCCS+TACS Mirror Group changes: EUAR-740, EUAR-746, EUAR-747, EUAR-701
Eu.SMI.291	Info	version number: 2.3 (0.A) date: 25.03.2025 author: Nico Huurman, Ricky Holz, Philipp Wolber review: - changes: EUAR-720, EUAR-766, EUAR-774
Eu.SMI.293	Info	version number: 2.3 (1.A) date: 19.06.2025 author: Nico Huurman, Ricky Holz, Philipp Wolber review: TCCS+TACS Mirror Group changes: EUAR-766, EUAR-777, EUAR-778, EUAR-779, EUAR-780, EUAR-782, EUAR-787, EUAR-788, EUAR-790, EUAR-791
Eu.SMI.154	Head	**1.2 Impressum**
Eu.SMI.193	Info	Publishers: **Europe's Rail Joint Undertaking** https://rail-research.europa.eu/ **EULYNX Initiative** https://eulynx.eu/
Eu.SMI.192	Info	Responsible for this document: EU-Rail System Pillar Transversal CCS Components domain

ID	Type	Requirements
Eu.SMI.208	Info	This document is drafted by and belongs to EU Rail. EU Rail encourages the distribution and re-use of this document, the technical specifications and the information it contains. EU Rail holds several intellectual property rights, such as copyright and trade mark rights, which need to be considered when this document is used. EU Rail authorizes you to re-publish, re-use, copy and store this document without changing it, provided that you indicate its source and include the following mention [EU Rail trade mark, title of the document, year of publication, version of document]. EU Rail makes no representation or warranty as to the accuracy or completeness of the information contained within these documents. EU Rail shall have no liability to any party as a result of the use of the information contained herein. EU Rail will have no liability whatsoever for any indirect or consequential loss or damage, and any such liability is expressly excluded. You may study, research, implement, adapt, improve and otherwise use the information, the content and the models in this document for your own purposes. If you decide to publish or disclose any adapted, modified or improved version of this document, any amended implementation or derivative work, then you must indicate that you have modified this document, with a reference to the document name and the terms of use of this document. You may not use EU Rail's trade marks or name in any way that may state or suggest, directly or indirectly, that EU Rail is the author of your adaptations. EU Rail cannot be held responsible for your product, even if you have used this document and its content. It is your responsibility to verify the quality, completeness and the accuracy of the information you use, for your own purposes.
Eu.SMI.153	Head	**1.3 Purpose**
Eu.SMI.191	Info	This document describes the Standard Maintenance Interface for the service functions Loading procedure between the Subsystem - Maintenance and Data Management (MDM) and the EULYNX field element subsystems.
Eu.SMI.190	Info	The service function Loading procedure is designed to provide both safety-critical and non-safety-critical configuration and engineering data, as well as updates of device software.
Eu.SMI.189	Info	This document contains general communication requirements and technical specifications (e.g. protocols and telegram definition) for the SMI (Standard Maintenance Interface) and forms the basis for manufacturer implementation.
Eu.SMI.187	Info	This document does not specify the behaviour of the respective communication partner (e.g. the system reaction in the event of a communication failure), and in particular it does not define when which telegram should be sent. This behaviour is the subject of the Maintenance and data management specification [Eu.Doc.18], the Generic interface and subsystem requirements for SMI [Eu.Doc.120] and further national requirements. Note: In future phases of the System Pillar, national specifications will be replaced by harmonised specifications.
Eu.SMI.186	Info	This document is intended for the following users: • safety authorities • infrastructure managers • safety assessors • signalling system suppliers • validators
Eu.SMI.260	Info	This document is applicable for both the EU-Rail System Pillar target architecture and the EULYNX architecture. The document is delivered as a single specification fitting both the System Pillar documentation sets and the EULYNX documentation sets. EU-Rail System Pillar is the technical authority for this document.
Eu.SMI.152	Head	**1.4 Applicable standards and regulations**
Eu.SMI.184	Info	A list of applicable standards and regulations used in EULYNX is listed in the EULYNX Reference Document List [Eu.Doc.12].
Eu.SMI.205	Info	The references listed in the EULYNX Reference Document List [Eu.Doc.12] shall be considered where they are indicated as being applicable to SMI in the "Applies to" column of the EULYNX Reference Document List [Eu.Doc.12]
Eu.SMI.151	Head	**1.5 Applicable documents**
Eu.SMI.183	Info	The current versions of documents used as input or related to this document are listed in the EULYNX Documentation Plan [Eu.Doc.11]. The relationships between the documents are displayed in the Appendix A1 Documentation plan and structure [Eu.Doc.11_A1].
Eu.SMI.150	Head	**1.6 Appendix**
Eu.SMI.175	Info	- *intentionally left blank* -
Eu.SMI.149	Head	**1.7 Terms and abbreviations**
Eu.SMI.169	Info	The terms and abbreviations are listed in the EULYNX Glossary [Eu.Doc.9].

ID	Type	Requirements
Eu.SMI.209	Head	**1.8 Variability management**
Eu.SMI.210	Info	This document describes harmonised requirements. Variability management is not applicable.
Eu.SMI.147	Head	**1.9 Definition of object types**
Eu.SMI.162	Info	The following definition for object types is applied in this document:
Eu.SMI.161	Info	• "Req" - This denotes a mandatory requirement.
Eu.SMI.158	Info	• "Info" - This denotes additional information to help understand the specification. These objects do not specify any additional requirements.
Eu.SMI.157	Info	• "Head" - This denotes chapter headings.
Eu.SMI.4	Head	**2 Requirements**
Eu.SMI.258	Req	All references to [Eu.Doc.120] refer to Generic interface and subsystem requirements for SMI version 1.2.
Eu.SMI.298	Req	The version number of the OPC UA Information model as described in this document is 1.1.0.
Eu.SMI.141	Head	**2.1 Definition of the SMI**
Eu.SMI.145	Info	The SMI is a message-based interface. It is composed of the transport layer and the application layer.
Eu.SMI.144	Info	The application protocols and the application-related functional requirements associated with it are described in detail in the following chapter titled "Loading procedure".
Eu.SMI.143	Info	The transport layer of the SMI required in line with the application is specified in the chapter titled "Loading procedure".
Eu.SMI.142	Info	The lower layers (network layer, data link layer and physical layer) are defined by the PoS-Signalling [Eu.Doc.100].
Eu.SMI.51	Info	The Standard Maintenance Interface (SMI) is identical for all connected systems in terms of functionality. Note: The Standard Maintenance Interface (SMI) is currently only defined with the EULYNX field element subsystems as connected systems.
Eu.SMI.256	Head	**2.2 Functional requirements**
Eu.SMI.257	Info	The functional requirements for SMI-XX are described in [Eu.Doc.120].
Eu.SMI.3	Head	**3 Loading procedure for Configuration data and Engineering data and for device software**
Eu.SMI.45	Head	**3.1 Overview**
Eu.SMI.212	Info	The service function Loading procedure supports the provision of configuration data and engineering data and of device software to the connected systems.
Eu.SMI.44	Head	**3.2 Standard Maintenance Interface: MDM – EULYNX field element subsystem**
Eu.SMI.294	Info	The EULYNX field element subsystem is the connected system of the Standard Maintenance Interface (SMI).
Eu.SMI.50	Head	**3.2.1 Communication requirements**
Eu.SMI.136	Req	The OPC UA protocol with binary binding via OPC UA Secure Conversation [OPC] via [TCP] shall be used for the Loading procedure.
Eu.SMI.228	Req	OPC UA uses a strict client server model. The server shall run on the connected system. The client shall be contained in the service function Loading procedure.
Eu.SMI.229	Req	The connection shall always be established through the service function Loading procedure.
Eu.SMI.247	Req	The service function Loading procedure may establish the connection as soon as communication on the PoS-Signalling is possible.
Eu.SMI.239	Req	The establishing of the OPC UA connection shall be triggered when the connected system is entering the sub-state "Waiting for data update" within the state "Initialising".

ID	Type	Requirements
Eu.SMI.238	Req	If no connection is available when the connected system expects to interact with the service function Loading procedure, the OPC UA server on the connected system shall trigger the opening of the OPC UA connection by the client via reverse connect. Note: Interaction with the service function Loading procedure is expected when the connected system is entering the sub-state "Waiting for data update" within the state "Initialising".
Eu.SMI.262	Req	In case the service function Loading procedure does not start with the establishment of the OPC UA connection as a reaction to the reverse connect within 5 seconds, the connected system shall resend the reverse connect.
Eu.SMI.296	Req	In case the connected system receives, directly after sending a reverse connect, a connection establishment from the service function Loading procedure that is not a reaction to the sent reverse connect, it shall accept the connection.
Eu.SMI.297	Req	In case the service function Loading procedure receives, directly after sending a connection establishment, a reverse connect from the connected system, it shall ignore the reverse connect.
Eu.SMI.223	Req	If two network channels are used for the service function Loading procedure:
Eu.SMI.138	Req	• the Loading procedure shall always take place via one of the two network channels
Eu.SMI.215	Req	• when the service function Loading procedure does not start with the establishment of the OPC UA connection as a reaction to the reverse connect within 5 second, the connected system shall retry using the other network channel.
Eu.SMI.248	Info	After the event MaintainingFinished the OPC UA connection may be either closed by the client or remains open.
Eu.SMI.286	Req	If no connection is available when the service function Loading procedure expects to interact with the OPC UA server, the service function Loading procedure shall establish the OPC UA connection.
Eu.SMI.230	Req	The communication between the OPC UA client and the OPC UA server shall be session-oriented. For OPC UA, a "telegram" consists of a communication session in which several OPC UA-specific messages are exchanged between the client and the server. The individual messages follow the OPC UA standard [OPC] and are not described here.
Eu.SMI.126	Req	The SubS_ID (see Eu.SAS.77) shall be used to identify the connected system in the service function Loading procedure accordingly.
Eu.SMI.263	Req	The target address(es) and the corresponding communication ports of the OPC UA client for initiating the reverse connect shall be configurable in the connected system. Note: If two network channels are used for the service function Loading procedure, two target addresses need to be configurable.
Eu.SMI.264	Req	The allowed communication ports of the OPC UA server for establishment of the OPC UA connection by the service function Loading procedure shall be configurable in the connected system.
Eu.SMI.265	Head	**3.2.2 General OPC UA requirements**
Eu.SMI.266	Info	An OPC UA NodeClass can be an Object, a Method or a Variable. The Objects have a TypeDefinition that behave similarly to classes in object-oriented programming languages. Through TypeDefintitions and DataTypes predefined by the OPC UA information models, a semantic clarification of the implementation is achieved. The use of the OPC UA information models is needed to make possible the namespace aggregation of field elements.
Eu.SMI.267	Req	OPC UA node IDs of system parts in the OPC UA server of the connected system shall remain unchanged after a reset of the connected system or of the OPC UA server, unless an OPC UA Node ID has been explicitly changed during a configuration update.
Eu.SMI.295	Req	The OPC UA Information model in the connected system shall be derived from the generic OPC UA information models as provided together with the specifications using the "HasSubtype" relationship. The instantiated objects derived from manufacturer-specific types created in this way shall be stored in the folders of the object model provided for this purpose - e.g. the product group model of a point in the "PointProductGroupSet" and the associated equipment model in the "PointEquipmentSet".
Eu.SMI.287	Req	All nodes in the information model that share the same parent must have a unique browse name. Note: This allows the use of simplified browse paths without namespace indices.
Eu.SMI.292	Req	The version number of the OPC UA Information model shall consist of 3 levels, M.m.c., expressing Major, minor and compatible changes in the information model. The upper two levels, M.m., shall be marked in the namespaceURI. All three levels, M.m.c., shall be marked in the OPC UA variable 'version'.
Eu.SMI.268	Req	Changes to the OPC UA Information model structure at runtime shall be communicated to the OPC UA client via model change events, as specified in [OPC].
Eu.SMI.269	Req	The OPC UA server in the connected system shall implement the "Embedded 2017 UA Server Profile". The binary protocol defined in the "Standard 2017 UA Server Profile" is used for communication.
Eu.SMI.270	Req	The following facets shall be implemented in addition:

ID	Type	Requirements
Eu.SMI.271	Req	• Reverse Connect Server Facet
Eu.SMI.272	Req	• Method Server Facet
Eu.SMI.273	Req	The OPC UA server on the connected system shall allow at least 5 subscriptions per session.
Eu.SMI.274	Req	The data contents and further functionality of the OPC UA server on the connected system shall only be provided after the security setup, as defined in [SP-SEC-CommSpec].
Eu.SMI.290	Req	The OPC UA server on the connected system shall respect the security permissions. The following security permissions are used: • eu.rail.smi.configuration-read • eu.rail.smi.configuration-distribute • eu.rail.smi.configuration-activate • eu.rail.smi.component-reset
Eu.SMI.231	Head	**3.2.3 Information model**
Eu.SMI.241	Info	The information model, as defined in [OPC], to be used is shown in the figure below.

ID	Type	Requirements
Eu.SMI.233	Info	Information model Configuration Subsystem1 Item1 AbortUpdateProcess ActivateItem ActivationState CurrentVersion PreloadedVersion PreloadFile PreloadState Item2 AbortUpdateProcess ActivateItem ActivationState CurrentVersion PreloadedVersion PreloadFile PreloadState MaintainingFinished MDMRequestReset MDMSafeMaintenance OperationState RegistrationsReady StartAsyncPreload Subsys_ID UpdateInitState

<table>
<tr><th>ID</th><th>Type</th><th>Requirements</th></tr>
<tr><td>Eu.SMI.244</td><td>Req</td><td>The table below contains clarifications regarding the information model in Eu.SMI.233
<table>
<tr><th>Name</th><th>NodeClass</th><th>Parent</th><th>Additional Information</th><th>Description</th><th>Permission</th></tr>
<tr><td>Subsystem 1,
.. ,
Subsystem n</td><td>Object</td><td>Configuration (global)</td><td></td><td>Contains all Configuration Items (CIs), methods and status
variables used for maintaining a specific EULYNX field element Subsystem.</td><td></td></tr>
<tr><td>Item 1,
...,
Item n</td><td>Object</td><td>Subsystem(n)</td><td></td><td>CIs can contain configuration and engineering data or device software. There can be 1..n CIs.</td><td>Configuration-distribute
Configuration-activate
Configuration-read</td></tr>
</table></td></tr>
<tr><td>Eu.SMI.253</td><td>Req</td><td>The table below contains clarifications regarding the information model in Eu.SMI.233
<table>
<tr><th>Name</th><th>NodeClass</th><th>Parent</th><th>Additional Information</th><th>Description</th><th>Permission</th></tr>
<tr><td>AbortUpdateProcess</td><td>Method</td><td>Item(n)</td><td>Inputs: no input arguments
Outputs: no output arguments</td><td>Abort the currently running update of configuration item n.</td><td>Configuration-distribute
Configuration-activate</td></tr>
<tr><td>ActivateItem</td><td>Method</td><td>Item(n)</td><td>Inputs: no input arguments
Outputs: no output arguments</td><td>Activate the previously transferred configuration item n.
Activation means that it is to be installed at the EULYNX
field element subsystem.</td><td>Configuration-activate</td></tr>
<tr><td>ActivationState</td><td>Variable</td><td>Item(n)</td><td>DataType: Enumeration
Values: NotYetActivatable, ReadyForActivation, Activating, ActivationAborted</td><td>Indicates the activation state of Configuration Item n.</td><td>Configuration-activate
Configuration-read</td></tr>
<tr><td>CurrentVersion</td><td>Variable</td><td>Item(n)</td><td>DataType: String</td><td>Currently applied version of the Configuration Item.
This value is set by the EULYNX field element Subsystem based on the version information read from the installed file.</td><td>Configuration-distribute
Configuration-activate
Configuration-read</td></tr>
</table></td></tr>
<tr><td>Eu.SMI.254</td><td>Req</td><td>The table below contains clarifications regarding the information model in Eu.SMI.233
<table>
<tr><th>Name</th><th>NodeClass</th><th>Parent</th><th>Additional Information</th><th>Description</th><th>Permission</th></tr>
<tr><td>PreloadedVersion</td><td>Variable</td><td>Item(n)</td><td>DataType: String</td><td>Version of a preloaded Configuration Item. Preloaded items can be activated immediately or at a later time if a two-step update procedure is applied.
This value is set by the EULYNX field element Subsystem based on the version information read from the preloaded file.</td><td>Configuration-distribute
Configuration-activate
Configuration-read</td></tr>
<tr><td>PreloadFile</td><td>Object</td><td>Item(n)</td><td>DataType: FileType (OPC UA)</td><td>The preload file for Configuration Item n.
PreloadFile.Open indicates that the MDM has opened the preload file before preloading.
PreloadFile.Write indicates that the MDM loads the preload file to the EULYNX field element Subsystem.
PreloadFile.Close indicates that the MDM has closed the preload file after preloading.</td><td>Configuration-distribute
Configuration-read</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>ID</th><th>Type</th><th>Requirements</th></tr>
<tr><td>Eu.SMI.285</td><td>Req</td><td>The table below contains clarifications regarding the information model in Eu.SMI.233
<table>
<tr><th>Name</th><th>NodeClass</th><th>Parent</th><th>Additional Information</th><th>Description</th><th>Permission</th></tr>
<tr><td>PreloadState</td><td>Variable</td><td>Item(n)</td><td>DataType: Enumeration
Values: NotYetPreloadable, ReadyForPreload, Preloading, PreloadingAborted</td><td>Indicates the preload state of Configuration Item n.</td><td>Configuration-distribute
Configuration-activate
Configuration-read</td></tr>
<tr><td>MaintainingFinished</td><td>Method</td><td>Subsystem(n)</td><td>Inputs: no input arguments
Outputs: no output arguments</td><td>Indicates that the maintenance process has been completed.</td><td>Configuration-activate</td></tr>
<tr><td>MDMRequestReset</td><td>Method</td><td>Subsystem(n)</td><td>Inputs: no input arguments
Outputs: no output arguments</td><td>Request a remote reset of the EULYNX field element
Subsystem.</td><td>Component-reset</td></tr>
<tr><td>MDMSafeMaintenance</td><td>Method</td><td>Subsystem(n)</td><td>Inputs: no input arguments
Outputs: no output arguments</td><td>Perform maintenance after the EULYNX field element
Subsystem was safely released from railway operation.</td><td>Configuration-activate</td></tr>
<tr><td>OperationState</td><td>Variable</td><td>Subsystem(n)</td><td>DataType: Enumeration
Values: NotMaintenance, Maintenance</td><td>Indicates the general operation state of the EULYNX field
element Subsystem in the context of configurability.</td><td>Configuration-distribute
Configuration-activate
Configuration-read</td></tr>
</table></td></tr>
<tr><td>Eu.SMI.288</td><td>Req</td><td>The table below contains clarifications regarding the information model in Eu.SMI.233
<table>
<tr><th>Name</th><th>NodeClass</th><th>Parent</th><th>Additional Information</th><th>Description</th><th>Permission</th></tr>
<tr><td>RegistrationsReady</td><td>Method</td><td>Subsystem(n)</td><td>Inputs: no input arguments
Outputs: no output arguments</td><td>Informs the EULYNX field element Subsystem that the
registration of OPC UA status variables has been finished.</td><td>Configuration-distribute
Configuration-activate</td></tr>
<tr><td>StartAsyncPreload</td><td>Method</td><td>Subsystem(n)</td><td>Inputs: no input arguments
Outputs: no output arguments</td><td>Start a download that can be performed in parallel to the
safe railway operation of an EULYNX field element Subsystem.</td><td>Configuration-distribute</td></tr>
<tr><td>Subsys_ID</td><td>Variable</td><td>Subsystem(n)</td><td>DataType: String</td><td>The unique identifier of the EULYNX field element Subsystem.</td><td>Configuration-distribute
Configuration-activate
Configuration-read</td></tr>
<tr><td>UpdateInitState</td><td>Variable</td><td>Subsystem(n)</td><td>DataType: Boolean</td><td>Indicates that initialisation of PreloadState and ActivationState variables has been finished.
This is the trigger that allows the MDM to iterate over the items and update them as needed.</td><td>Configuration-distribute
Configuration-activate
Configuration-read</td></tr>
</table></td></tr>
<tr><td>Eu.SMI.242</td><td>Req</td><td>Generic events as defined in [Eu.Doc.120] are implemented by the following mechanisms defined in [OPC], as below:</td></tr>
<tr><td>Eu.SMI.232</td><td>Req</td><td>T1in_Maintaining_finished shall be triggered by calling MaintainingFinished</td></tr>
<tr><td>Eu.SMI.234</td><td>Req</td><td>T2in_Preload_item_i_started shall be triggered by calling PreloadFile.Open</td></tr>
<tr><td>Eu.SMI.235</td><td>Req</td><td>T3in_Preload_file_i_close shall be triggered by calling PreloadFile.Close</td></tr>
<tr><td>Eu.SMI.236</td><td>Req</td><td>T4in_Activating_item_i shall be triggered by calling ActivateItem</td></tr>
</table>

ID	Type	Requirements
Eu.SMI.237	Req	T6in_Update_process_aborted shall be triggered by calling AbortUpdateProcess
Eu.SMI.249	Req	T19in_Start_async_preload shall be triggered by calling StartAsyncPreload
Eu.SMI.250	Req	T30in_MDM_Request_Reset shall be triggered by calling MDMRequestReset
Eu.SMI.251	Req	T32in_Registrations_Ready shall be triggered by calling RegistrationsReady
Eu.SMI.252	Req	T36in_MDM_Safe_Maintenance shall be triggered by calling MDMSafeMaintenance
Eu.SMI.42	Head	**3.3 Standard Maintenance Interface: MDM – Subsystem - Electronic Interlocking (SMI-EIL)**
Eu.SMI.46	Info	*- The procedure is not currently specified -* Note: In future phases, this procedure will be based on harmonised specifications published by the Eu-Rail System Pillar.