System Interface Description OPC UA Generic

Abstract	This document is a System Interface Description deliverable which is required per EN 50126-1:2017 - phase 5 for a system under consideration.
Config Item	System Interface Description
Document ID	System Level 5/CONEMP_ Generic OPC UA requirements NEW#902442 System Interface Description OPC UA Generic
Classification	Public
Status	Open (started work of the authors)
Version	0.5
Revision	902442
Last Change Date	2026-05-28

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

SPT2TS-132552 - Approval by reviewers (captured at end of 'In Review by System Pillar') **[** Open **]**

SPT2TS-132563 - Approval by approvers (captured at end of 'In Approval by System Pillar') **[** Open **]**

Table of contents

1 Preamble

1.1 Purpose

This document describes the general OPC UA requirements for the Standard Diagnostic Interface (SDI) and the Standard Maintenance Interface (SMI) between an OPC UA client and an OPC UA server as required per SPPRAMSS-349 - [EN 50126-1:2017] phase 5 (architecture and apportionment of system requirements).

The interface description could be seen as an extension of:

- The System Interface Description SDI, which describes the diagnostic interface between the Service Function Diagnostic (SFD) and the diagnostable Building Block (BB).
- The System Interface Description SMI, which describes the maintenance interface between the Service Function Maintenance (SFC) and the configurable Building Block (BB).

This document serves as foundation for the system interface SDI and SMI providing the generic OPC UA requirements applicable for both interface types, except where explicitly specified

otherwise.

1.2 Intended Audience

ToDo: Add content (to be harmonized with other CONEMP documents -> EET, Ibtihel)

This document is intended for the following users:
Any EU-Rail System Pillar members and volunteer

1.3 Document Context

ToDo: Add content

Scope: Generic OPC UA client / server requirements
-- mapping to functional units (components is done in the specific document

1.4 Glossary

The terms and abbreviations are listed in the table below.

ToDo: Add Terms and ABB (refer to CONEMP Glossary and specific ABB)

No references

2 Interface Requirements OPC UA Generic

This chapter defines requirements for the OPC UA communication between an OPC UA client and an OPC UA server. The requirements are applicable for all OPC UA-based communication interfaces, except where explicitly specified otherwise.

2.1 General

ToDo: Add content
relationship to security:
- SPPRAMSS-2785: SP-SEC-CommSpec - OPC UA Secure Communication Requirements
- SPPRAMSS-1495: SP-SEC-ServSpec - Shared Cybersecurity Services Requirements

configuration specific
- SPPRAMSS-1563: SP-SEC-CompSpec - Software Update Requirements

- SPPRAMSS-1553: SP-SEC-CompSpec - Secure Component Configuration Requirements
- SPPRAMSS-2465: SP-SEC-CompSpec - Device Software Requirements

2.2 Protocol Stack

Eu.SMI.136 - The OPC UA endpoint shall use OPC UA Secure Conversation with binary encoding over TCP (UA-TCP UA-SC UA-Binary) ☐ ~~SPT2TS-130528~~ - *Missing cross-reference*

Eu.SDI.57 - The OPC UA endpoint shall use OPC UA Secure Conversation with binary encoding over TCP (UA-TCP UA-SC UA-Binary). todo : cross reference

The higher layers form the specific functional interface like SMI built on the application-specific OPC UA information model.

[old ID] - The OPC UA endpoint shall use OPC UA Secure Conversation with binary encoding over TCP (UA-TCP UA-SC UA-Binary). *MERGE*

The lower layers of the protocol stack (network layer, data link layer and physical layer) form the communication network interface according to **[Eu.Doc.100]**.

[old ID] - OPC UA uses a strict client server model. The OPC UA server used for SMI communication shall be realized on the configurable Building Block (BB). The OPC UA client shall be realized on the Service Function Maintenance (SFC). *MERGE*

[old ID] - OPC UA uses a strict client server model. The OPC UA server used for SDI communication shall be realized on the diagnostable Building Block (BB). The OPC UA client shall be realized on the Service Function Diagnostic (SFD). *MERGE*

2.3 Capabilities

ToDo: Add content
- Security related OPC UA capabilities for interfaces -> refer to System Pillar Cybersecurity documents

2.3.1 OPC UA server capabilities

2.3.1.1 OPC UA server Application Profile

Eu.SDI.218 - 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.

[old ID] - The OPC UA server in the connected system shall implement the **"Embedded 2017 UA Server Profile"**.

2.3.1.2 OPC UA server Facets

[old ID] - The OPC UA server used for SMI shall implement the **"Method Server Facet"**.

[old ID] - The OPC UA server used for SMI shall implement the **"File Access Server Facet"**.

[old ID] - The OPC UA server shall implement the **"Reverse Connect Server Facet"**.

2.3.1.2.1 Event Access

[old ID] - The OPC UA server used for SDI shall implement the **"Standard Event Subscription Server Facet"**.

Note: This facet specifies the standard support for subscribing to events and is intended to supplement any of the full featured OPC UA profiles.

[old ID] - The OPC UA server shall implement the **"Address Space Notifier Server Facet"**.

Note: This facet enables OPC UA servers to provide structured event notifications based on a hierarchical organization of nodes.

Note: *Although [IEC 62541] provides additional facets for event access, they are not required.*

2.3.1.2.2 Historical Access

[old ID] - The OPC UA server used for SDI shall implement the **"Base Historical Event Server Facet"**.

Note: This facet defines the server requirements to support basic historical event functionality, including simple filtering and general access.

[old ID] - The OPC UA server used for SDI shall implement the **"Historical Data AtTime Server Facet"**.

Note: This facet indicates that the server supports reading data by specifying specific timestamps.

Note: Although [IEC 62541] provides additional facets for historical access, they are not required.

2.3.1.3 OPC UA server Conformance Units

[old ID] - The OPC UA server used for SDI shall implement the conformance unit **"Session Minimum 10 Parallel"**.

The minimum of at least 10 parallel sessions applies to all clients in total.

[old ID] - The OPC UA server shall implement the conformance unit **"Subscription Minimum 05"** (per Session).

[old ID] - The OPC UA server used for SMI shall implement the conformance unit **"Base Info FileType Write"**.

[old ID] - The OPC UA server used for SDI shall implement the conformance unit **"Data Access MultiState"**.

Note: This conformance unit adds support for MultiStateDiscrete Type variables with corresponding properties.

2.3.2 OPC UA client capabilities

2.3.2.1 OPC UA client Application Profile

[old ID] - The OPC UA client shall implement the **"Standard UA Client 2017 Profile"**.

2.3.2.2 OPC UA client Facets

[old ID] - The OPC UA client used for SMI shall implement the **"File Access Client Facet"**.

Note: This facet defines the ability to use file transfer via the defined FileType. This includes reading and optionally writing

[old ID] - The OPC UA client used for SMI shall implement the **"Diagnostic Client Facet"**.

Note: This facet defines the ability to read and process diagnostic information that is part of the OPC UA information model.

[old ID] - The OPC UA client shall implement the **"Reverse Connect Client Facet"**.

Note: This facet defines support of reverse connectivity in a client.

2.3.2.2.1 Event Access

[old ID] - The OPC UA client used for SDI shall implement the **"Event Subscriber Client Facet"**.

Note: This facet defines the ability to subscribe for event notifications. This includes basic AddressSpace concept and the browsing of it, adding events and event filters as monitored items and adding subscriptions.

[old ID] - The OPC UA client used for SDI shall implement the **"Notifier and Source Hierarchy Client Facet"**.

Note: This facet defines the ability to find and use a hierarchy of objects that are event notifier and nodes that are event sources in the server AddressSpace.

Note: Although [IEC 62541] provides additional facets for event access, they are not required.

2.3.2.2.2 Historical Access

[old ID] - The OPC UA client used for SDI shall implement the **"Historical Data AtTime Client Facet"**.

Note: This facet defines the ability to access data at specific instances in time.

[old ID] - The OPC UA client used for SDI shall implement the **"Historical Events Client Facet"**.

Note: This Facet defines the ability to read historical events, including simple filtering.

Note: Although [IEC 62541] provides additional facets for historical access, they are not required.

2.3.2.3 OPC UA client Conformance Units

[old ID] - The OPC UA client used for SMI shall implement the conformance unit **"Base Info Client FileType Write"**.

Note: This conformance unit adds support for the client to access a FileType Object to transfer a file from the client to the server. This includes large files.

[old ID] - The OPC UA client used for SDI shall implement the conformance unit **"Data Access Client MultiState"**.

Note: This conformance unit adds Support to understand MultiState DiscreteType variables with corresponding properties.

2.4 Configuration

The behavior of OPC UA based interfaces depends on different parameters. This section contains some useful parameters and recommended default values. The parameters should be configurable allowing users to adjust according to their needs.

Please note that this list of parameters is not exhaustive. The full parameter set is defined in [IEC 62541].

SecurityToken: The requested lifetime of the OPC UA SecurityToken should be configurable at the OPC UA client. The minimum and maximum supported lifetime of the OPC UA SecurityToken should be configurable at the OPC UA server.

Note: The lifetime is negotiated between OPC UA client and OPC UA server on OpenSecureChannel according to part 4 of [IEC 62541].

Unit: milliseconds

Data Type: UInt32

Recommended requested default: 86,400,000 (24 hours)

SessionTimeout: The requested SessionTimeout should be configurable at the OPC UA client. The maximum supported SessionTimeout should be configurable at the OPC UA server.

Note: The SessionTimeout is negotiated between OPC UA client and OPC UA server on CreateSession according to part 4 of [IEC 62541].

Unit: milliseconds

Data Type: UInt32

Recommended requested default: 3,600,000 (1 hour)

maxResponseMessageSize: The OPC UA maxResponseMessageSize should be configurable at the OPC UA client.

Note: Used by the OPC UA client on CreateSession. According to part 4 of [IEC 62541] the OPC UA client and OPC UA server reduce the chances of a fatal error by exchanging their message size limits in the CreateSession service. This will allow either party to avoid sending a message that causes a communication fault.

Unit: bytes

Data Type: UInt32

Recommended default: 10,485,760 (10 MB)

maxRequestMessageSize: The OPC UA maxRequestMessageSize should be configurable at the OPC UA server.

Note: Used by the OPC UA server on CreateSession. According to part 4 of [IEC 62541] the OPC UA client and OPC UA server reduce the chances of a fatal error by exchanging their message size limits in the CreateSession service. This will allow either party to avoid sending a message that causes a communication fault.

Unit: bytes

Data Type: UInt32

Recommended default: 10,485,760 (10 MB)

LifetimeCount: The requested OPC UA LifetimeCount should be configurable at the OPC UA client. The lifetime count should be a minimum of three times the keep-alive count (according to

[IEC 62541] part 4). The minimum and maximum OPC UA LifetimeCount should be configurable at the OPC UA server.

Note: Negotiated between OPC UA client and OPC UA server on CreateSubscription. According to [IEC 62541] part 4 an OPC UA server should delete a subscription, when the publishing timer has expired this number of times without a publish request being available to send a NotificationMessage.

Data Type: UInt32

Recommended requested default: 3600

Example: With a publishing interval of 1000 ms a LifetimeCount of 3600 leads to life time of a subscription of 1 hour even when the OPC UA client does not send any request messages. So if an OPC UA client reactivates an old session, the subscription will still be available.

PublishingInterval: The requested OPC UA PublishingInterval should be configurable (as a default value for all subscriptions) at the OPC UA client. The OPC UA client should allow optional override of the default value per OPC UA server. The minimum and maximum allowed OPC UA PublishingInterval should be configurable at the OPC UA server.

Note: Negotiated between OPC UA client and OPC UA server on CreateSubscription (according to [IEC 62541] part 4). This interval defines the cyclic rate that the OPC UA subscription is being requested to return notifications to the OPC UA client. This interval is expressed in milliseconds. The negotiated value for this parameter returned in the response is used as the default sampling interval for MonitoredItems assigned to this Subscription.

Unit: milliseconds

Data Type: UInt32

Recommended requested default: 1000 ms

MaxKeepAliveCount: The requested OPC UA MaxKeepAliveCount should be configurable (as a default value for all subscriptions) at the OPC UA client. The OPC UA client should allow optional override of the default value per OPC UA server. The maximum allowed MaxKeepAliveCount should be configurable at the OPC UA server.

Note: Negotiated between OPC UA client and OPC UA server on CreateSubscription. With a publishing interval of 500 ms a MaxKeepAliveCount of 10 leads to a keep-alive message every 5 seconds when no other notification message is sent. The MaxKeepAliveCount value is also used to calculate the lifetime of a subscription. After this time a subscription is closed by the server if there are no messages from the OPC UA client. See also "Subscription model" in part 4 of [IEC 62541].

Data Type: UInt32

Recommended requested default: 10

SamplingInterval: The requested OPC UA SamplingInterval should be configurable (as a default value for all nodes) at the OPC UA client. The OPC UA client should allow optional override of the default value per node. The minimum allowed OPC UA SamplingInterval should be configurable

at the OPC UA server.

Note: Negotiated between OPC UA client and OPC UA server on CreateMonitoredItems. When values change more often than the configured interval, changes will be omitted. The value 0 indicates that the OPC UA server should use the fastest practical rate. The value -1 indicates that the default sampling interval defined by the publishing interval of the subscription is requested. Any negative number is interpreted as -1.

Unit: milliseconds

Data Type: UInt32

Recommended requested default: 1000 ms

SamplingQueueSize: The requested OPC UA SamplingQueueSize should be configurable (as a default value for all nodes) at the OPC UA client. The OPC UA client should allow optional override of the default value per node. The maximum allowed OPC UA SamplingQueueSize should be configurable at the OPC UA server.

Note: Negotiated between OPC UA client and OPC UA server on CreateMonitoredItems according to [IEC 62541].

Data Type: UInt32

Recommended requested default: 10

2.5 Connectivity

2.5.1 Connection Establishment

Basically a connection may be initiated by the OPC UA client or by the OPC UA server when they create a transport connection and establish a communication with their peers. For details refer to "*7.1.3 Establishing a connection*" in part 6 of [IEC 62541].

It is recommended that the endpointURL for the provided OPC UA endpoints of the OPC UA server used for SDI or SMI contains the interface name as a path postfix. The endpointURL should be completely configurable at the OPC UA server.
Example:
- opc.tcp://10.12.25.100:4840/SDI
- opc.tcp://10.12.25.101:4840/SMI

[old ID] - The ApplicationUri of the OPC UA server shall be a valid URL [RFC 1738] or URN [RFC 2141]. The ApplicationUri shall be unique for each OPC UA server.

Note: [IEC 62541] (Part 4, 6.1.3) requires that the ApplicationUri must also be part of the communication certificate (subject alternate name), otherwise the connection attempt will fail with an "Bad_CertificateUriInvalid" error. It could be considered to make the ApplicationUri completely configurable.

Data Type: String

[old ID] - The data contents and further functionality of the OPC UA server shall only be provided

after the security setup according to the system pillar cybersecurity specifications. *MERGE*

[old ID] - An OPC UA client according to [IEC 62541] shall monitor the OPC UA connection for interruptions and degradation.

Note: See also "Re-establishing connections" in part 4 of [IEC 62541].

2.5.2 Redundancy

OPC UA enables OPC UA servers, OPC UA clients and networks to be redundant. For details refer to "*6.6 Redundancy*" in part 4 of [IEC 62541].

The following options are support by the OPC UA specifications:

- **Server Redundancy:**
  Server Redundancy allows OPC UA clients to have multiple sources from which to obtain the same data. Server Redundancy can be achieved in multiple manners, some of which require OPC UA client interaction, others that require no interaction from an OPC UA client.

  There are two general modes of server redundancy, transparent and non-transparent:
  - ***Server Redundancy Transparent****:* The failover of OPC UA server responsibilities from one OPC UA server to another is transparent to the OPC UA client. The OPC UA client does not need to perform any actions to continue to send or receive data.
  - ***Server Redundancy Non-Transparent****:* The Failover from one OPC UA server to another and actions to continue to send or receive data are performed by the OPC UA client. The OPC UA client shall perform the required actions to benefit from the OPC UA server redundancy.

- **Client Redundancy:**
  Client Redundancy allows identically configured OPC UA clients to behave as if they were single OPC UA clients, but not all OPC UA clients are obtaining data at a given time. Ideally there should be no loss of information when an OPCUA vlient failover occurs

- **Network Redundancy:**
  Network Redundancy allows an OPC UA client and an OPC UA server to have multiple communication paths to obtain the same data.

Any combination of the described redundancy options could coexist in a system.

2.5.3 Scalability

ToDo: Add content

2.6 Miscellaneous

2.6.1 Timestamp

[old ID] **-** The OPC UA server used shall use the system time to set the SourceTimeStamp and the ServerTimeStamp in accordance with **[IEC 62541].**

The SourceTimeStamp is used to reflect the timestamp that was applied to a variable value by the data source. This is an end-to-end information. The ServerTimeStamp is used to reflect the time that the OPC UA server received a variable value. This is a hop-to-hop information.

2.6.2 OPC UA information model

2.6.2.1 Version Handling

Eu.SDI.246 - 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'.

[old ID] **-** 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'. *MERGE*

2.6.2.2 OPC UA Node IDs

Eu.SDI.238 - OPC UA node IDs of system parts in the OPC UA server of the Diagnosable Building Block (BB) shall remain unchanged after a reset of the Diagnosable Building Block (BB) or of the OPC UA server, unless an OPC UA Node ID has been explicitly changed during a configuration update.

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 TypeDefinitions and DataTypes predefined by the OPC UA information models, a semantic clarification of the implementation is achieved.

[old ID] - OPC UA node IDs of runtime-static nodes in the OPC UA server shall remain unchanged after a restart of the OPC UA server, unless an OPC UA Node ID has been explicitly changed during a configuration update.

Runtime-static OPC UA nodes are loaded during startup of the OPC UA server and will stay unchanged during the complete runtime of the OPC UA server. This does not mean that corresponding node values are also static.

[old ID] - Two nodes in the information model that share the same parent must not have a browse name that only differs in namespace. *MERGE*

This allows the use of simplified browse paths without namespace indices.

Eu.SDI.237 - 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.

2.6.2.3 OPC UA model changes

Eu.SDI.217 - Changes to the OPC UA Information model structure at runtime shall be communicated to the OPC UA client via model change events.

[old ID] - 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 **[IEC 62541].** *MERGE*

2.6.2.4 Permissions

The OPC UA permissions should follow the pattern definition of the System Pillar Cybersecurity Communication Specification. Refer to SPPRAMSS-8868 OPC UA Permission Pattern.

[old ID] - The OPC UA server on the Building Block (BB) shall respect the security permissions.

Note: The security permissions are defined within the specific interface specifications e.g. SMI.

For SSI interfaces based on OPC UA (SSI-MNT, SSI-BKP) the security permissions are specified in the "System Pillar Cybersecurity Communication Specification". Refer to SPPRAMSS-8867 Security Permissions for SSI interfaces.

2.6.2.5 Subscriptions

OPC UA clients define MonitoredItems as a part of an OPC UA Subscription to subscribe for value updates of data points.

Each MonitoredItem is assigned a sampling interval that is either inherited from the publishing interval of the Subscription or that is defined specifically to override that rate. A negative number indicates that the default sampling interval defined by the publishing interval of the Subscription is requested. The sampling interval indicates the fastest rate at which the Server should sample its underlying source for data changes.

If the server specifies a value for the MinimumSamplingInterval attribute it returns to the OPC UA client a sampling interval that is equal or higher than the MinimumSamplingInterval. The MinimumSamplingInterval can be specified in the OPC UA model.

See also MonitoredItem and Subscription Service Set in part 4 of [IEC62541].

[old ID] - The OPC UA client shall send "Publish Requests" in accordance with to [IEC 62541].

[old ID] - The OPC UA server shall respond to the telegram "Publish Request" in accordance with [IEC 62541].

2.6.2.6 Not supported Methods

2.6.2.6.1 StatusCode

[old ID] - In order to prevent an OPC UA client from calling OPC UA methods which are not supported by the OPC UA server, the OPC UA server shall return an appropriate StatusCode on method execution requests.

If the execution of an OPC UA method fails in the OPC UA server, the method shall return a "Bad" StatusCode according [IEC 62541] Part 4 chapter "5.11 Method Service Set", chapter "5.3 Service results" and chapter "7.34 StatusCode", respectively.
Example: If a OPC UA method is not supported by the OPC UA server, the SFC can be informed e.g. by returning the StatusCode "Bad_NotSupported" or "Bad_NotImplemented".

2.6.2.6.2 Executable Attribute

[old ID] - In order to prevent an OPC UA client from calling OPC UA methods which are not supported by the OPC UA server and not defined in [IEC 62541], the OPC UA server shall set the method to not executable.

The executable attribute of an OPC UA method indicates if the method is currently executable. If it is set to "False", the method is not executable. This should only be used for methods not defined in [IEC 62541]. According to [IEC 62541] part 5 chapter "5.5 Methods" all methods defined in OPC UA standard should be executable (executable attribute set to “True”), unless it is defined differently in the method definition.

2.6.2.7 Manufacturer Specific Extensions

? MOVE TO User-Modelling-Guidline

[old ID] - The namespace name of the OPC UA model parts, which provided as manufacturer / project specific extension (not provided by CONEMP), shall be defined using the following pattern:

"http://[top level domain]/[origin]/[name]/[version]/".

[top level domain] shall be replaced by the top level domain of the organization, which has provided the model.
[origin] shall be replaced by the origin of the data point. Each product or project specific data point instance or type shall be in a namespace with [origin] replaced by a corresponding product or project identifier.
[name] shall be replaced by the name, for example of the interface.
[version] shall be replaced with the version number of the OPC UA information model for the given namespace. Each time the model changes, the version number shall be increased.

Examples:
"http://company.com/PRODUCT550/SDI-GEN/1.0/"
"http://company.com/SBB/SDI-TDS/1.1/"

This definition ensures an effective aggregation of multiple product groups, a commonly used pattern and conflicting namespaces.

3 Non-functional characteristics

ToDo: Add content e.g. security, PRAMS. etc.

4 Appendix

ToDo

Topics to be considered:

- ~~Permissions (Generic, Specific at SMI / SDI)~~
- ~~Security => Part of the interface specific documents~~
- ~~Direct Connect, ReverseConnect (Generic, Specific at SMI / SDI)~~
- ~~OPC UA Connection Handling~~
- ~~Redundancy~~
- ? OPC UA version
- ? Modelling requirements shall be part of modelling guideline (iuser, product), SDI-GEN