OPC UA over TSN

Technology description

Version: **1.00 (May 2021)**
Order no.:

Translation of the original documentation

Publishing information

B&R Industrial Automation GmbH

B&R Strasse 1

5142 Eggelsberg

Austria

Telephone: +43 7748 6586-0

Fax: +43 7748 6586-26

office@br-automation.com

Disclaimer

1610288938981-1.00

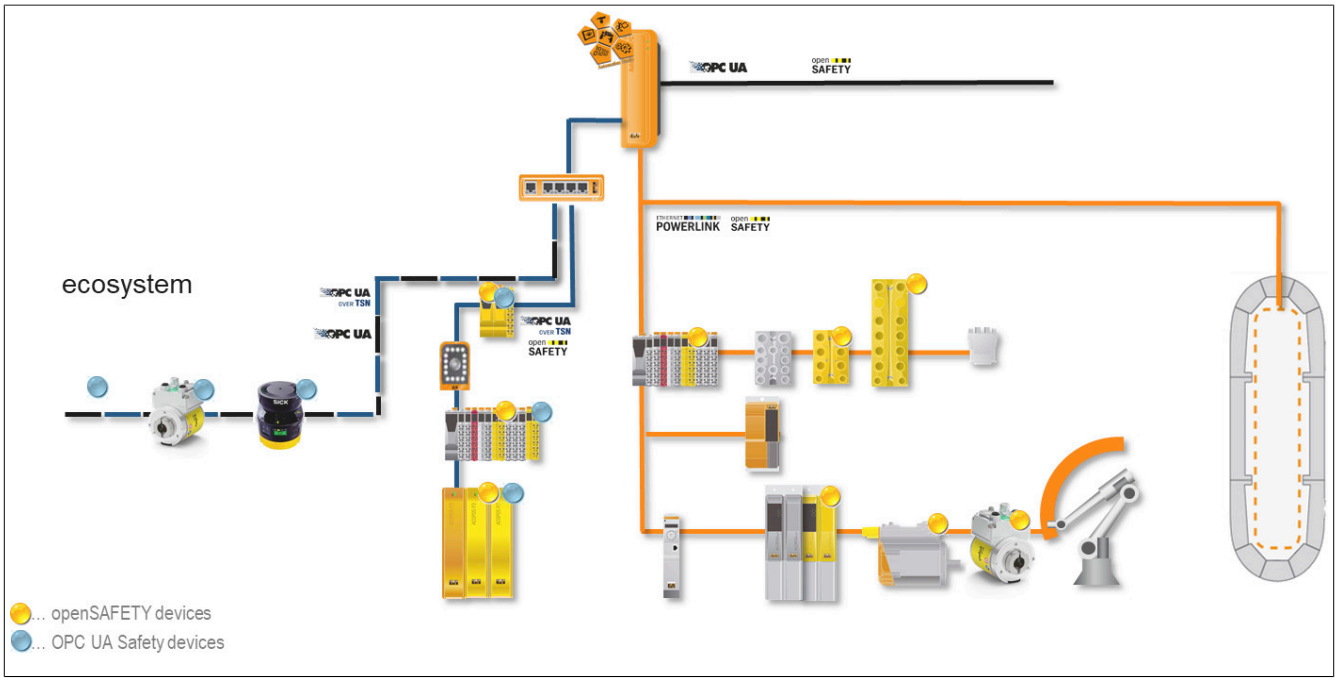
1 Introduction

OPC UA over TSN constitutes B&R's heartbeat technology for industrial (real-time) communication with PLC to PLC communication since 2021. It enables IT/OT convergence, while maintaining and extending the OT-relevant features of communication. Those are

- Superior deterministic real-time performance
- Maximum useable non-real-time bandwidth
- Integrated Security
- Integrated Safety
- Self-describing device information
- 100% Standard OPC UA
- 100% Standard Ethernet with TSN
- Application profiles and companion specifications
- Vast ecosystem

Typical OPC UA over TSN devices provide an OPC UA Server that hosts an information model exposing device information, e.g. identification properties, asset model, configuration parameters and available channels, diagnostic information, and device-type specific functionality. Additionally, devices provide one or more Ethernet network interfaces with TSN support. Devices with more than one interface may provide an internal TSN switch, whose features and configuration parameters are exposed via standardized YANG models and configured employing the NETCONF protocol. In short, Ethernet TSN constitutes the network infrastructure to connect OPC UA devices. The Field Level Communications initiative in the OPC Foundation (among many others) defines a way to utilize and configure various TSN mechanisms in a vendor-interoperable and protocol-interoperable way (to other IEC/IEEE 60802-compliant networks).

Possible topology of a B&R system



2 Scenarios

The scenarios described in this chapter show the areas of application in which OPC UA over TSN or parts of it can be used.

2.1 PubSub extension

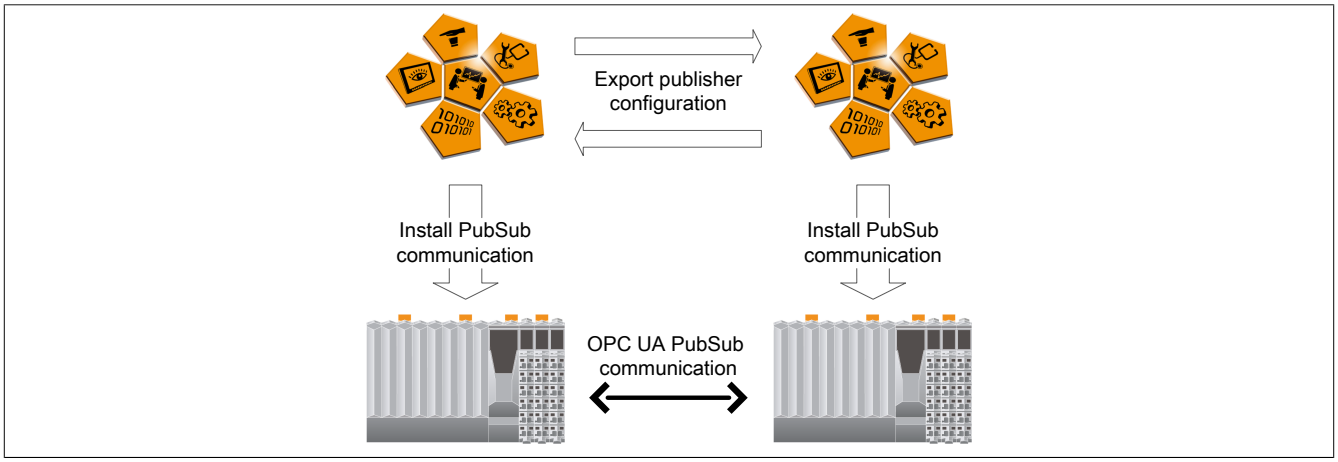
2.1.1 PubSub communication between B&R controllers

OPC UA PubSub can be used to exchange non-real-time-critical data in an existing Ethernet network between different machines using B&R controllers. PubSub can be used in all areas of application that were previously covered by Modbus TCP, PROFINET or EtherNet/IP, for example. In contrast to OPC UA client/server connections, CPU utilization is significantly reduced, especially with multiple communication partners. In addition, the real-time capability is improved.

It is also possible to achieve real-time communication only by using OPC UA PubSub without TSN mechanisms on an underutilized network and on underutilized controllers. If it is sufficient for the process being controlled by the controllers that communication takes several milliseconds and hard real time is not required, OPC UA PubSub without TSN can be used for such applications.

To implement this scenario, both B&R controllers are configured with Automation Studio; the publisher configurations are exchanged between the Automation Studio configurations of the two controllers using an export/import mechanism.

For detailed information about the necessary configuration steps, see section "PubSub configuration" in Automation Help.

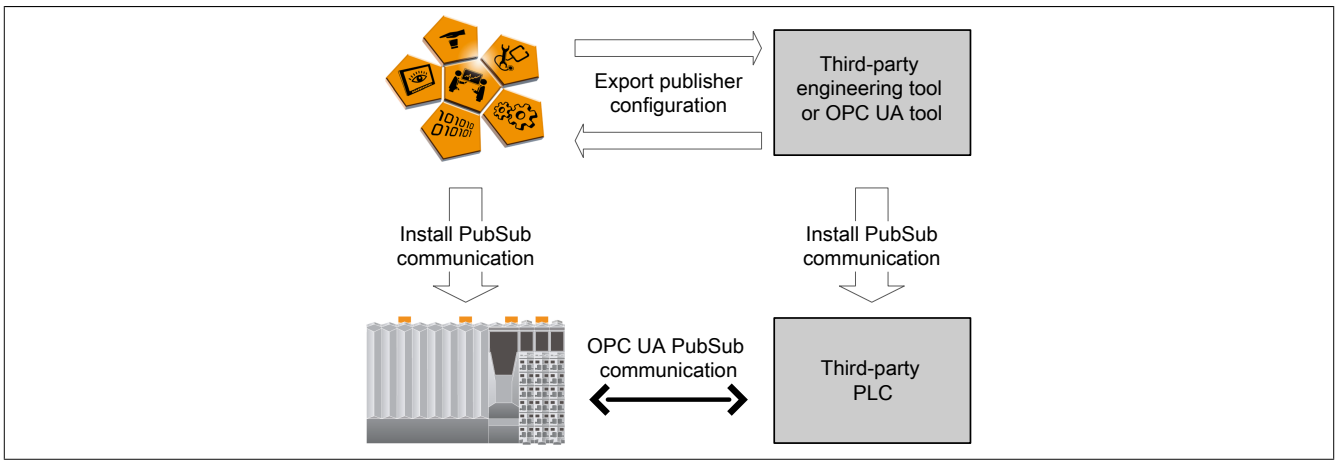


2.1.2 PubSub communication with third-party manufacturers

OPC UA PubSub can also be used for communication with third-party devices, as in the scenario for OPC UA PubSub communication between B&R controllers.

The 3rd-party device is configured with the engineering tool of the 3rd-party device manufacturer or with other OPC UA tools. The PubSub configuration is exchanged with Automation Studio using a standardized file format so that the data offered by the third-party device can be used for a subscriber configuration in Automation Studio and the third-party device can interpret the publisher configuration created by Automation Studio.

For detailed information about the required configuration steps for the Automation Studio part, see section "PubSub configuration" in Automation Help.

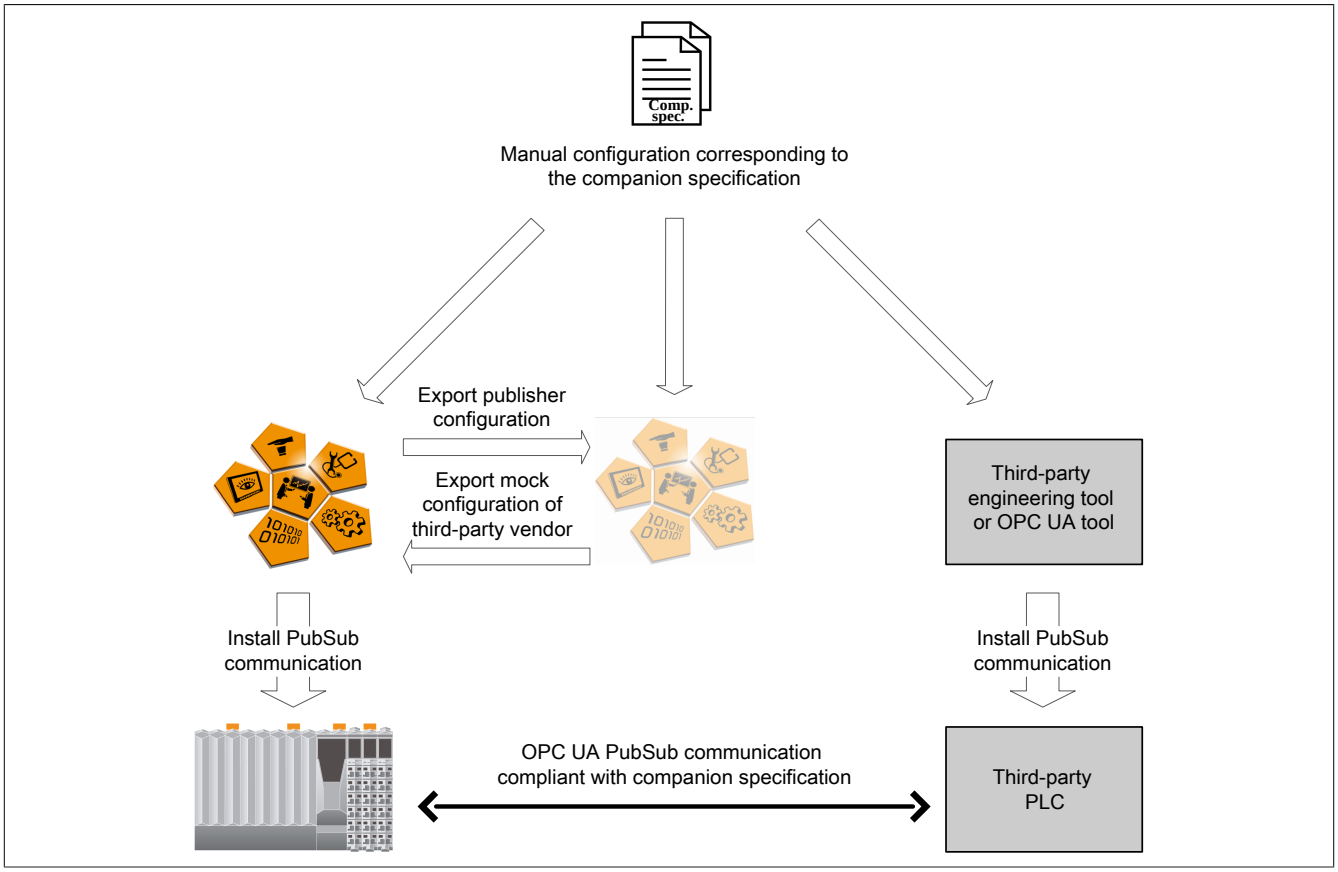


2.1.3 PubSub and companion specifications

Like OPC UA client/server, there are also companion specifications for OPC UA PubSub that define which data should be communicated in which form for certain machine types. Adhering to such companion specifications makes it possible for machines from different manufacturers to work together easily.

The companion specification defines how the data to be published and subscribed should look. To be in compliance with a certain companion specification, the PubSub editor in Automation Studio must be used (see section "PubSub configuration" in Automation Help) to manually create PubSub configurations (PubSub WriterGroups, PubSub DataSets, etc.) according to the companion specification.

It is also possible to create the configuration of the expected remote station according to the companion specification. This makes it possible to completely configure a customized controller without knowing the specific machine of the third-party manufacturer.

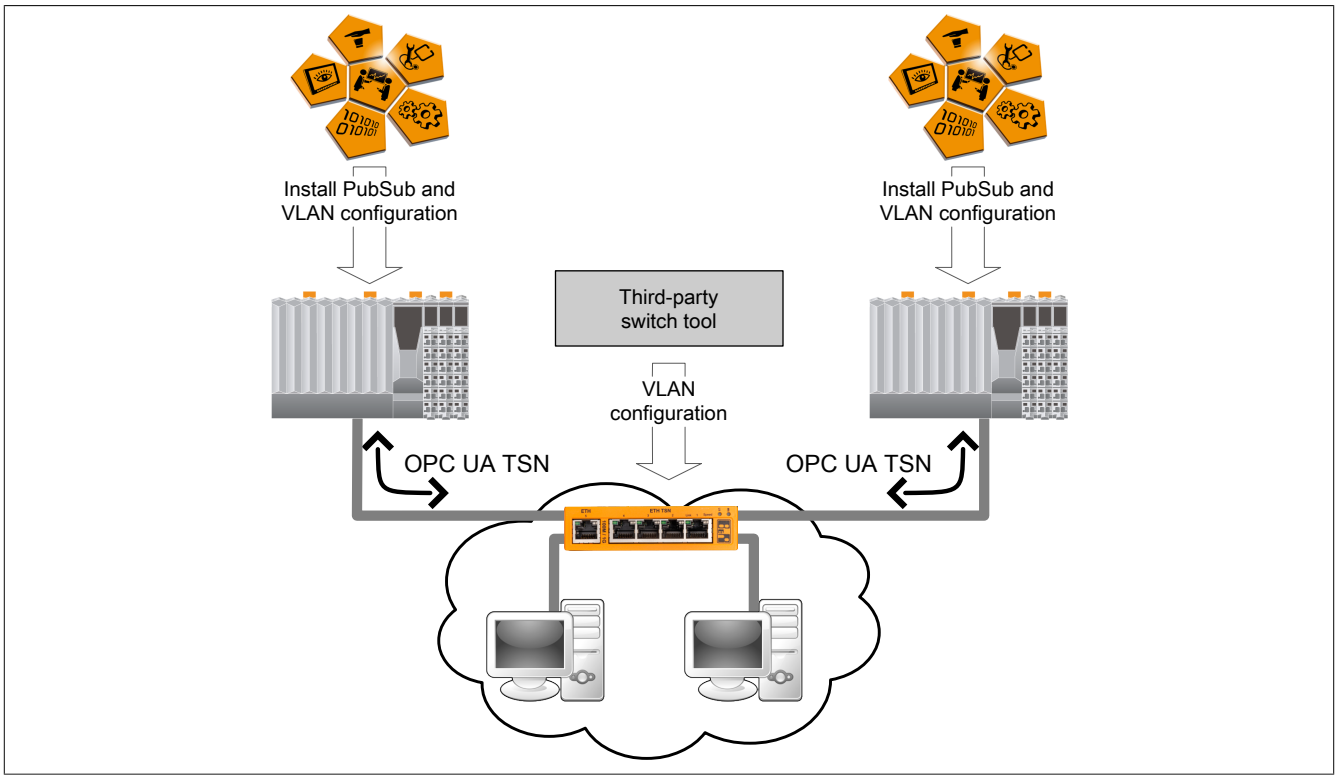


2.1.4 Transfer guarantee on the TSN network

With OPC UA over TSN, B&R controllers offer a communication option with transfer guarantee that can be used over converged TSN network infrastructures. This allows machines connected to a plant's existing IT network to communicate without this communication being endangered by best-effort traffic on the network. For this purpose, PubSub messages are marked with VLAN tags so that a TSN-capable switch can handle them with higher priority; this traffic is not disturbed by the remaining best-effort traffic.

The B&R controllers are configured with regard to PubSub as described in section "PubSub configuration" in Automation Help and VLAN tags are also set for PubSub messages. This VLAN information must then also be configured on the TSN switch so that it prioritizes PubSub messages higher so that other network traffic does not interfere with these messages.

In order to execute simultaneous events on the controllers, synchronization of task class 1 by PTP can also be configured in this case as described in the following scenario (section "PTP configuration" in Automation Help).



2.2 PTP extension

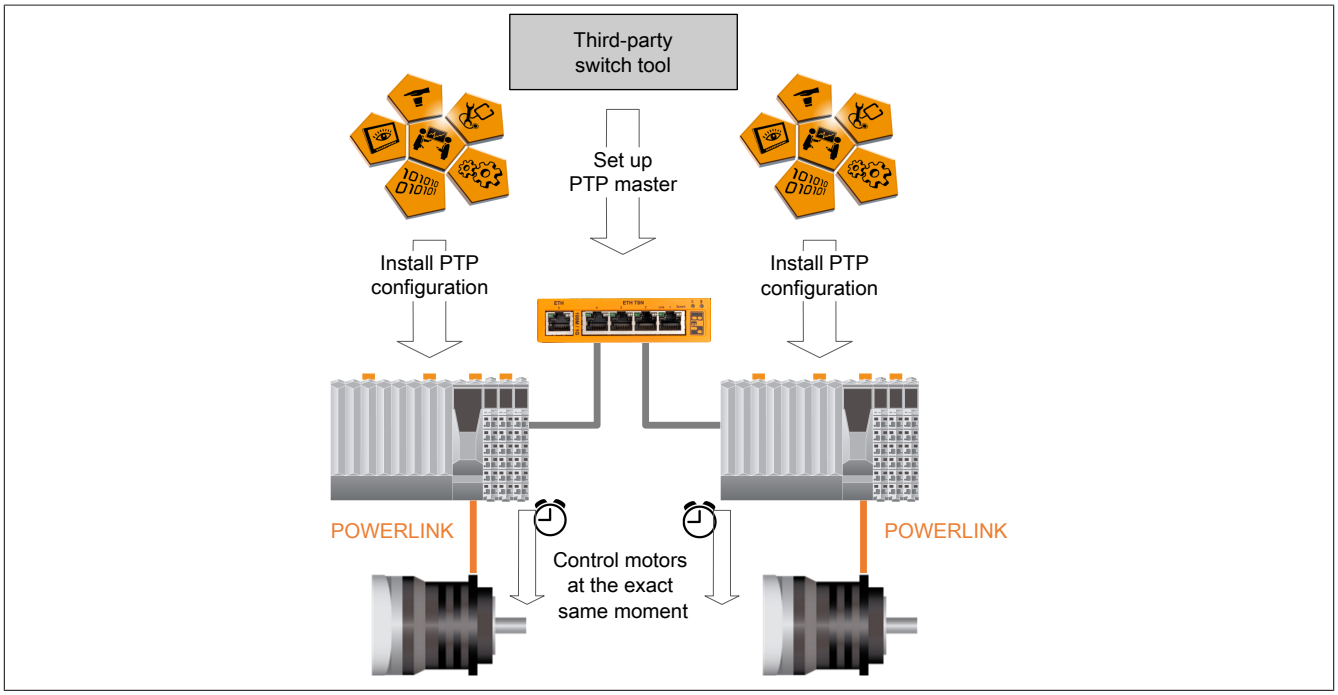
2.2.1 Simultaneous control

IEEE 802.1AS (gPTP) allows very accurate time synchronization. B&R controllers make it possible to synchronize the system tick to PTP within the range of a few microseconds. If the same cycle time for task class 1 is selected on the B&R controllers and this cycle time is equal to the system tick, the cycles of task class 1 are also synchronized within a few microseconds.

With the synchronized task class, it is possible to trigger an event (e.g. controlling a motor) on different B&R controllers simultaneously. The controllers can exchange information, e.g. via PubSub, about which events should take place and when they should take place. The current PTP time can be read out on the controllers while task class 1 is processed so that both controllers can execute the planned event at exactly the same time.

This allows events to be executed simultaneously on different controllers, but not yet in real time. This would additionally require deterministic communication between the controllers, which is not supported by the current implementation at the moment. Like Modbus TCP, PROFINET or EtherNet/IP, PubSub can currently be used to achieve fast and reliable communication (with high probability on not-fully-loaded controllers), but hard real time cannot be guaranteed (see "PubSub for real-time communication" on page 19).

PTP must be enabled on the controllers and configured as a timer for the system tick (section "PTP configuration" in Automation Help). In addition, a PTP master (e.g. TSN switch) must be available in the network.



2.3 Real-time PubSub extension

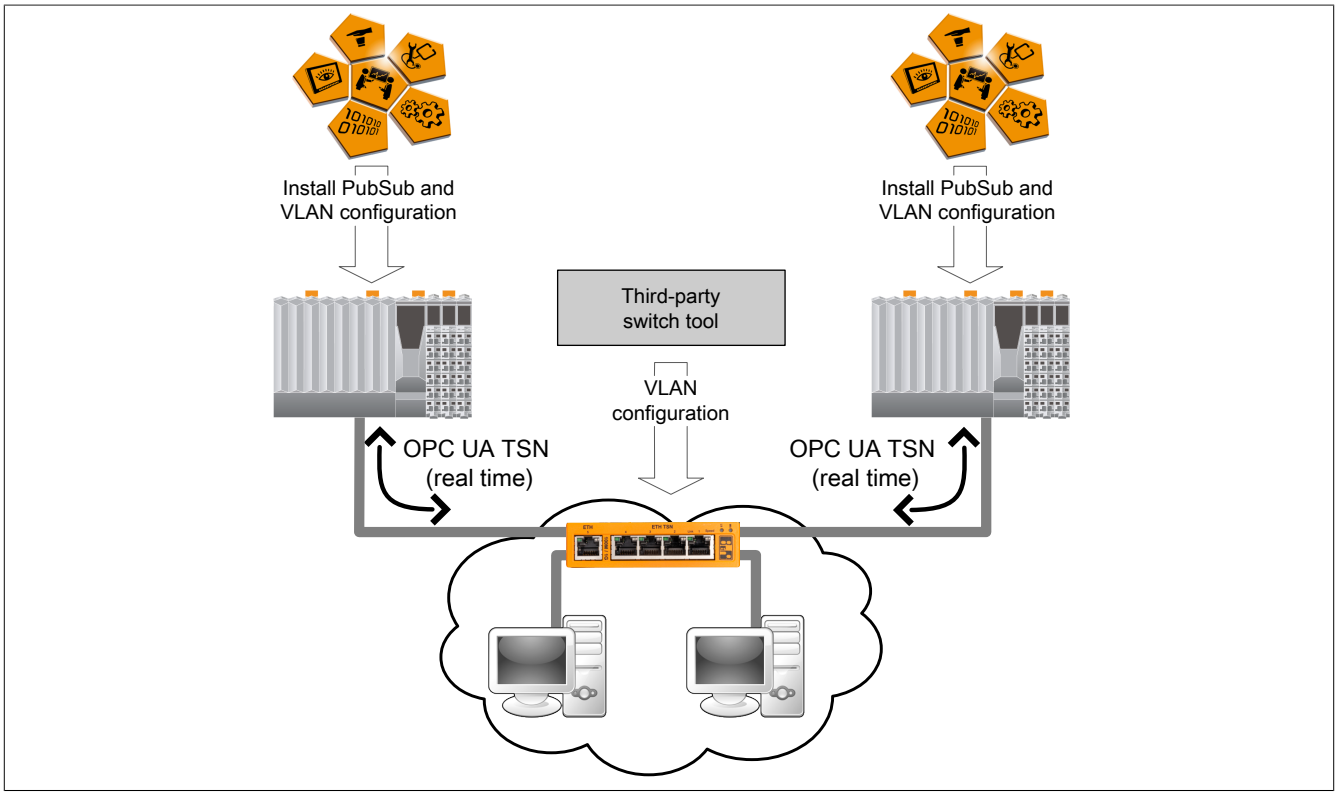
2.3.1 Real-time communication between B&R controllers

The defined structure of the PubSub frames[1)] and the use of standardized TSN mechanisms enable real-time capable and high-performance communication via standard Ethernet hardware. This means that use cases that were previously only possible with special hardware support, for example POWERLINK, can now be implemented without such hardware support.

PubSub can therefore be used to implement real-time communication between controllers ("controller-to-controller communication"). Compared to POWERLINK iCN, PubSub has the advantage that communication can take place over a network on which additional Ethernet traffic is possible. Another advantage is that several third-party manufacturers support PubSub technology.

In order to be able to distinguish high-priority PubSub data from the remaining data on the network, PubSub data is provided with a VLAN tag. A TSN-capable switch therefore treats it with higher priority and this traffic is not disturbed by the remaining traffic.

The PubSub configurations are exchanged (section "PubSub configuration" in Automation Help) via export and import of the configurations between the communication partners involved. The TSN switch is configured using any third-party tool for switch configuration.



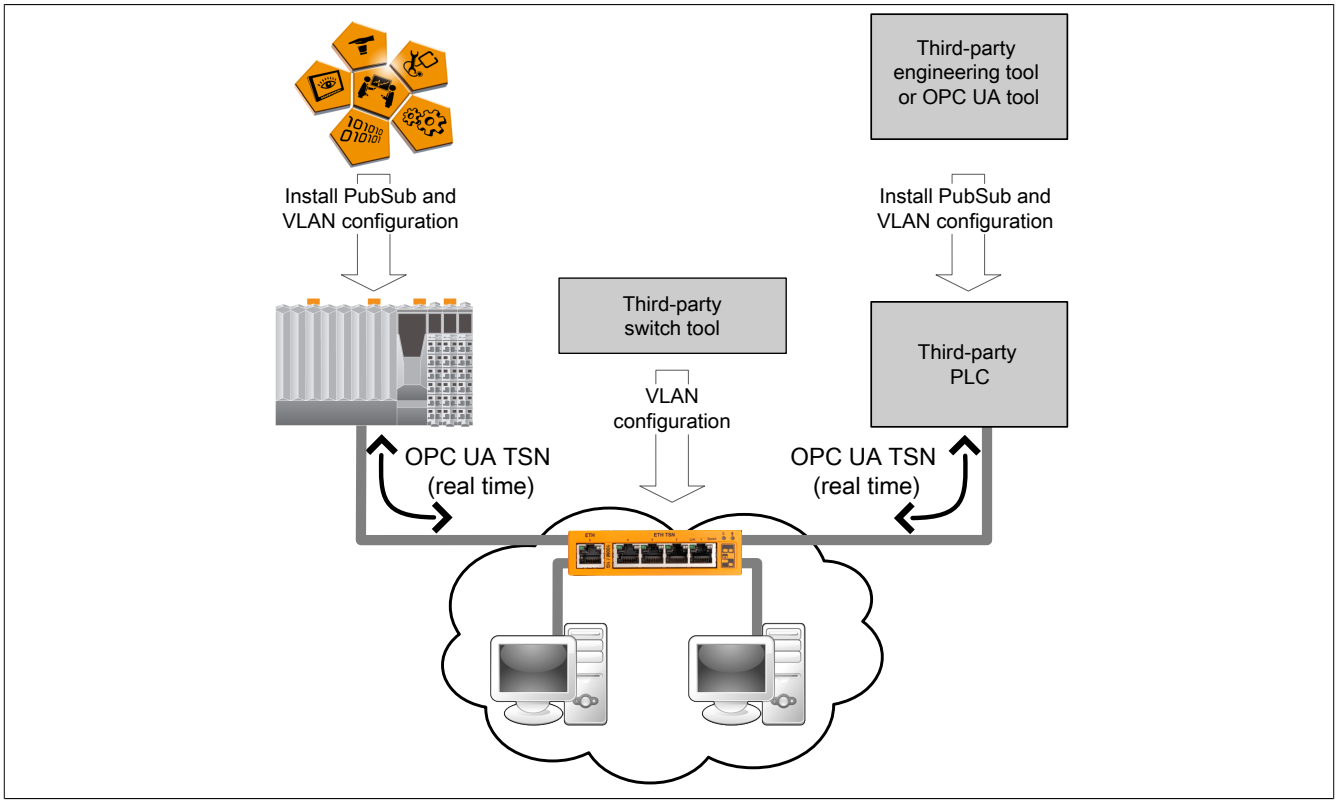
[1)] In "periodic fixed" frame format

2.3.2 Real-time communication with third-party devices

Standardized PubSub technology makes it possible to communicate with third-party controllers. As long as these manufacturers also provide real-time capable communication, it is possible to communicate with their devices in real time using standardized technology and a standard Ethernet network.

Especially in combination with OPC UA PubSub companion specifications, this opens up new possibilities. Machines that are part of a real-time-critical process can be combined and replaced easily, regardless of the manufacturer.

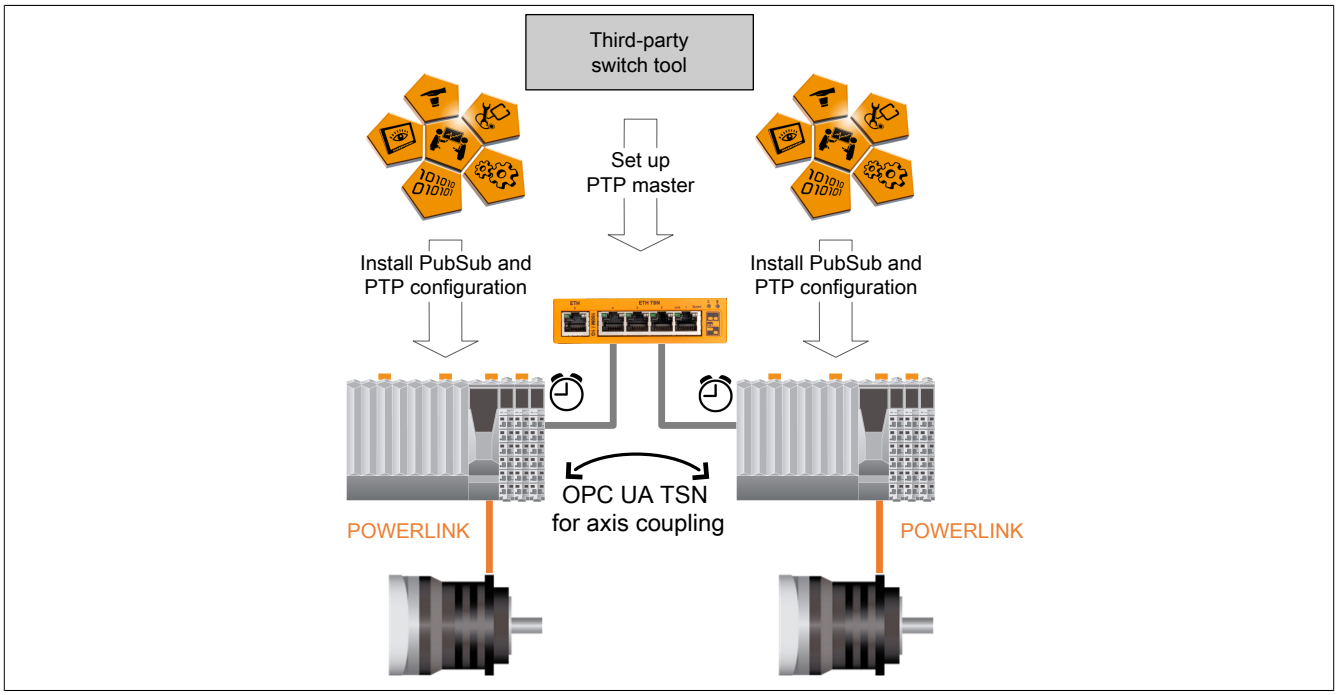
PubSub is configured as described in the previous chapter on real-time communication between B&R controllers. This also applies to the configuration of the TSN switch; configuration can also take place via the third-party controller.



2.3.3 Axis coupling

Due to the high-performance PubSub real-time communication and the possibility of synchronizing the controllers via IEEE 802.1AS (gPTP), POWERLINK axes can be coupled to different controllers.

PubSub communication allows cycle times from 400 µs with a latency of only 4 cycles. PTP can be used to synchronize the communicating controllers within a few microseconds. The POWERLINK devices connected to the controllers are also precisely synchronized this way. This makes it possible to synchronize axes with the same cycle time between the controllers.

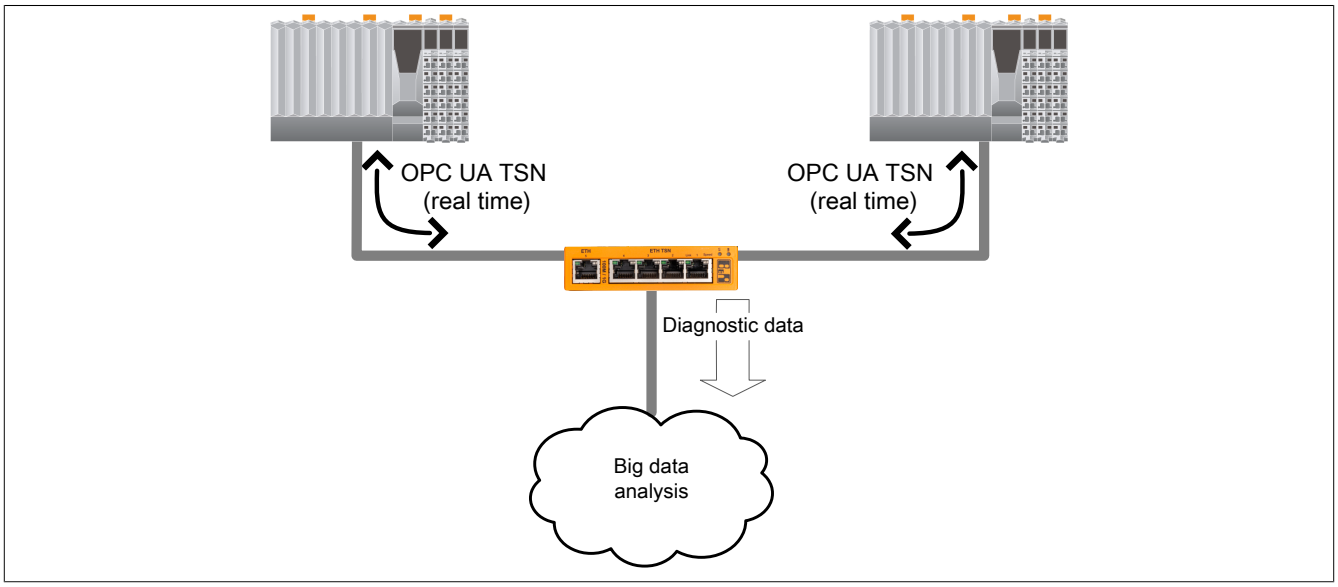


2.3.4 Big data

TSN mechanisms can be used to prioritize the communicated data. It is possible to distinguish between important and less important data both on the network and on the controller. The network can therefore be fully utilized by best-effort traffic, still guaranteeing correct real-time traffic.

This is a big difference in comparison to existing communication technologies that are offered via standard Ethernet. Technologies such as PROFINET or Modbus TCP make it possible to operate in "soft real time" under certain conditions such as very low network load. With OPC UA TSN, however, it is possible to operate in "hard real time", even if the network is heavily loaded.

This makes use cases possible in which a large amount of data is accessed from controllers that simultaneously handle real-time-critical communication.

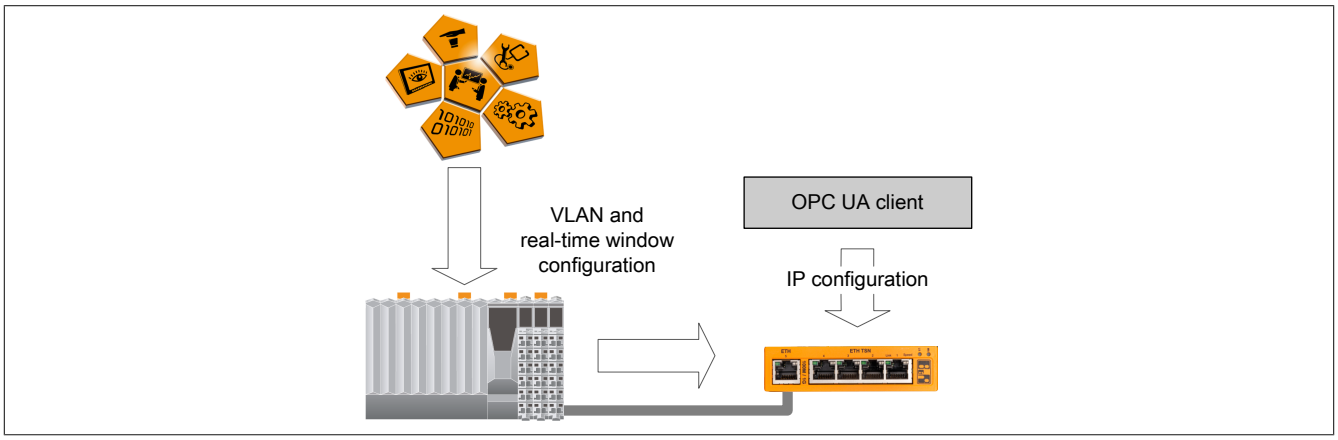


2.4 Expansion stage switch configuration

In order to simplify the configuration of the TSN switch timing parameters for the user, it should be created using Automation Studio. A configuration created in Automation Studio is then transferred to Automation Runtime, and Automation Runtime configures the switch. This way of creating a configuration even makes it possible to replace defective switch hardware and start up the new hardware without using Automation Studio.

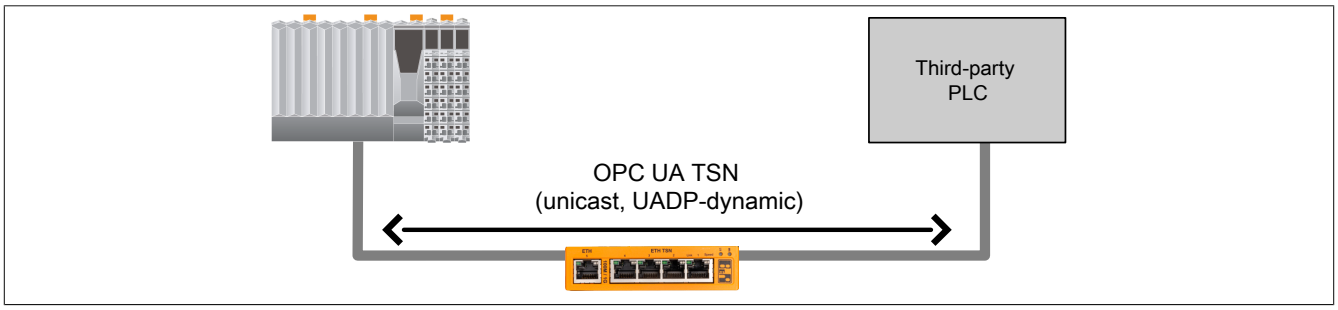
It is possible to configure time windows for the TSN switch, in which only high-priority PubSub traffic marked with VLANs is forwarded. This way it can be ensured that this traffic is not disturbed. In order for the TSN switch to be identified, it must have a unique IP address. This IP address must be configured in advance on the switch using an OPC UA client.

With the additional feature of configuring time windows for the TSN switch, the use cases listed in the previous chapters concerning networks that also include best-effort stations are now easier to implement. It is no longer necessary to use a third-party tool for configuring the time windows.



2.5 PubSub features extension

To support PubSub communication with as many third-party manufacturers as possible, additional PubSub features such as communication via unicast addresses or communication via profile UADP Dynamic Fixed are supported. The support of the features standardized by PubSub makes it possible to communicate with various third-party manufacturers, which only support a small part of the PubSub features.



3 Supported products

OPC UA is supported in SG4 Automation Runtime version 4.0 and later.

OPC UA PubSub is supported in SG4 Automation Runtime version 4.90 and later[2)].

IEEE 802.1AS (gPTP) will be tentatively supported in SG4 Automation Runtime version 4.92 and later.

High precision time synchronization (<5 µs) will be supported on the following target families: X20CPx86x, xPC2200, xPC3100, PPC900, C80, and products: 0ACST052.1.

OPC UA PubSub TSN will be tentatively supported in SG4 Automation Runtime version 4.92 and later.

High precision time-triggered sending and receive window control will be supported on the following target families: X20CPx86x, xPC2200, xPC3100, PPC900, C80.

[2)] The following targets/features are not supported: Controller redundancy, CiR, Hypervisor and Application Modules

4 OPC UA

Sections 4.1 to 4.3.1 have been taken from the website of the OPC Foundation[3].

4.1 What is OPC?

OPC is the interoperability standard for the secure and reliable exchange of data in the industrial automation space and in other industries. It is platform independent and ensures the seamless flow of information among devices from multiple vendors. The OPC Foundation is responsible for the development and maintenance of this standard.

The OPC standard is a series of specifications developed by industry vendors, end-users and software developers. These specifications define the interface between Clients and Servers, as well as Servers and Servers, including access to real-time data, monitoring of alarms and events, access to historical data and other applications.

When the standard was first released in 1996, its purpose was to abstract PLC specific protocols (such as Modbus, Profibus, etc.) into a standardized interface allowing HMI/SCADA systems to interface with a "middle-man" who would convert generic-OPC read/write requests into device-specific requests and vice-versa. As a result, an entire cottage industry of products emerged allowing end-users to implement systems using best-of-breed products all seamlessly interacting via OPC.

Initially, the OPC standard was restricted to the Windows operating system. As such, the acronym OPC was borne from OLE (object linking and embedding) for Process Control. These specifications, which are now known as OPC Classic[4], have enjoyed widespread adoption across multiple industries, including manufacturing, building automation, oil and gas, renewable energy and utilities, among others.

With the introduction of service-oriented architectures in manufacturing systems came new challenges in security and data modeling. The OPC Foundation developed the OPC UA[5] specifications to address these needs and at the same time provided a feature-rich technology open-platform architecture that was future-proof, scalable and extensible.

Today the acronym OPC stands for Open Platform Communications.

4.2 OPC UA Client Server

The OPC Unified Architecture (UA), released in 2008, is a platform independent service-oriented architecture that integrates all the functionality of the individual OPC Classic specifications into one extensible framework.

This multi-layered approach accomplishes the original design specification goals of:

- Functional Equivalence: all COM OPC Classic specifications are mapped to UA
- Platform independence: from an embedded micro-controller to cloud-based infrastructure
- Security: encryption, authentication, and auditing
- Extensible: ability to add new features without affecting existing applications
- Information Modeling and Access: for defining complex information

[3] https://opcfoundation.org/about/what-is-opc
[4] https://opcfoundation.org/about/opc-technologies/opc-classic/
[5] https://opcfoundation.org/about/opc-technologies/opc-ua/

4.2.1 Functional Equivalence

Building on the success of OPC Classic, OPC UA was designed to enhance and surpass the capabilities of the OPC Classic specifications. OPC UA is functionally equivalent to OPC Classic, yet capable of much more:

- **Discovery:** find the availability of OPC Servers on local PCs and/or networks
- **Address space:** all data is represented hierarchically (e.g. files and folders) allowing for simple and complex structures to be discovered and utilized by OPC Clients
- **On-demand:** read and write data/information based on access-permissions
- **Subscriptions:** monitor data/information and report-by-exception when values change based on a client's criteria
- **Events:** notify important information based on client's criteria
- **Methods:** clients can execute programs, etc. based on methods defined on the server

Integration between OPC UA products and OPC Classic products is easily accomplished with COM/Proxy wrappers that are available in the download section.

4.2.2 Platform independence

Given the wide array of available hardware platforms and operating systems, platform independence is essential. OPC UA functions on any of the following and more:

- **Hardware platforms:** traditional PC hardware, cloud-based servers, PLCs, micro-controllers (ARM etc.)
- **Operating Systems:** Microsoft Windows, Apple OSX, Android, or any distribution of Linux, as well as PLC runtime environments like Automation Runtime, CoDeSys, etc.

OPC UA provides the necessary infrastructure for interoperability across the enterprise, from machine-to-machine, machine-to-enterprise and everything in-between.

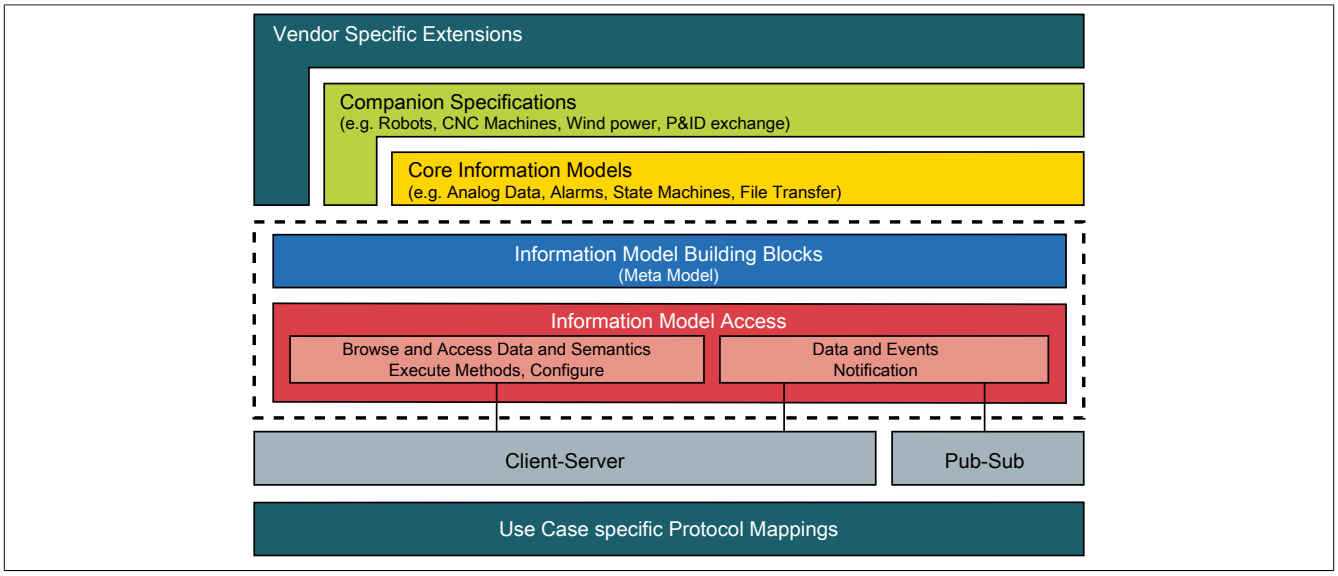
4.2.3 Security

One of the most important considerations in choosing a technology is security. OPC UA is firewall-friendly while addressing security concerns by providing a suite of controls:

- **Transport:** numerous protocols are defined providing options such as the ultra-fast OPC-binary transport or the more universally compatible JSON over Websockets, for example
- **Session Encryption:** messages are transmitted securely at various encryption levels
- **Message Signing:** with message signing the recipient can verify the origin and integrity of received messages
- **Sequenced Packets:** exposure to message replay attacks is eliminated with sequencing
- **Authentication:** each UA client and server is identified through X509 certificates providing control over which applications and systems are permitted to connect with each other
- **User Control:** applications can require users to authenticate (login credentials, certificate, web token etc.) and can further restrict and enhance their capabilities with access rights and address-space "views"
- **Auditing**: activities by user and/or system are logged providing an access audit trail

4.2.4 Extensible

The multi-layered architecture of OPC UA provides a "future proof" framework. Innovative technologies and methodologies such as new transport protocols, security algorithms, encoding standards, or application-services can be incorporated into OPC UA while maintaining backwards compatibility for existing products. UA products built today will work with the products of tomorrow.



4.2.5 Information Modeling and Access

The OPC UA information modeling framework turns data into information. With complete object-oriented capabilities, even the most complex multi-level structures can be modeled and extended.

This framework is the fundamental element of OPC Unified Architecture. It defines the rules and base building blocks necessary to expose an information model with OPC UA. While OPC UA already defines several core models that can be applied in many industries, other organizations build their models upon them, exposing their more specific information with OPC UA.

OPC UA also defines the necessary access mechanisms to information models.

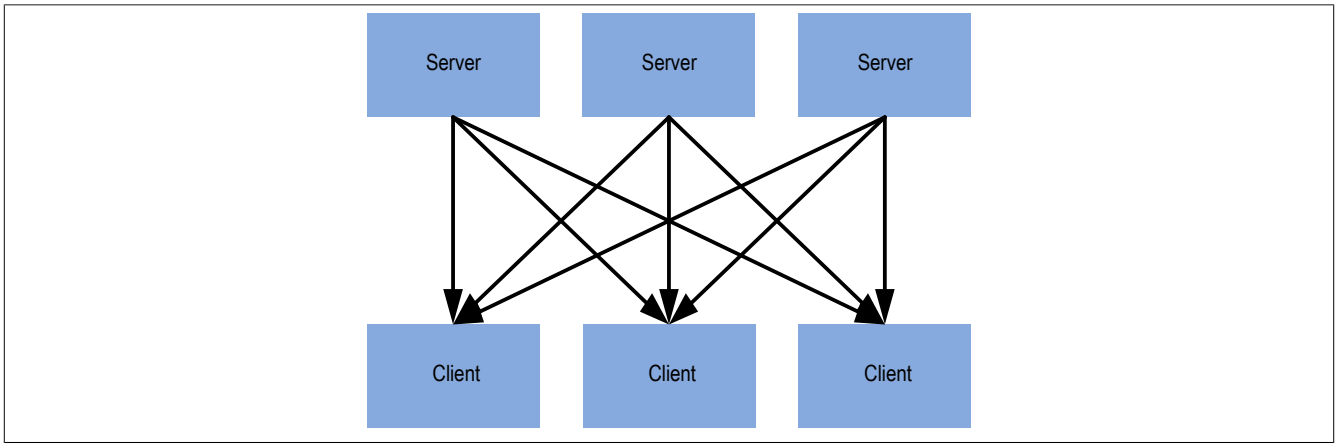
- Look-up mechanism (browsing) to locate instances and their semantic
- Read and write operations for current data and historical data
- Method execution
- Notification for data and events

For Client-Server communication the full range of information model access is available via services and in doing so follows the design paradigm of service-oriented architecture (SOA), with which a service provider receives requests, processes them and sends the results back with the response.

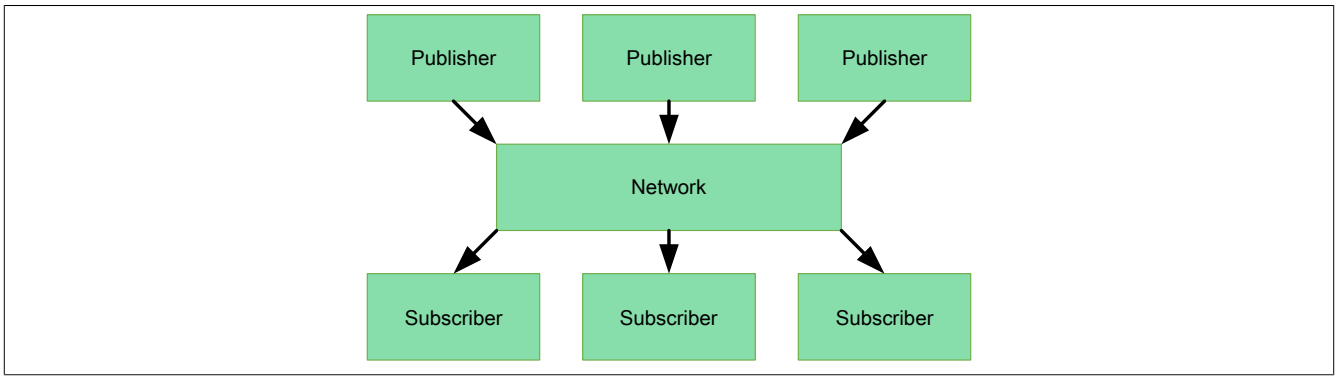
4.3 OPC UA PubSub

4.3.1 What is PubSub?

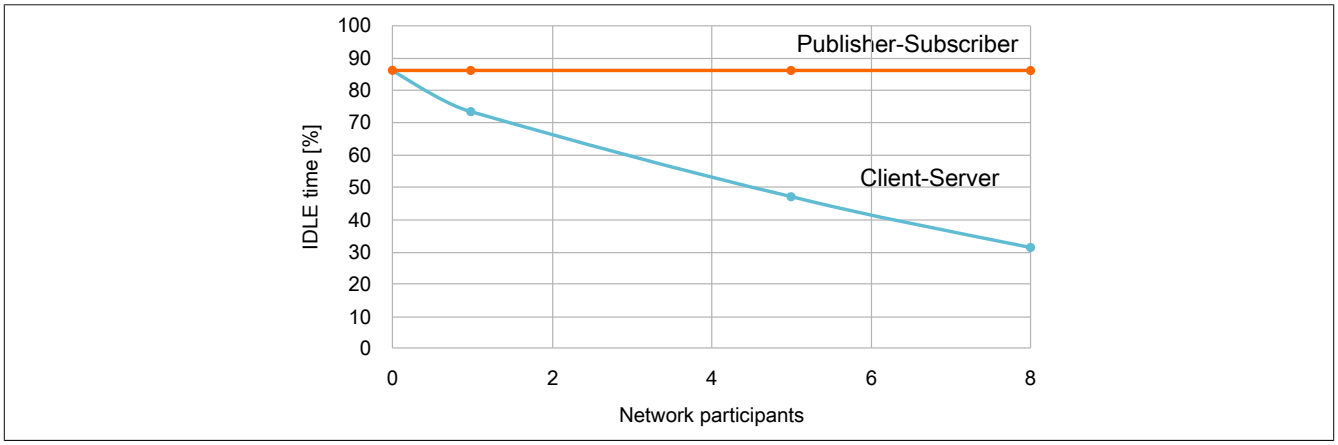
Publish-Subscribe (PubSub), provides an alternative mechanism for data and event notification. While in Client-Server communication each notification is for a single client with guaranteed delivery, PubSub has been optimized for many-to-many configurations.



With PubSub, OPC UA applications do not directly exchange requests and responses. Instead, Publishers send messages to a Message Oriented Middleware, without knowledge of what, if any, Subscribers there may be. Similarly, Subscribers express interest in specific types of data, and process messages that contain this data, without a need to know where it originated from.

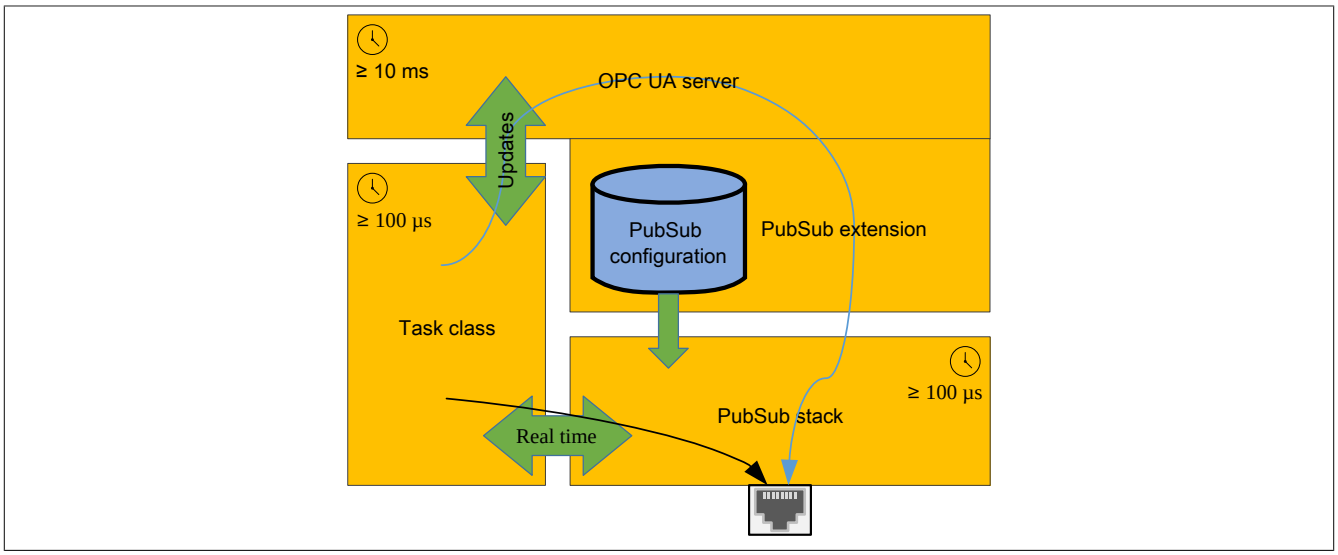


This way, a publisher that communicates with several subscriber requires much less computational resources than a server communicating with several clients.



4.3.2 PubSub for real-time communication

In addition to the support for many-to-many communication relationships, OPC UA PubSub also significantly improves real-time capable communication. The PubSub stack is executed as a separate process that can be connected to the OPC UA server as well as to specific "data providers". When connecting to the server, data updates depend on the speed of the server, which is executed in the B&R system during the idle time. The task class system, however, can be regarded as a separate data provider, where communication is achieved with the same real-time quality as with POWERLINK. The OPC UA server also receives process variable updates from the data provider so that relatively current values are also available in this way.



Each PubSub frame can be sent with its own cycle time and a time offset within the cycle. The network must provide appropriate mechanisms to also guarantee these hard real-time requirements on the receiver side of the network.

These mechanisms are described in the following chapter about TSN.

5 TSN

5.1 Introduction

TSN refers to real-time extensions of the IEEE Ethernet standard. The IEEE is a worldwide organization of mainly electrical engineering experts that hold scientific conferences, issue journal publications and drive standardization. The best-known standardization project is IEEE 802. This project deals with all the topics related to local area networks (LANs). IEEE 802.3 defines Ethernet (single ports) and IEEE 802.11 defines wireless LAN, while IEEE 802.1 deals with "LAN/MAN bridging and management" (also referred to as "switches").

Time Sensitive Networking (TSN) is a working group (WG) of IEEE 802.1 [6)].

The first real-time Ethernet extensions were developed for professional audio and video technology and released between 2009 to 2011. As the emerging opportunities in standardization became more apparent, working group Audio Video Bridging (AVB) was renamed to working group TSN in 2012. It had published approximately 10 standards and 4 AVB standards by the end of 2019. The standards and projects can be divided into 5 major areas:

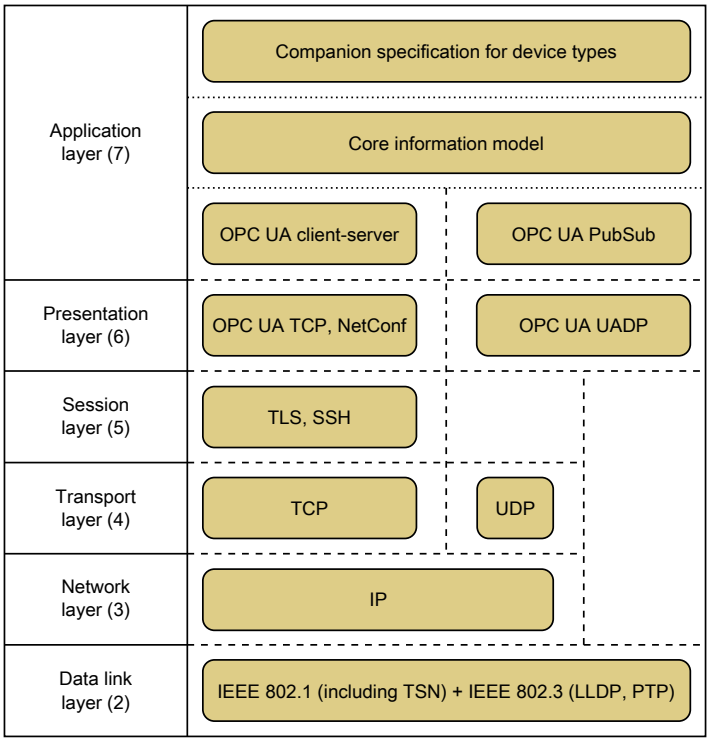
- Time synchronization
- Redundancy
- Short latency limited upwards
- Resource management
- TSN profiles

The use of TSN allows a degree of determinism for data transfer that was previously reserved for industrial Ethernet. In addition, the TSN extensions are specified in such a way that interoperability is possible in the same network. Hard real-time traffic, soft real-time traffic, reserved bandwidths for video streams, events and configuration as well as background traffic, etc. can be implemented simultaneously on the same cable with individual guarantees (see 5.4 "Traffic types"). In order to be able to plan a network configuration and to assign guarantees, the traffic in TSN networks is divided into stream and non-stream traffic. A stream is a directed flow of information from a "talker" to one or more "listeners" via a defined path on the network (switches). Such streams are usually requested with their desired real-time properties from the configuration instance of the network. It can permit the stream, configure the infrastructure accordingly and tell the talker and listener the communication parameters to use. Like Ethernet with QoS, non-stream traffic can be divided into classes. The classes can then receive a guarantee as a whole (e.g. reserved bandwidth for configuration data) or a relative priority to each other.

6) https://1.ieee802.org/tsn/

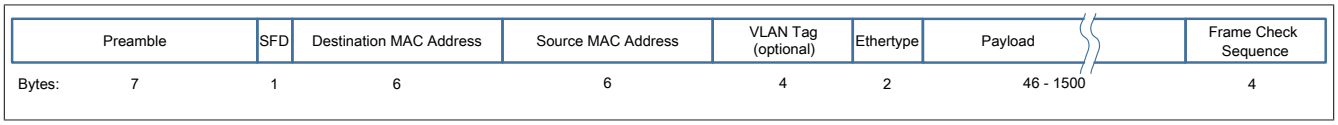
5.2 ISO/OSI model

In the ISO/OSI reference model for communication systems, OPC UA over TSN utilizes all available layers.



Data link - Layer 2

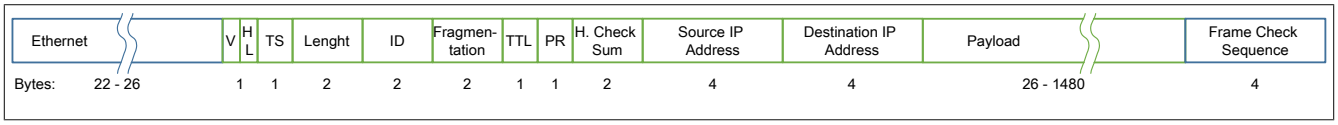
Ethernet (with TSN) forms layer 2. Ethernet frames (often just called frames) are sent on this layer and identified by an Ethernet header with an optional VLAN tag at the beginning of the frame[7)].



The VLAN tag is mandatory for (TSN) streams. The combination of destination MAC[8)] and VLAN ID (part of the VLAN tag) is used and interpreted as the stream ID. EtherType is a 16-bit value that specifies which protocol is used on the next layer (e.g. IP). The next layer in the ISO/OSI model is always encapsulated in the payload of the layer below it (see the following image). For PubSub frames, it is possible to embed a message (network message) directly in a frame and mark it with its own EtherType. This avoids IP and UDP header overhead if they are not needed.

Network - Layer 3

Internet Protocol (IP) is used on layer 3. IP packets (often just called packets), which are identified by an IP header at the beginning, are sent on this layer. The IP header contains the IP addresses of the transmitter and the receiver. The protocol field (PR) specifies which protocol is used on the next layer (e.g. UDP).



Such packets can be transmitted across routers that change the DMAC address of the Ethernet frame accordingly (next router on the path or receiver).

Transport - Layer 4

The two transport protocols UDP and TCP are used on layer 4. UDP is a connectionless protocol that sends datagrams, which have a UDP header (8 bytes) at the beginning that contains the source and destination port of the connection. UDP does not provide any information about whether datagrams have been received by the receiver. This must therefore be checked on higher levels if necessary. TCP is a connection-oriented protocol that sends segments, which have a TCP header (20 bytes) at the beginning that contains the source and destination ports

7) ETH and IP header fields that are not required/written are only mentioned in their abbreviations
8) The TSN streams use multicast DMACs per convention to make them more distinguishable from physically available addresses (even if the stream is addressed to only one listener and thus corresponds to a unicast frame)

of the connection as well as information about the segment. TCP ensures (within defined limits) that a segment arrives at the receiver and that the segments can be reconstructed in the correct order. Due to this mechanism, TCP is not suitable for transferring real-time-critical data with short latencies, but is mainly used for asynchronous services with a focus on secure transfer. TCP is also used as the transport layer for most security protocols.

Session - Layer 5

Security protocols are used on layer 5. TLS is an important protocol for OPC UA over TSN. It establishes the connection and checks the identity of the communication partners using unique keys. Certificates or user/password information can be used for this. For the actual data transfer, the channel can then be signed and/or encrypted[9)].

Presentation - Layer 6

Layer 6 contains the encoding of the message contents, which can vary depending on the protocol used on the application layer (OPC UA, NETCONF, etc.).

Application - Layer 7

Layer 7 is where interaction with the application or application protocol takes place.

A message transmitted from one application to another contains information from all ISO/OSI layers at the same time. It is therefore simultaneously a frame, package and segment, for example, depending on which layer in the ISO/OSI model is currently being considered.

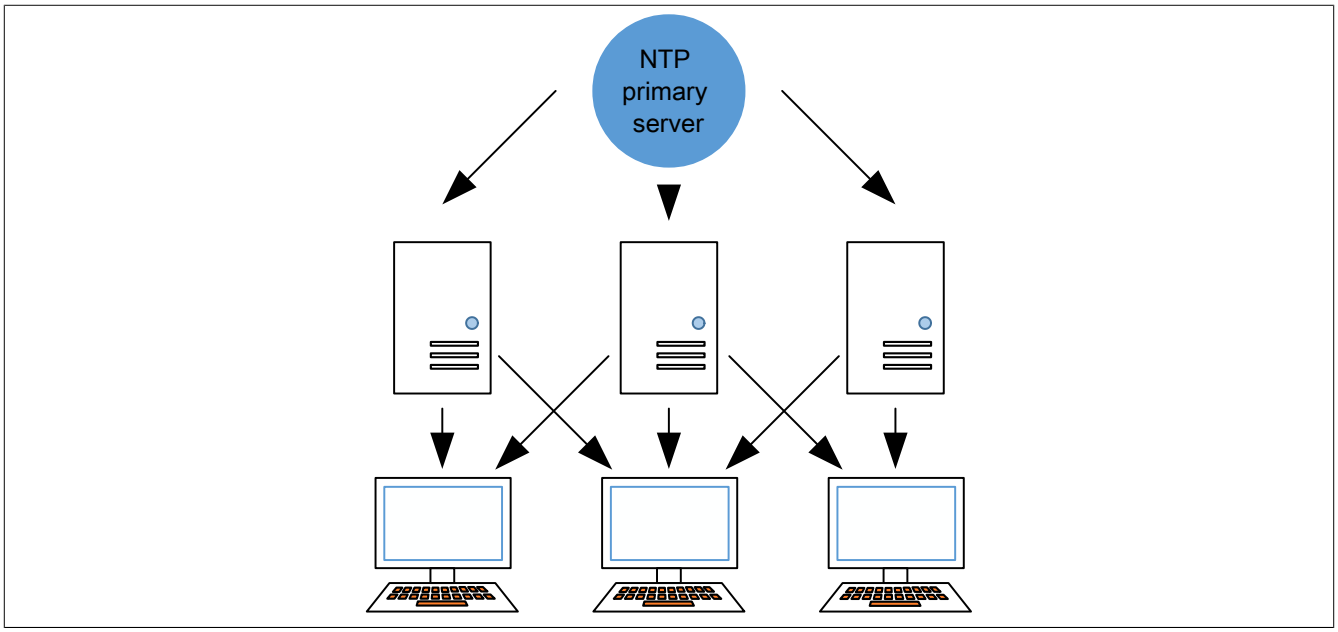
9) Via a session key that is renewed regularly

5.3 Time synchronization

Time synchronization forms the foundation of TSN. In order to function properly, many traffic shapers, which can be used to implement various guarantees on the network by shaping (regulating) traffic, need the same time basis on all infrastructure components that is as exact as possible. The time is used for transmitting and forwarding times as well as for calculating bandwidth and frame rates. For both calculations, a common understanding of the progression of time without knowledge about absolute values (synchronization) is sufficient. Time synchronization in Ethernet networks is a relatively old problem with just as many solutions based on different requirements. Most important available solutions:

- (S)NTP - (Simple) Network Time Protocol
- IEEE 1588 - Precision Time Protocol
- IEEE 802.1AS - Generalized Precision Time Protocol
- Industrial Ethernet

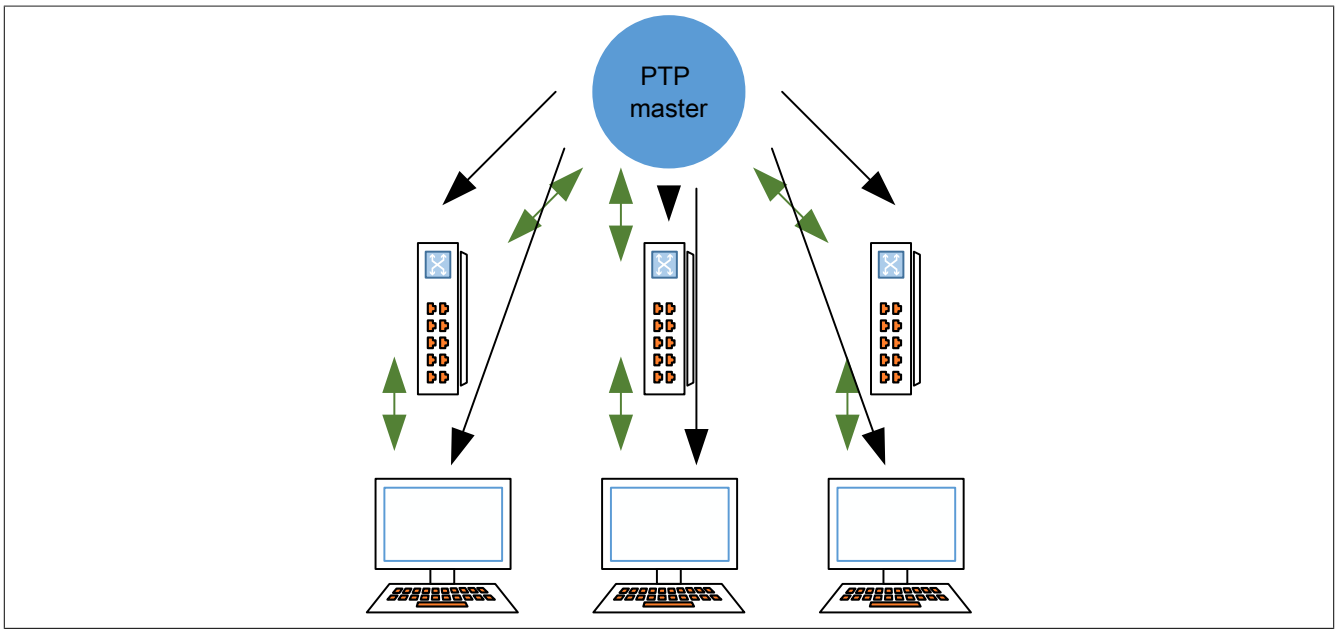
NTP (previously SNTP) is mainly used in applications in which the level of accuracy is sufficient in human terms (desktop PCs, mobile phones) and a global time base should be synchronized. A highly accurate master distributes the information about the current time to the end devices via various intermediate layers, searching for the fastest connection in each case. Due to its easy and efficient implementation, it is also widely used for industrial applications with time accuracy requirements ≥ 5-10 ms (e.g. process automation)[10].



NTP is not suitable for industrial real-time Ethernet with accuracy requirements in the microsecond and sub-microsecond range, however, so most technologies use their own mechanisms for time synchronization. POWERLINK, for example, uses the first frame in the cycle (SoC) that is sent by the master at highly precise, equidistant times to synchronize the stations. In other technologies (e.g. PROFINET, CIP Sync with EtherNet/IP), IEEE 1588 is used.

[10] The specified 5-10 ms refer to synchronization over the Internet. In local networks, values <500 µs can be reached.

IEEE 1588 was developed for measuring applications and is characterized by the fact that, in addition to the time distribution of the local master, the propagation delay of the messages is also accurately compensated[11]. This makes it possible to significantly increase accuracy compared to NTP. If the receive and transmission instants of the synchronization frames and frames for propagation delay measurement can be measured with hardware support, accuracies < 1 µs are possible. If measured only with software, accuracy is ~ 1 ms.



IEEE 1588 is a very comprehensive standard with many options and protocol mappings, which has resulted in the specification of various profiles (power, telecommunications, measurement, audio/video, etc.) that are not compatible with each other.

IEEE 802.1AS is also a profile of IEEE 1588 specific for AVB LANs with only a few configurable options to enable interoperability in multi-vendor TSN networks. In version 802.1AS-2020, multiple time domains (a working clock for time-controlled communication and motion applications[12] and a global time[13] for logging) and grand master redundancy are supported for the first time. In addition, 802.1AS has been optimized for large Ethernet networks. With IEEE 1588, however, control fading effects that influence synchronization accuracy can occur[14]. Because IEEE 1588 has been used in the field longer than 802.1AS, time gateway functions are offered on the market in order to be able to include networks with the two synchronization variants in a common time base.

[11] Marked in green in the image. Any time information from the master to a slave "collects" information about the propagation delay, allowing the slave to accurately back-calculate the time of the master.
[12] A strictly monotonically increasing counter that is not permitted to jump
[13] Corresponding to UTC time
[14] With IEEE 1588, the residence time of the frames is measured on the device with the already controlled software clock. With 802.1AS, this is measured using the free-running hardware clock. The absolute differences are absolutely negligible, but there is a mutual influence between propagation delay correction and the calculated time in IEEE 1588. This is why normal measurement inaccuracies always cause beat effects in larger networks.

5.4 Traffic types

Most Industrial Ethernet technologies support 2 types of data transfer:

- Real-time critical
- Non-real-time-critical

Real-time-critical data transfer can be divided into hard and soft real time. With hard real time, the times for transmitting, forwarding and receiving messages in the cycle are always the same. With soft real time, jitter occurs on the network here, but it is assumed that one frame can be sent and received per cycle.

With non-real-time-critical data transfer, there is a need to prioritize messages depending on the application. There are different implementations in industrial Ethernet for this. In 2017, the IIC TSN testbed examined the different traffic types established in industrial applications and their requirements for the first time and summarized them in a white paper[15]. The following traffic types were identified:

Types	Periodicity	Period	Synchronized with network	Data transfer guarantee	Tolerance to disturbances	Tolerance to data loss	Typical data size of the application	Criticality
Isochronous	Periodic	<2 ms	Yes	Deadline	0	None	Permanent: 30 to 100 bytes	High
Cyclic	Periodic	2 to 20 ms	No	Latency	≤ Latency	1 to 4 frames	Permanent: 50 to 1000 bytes	High
Events	Sporadic	N/A	No	Latency	N/A	Yes[1)]	Variable: 100 to 1500 bytes	High
Network monitoring	Periodic	50 ms to 1 s	No	Bandwidth	Yes	Yes[1)]	Variable: 50 to 500 bytes	High
Configuration and diagnostics	Sporadic	N/A	No	Bandwidth	N/A	Yes[1)]	Variable: 500 to 1500 bytes	Medium
Best-effort	Sporadic	N/A	No	None	N/A	Yes[1)]	Variable: 30 to 1500 bytes	Low
Video	Periodic	Frame rate	No	Latency	N/A	Yes[1)]	Variable: 1000 to 1500 bytes	Low
Audio/Voice	Periodic	Sampling time	No	Latency	N/A	Yes[1)]	Variable: 1000 to 1500 bytes	Low

1) Even with non-cyclic traffic types, there is often no room for packet loss from an application point of view (e.g. for events that are only transferred once). These losses are compensated by protocols on higher layers (with corresponding latencies), so no corrective measure for packet loss is required for layer 2.

It can be assumed that at least the following traffic types are used in industrial applications:

- Types for periodic process data exchange (isochronous or cyclic; isochronous in the B&R system)
- Network monitoring (messages required to maintain network function, e.g. time synchronization, LLDP)
- Configuration
- Best-effort[16)]

Depending on the application, other traffic types may be required for which transfer guarantees should be provided by the network.

15) https://www.iiconsortium.org/pdf/IIC_TSN_Testbed_Char_Mapping_of_Converged_Traffic_Types_Whitepaper_20180328.pdf
16) Messages without transfer guarantee that can use the remaining free bandwidth

5.5 Quality of service (QoS)

Quality of service (QoS) is a general term for certain data on a network that should be specially marked and treated according to defined rules (better than best-effort). For layers 2 and 3 of the ISO/OSI model, there are already QoS mechanisms used for classifying and prioritizing data streams (e.g. for prioritizing Voice over IP). TSN QoS refers to mechanisms newly specified by the IEEE TSN working group for regulating data transfer on layer 2 of the ISO/OSI model (for example 802.1Qbv or 802.1Qch). Classification is carried out via a VLAN tag[17)] that adds an additional 16 bits to the Ethernet header. 12 of these bits represent the VLAN ID (VID) and 3 bits represent the priority code point (PCP). The 16th bit is not used.

PCP allows 8 priorities to be mapped, which are used by a switch to sort the frames to be forwarded into up to 8 queues on each output port (egress port, in contrast to the ingress port for the input side). These queues are then normally emptied according to their priority (7 as the highest priority first; when this queue is empty, then priority 6, etc.), and the frames are sent via the corresponding egress port. This is referred to as "strict priority transmission selection". Via this priority control, Voice over IP, process data traffic on some Industrial Ethernet networks or alarms are treated with a higher priority than best-effort traffic.

The VID is used to virtually segment Ethernet networks. An Ethernet interface must be configured to participate in certain VLANs; otherwise, incoming tagged traffic is discarded. A popular VLAN variant is port-based VLAN, where a managed infrastructure switch is configured to assign all devices connected to a specific port to a VLAN. Traffic between devices in this segment is unaffected (e.g. if multiple devices are connected in line on this port). When frames from the segment are forwarded to the switch, however, the switch automatically adds a VLAN tag before forwarding. Devices outside the segment would also need to be assigned to this VLAN ID in order to communicate with devices in the VLAN. Communication is only possible between stations in the same VLAN (identical VLAN ID); otherwise, these devices are "invisible".

The concept of TSN streams is a logical evolution of the VLAN concept. A stream must be configured with a stream reservation method for all switches through which it should be sent. Otherwise, it will be discarded. A stream forms a VLAN for a single frame (unidirectional data transfer from a transmitter (TSN "talker") to one or more receivers (TSN "listener")).

[17)] See also https://de.wikipedia.org/wiki/IEEE_802.1Q

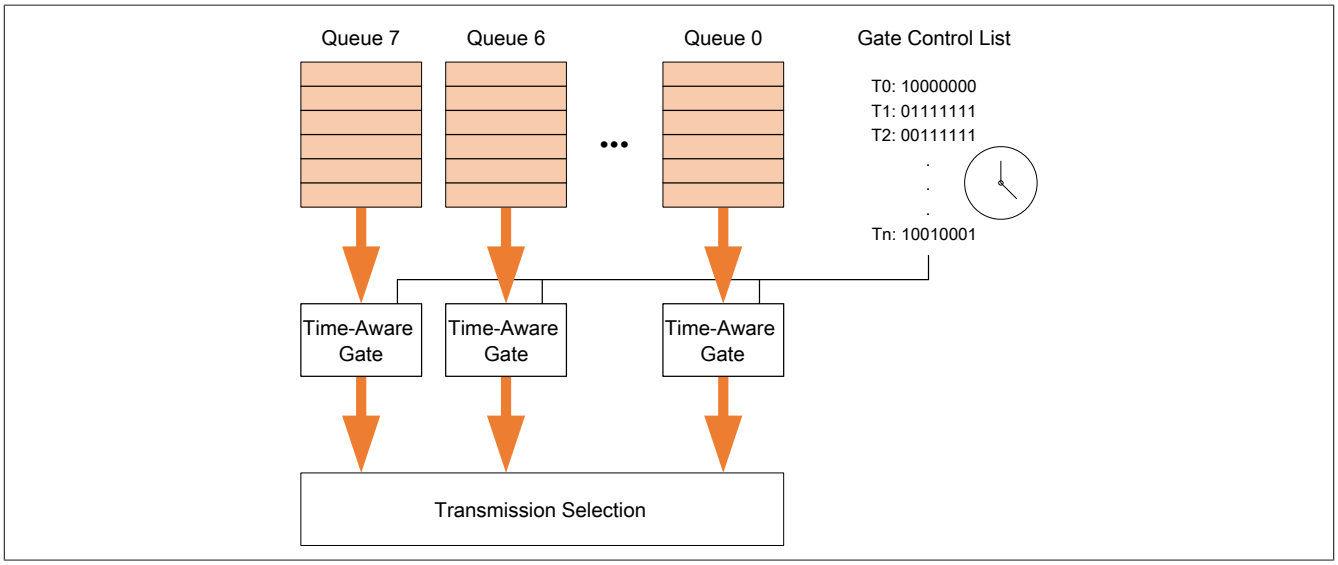
5.6 Mechanisms

The TSN working group deals with a variety of different mechanisms to meet the objectives in the above areas. Only a few are relevant for industrial applications as described in the following chapters. A network could theoretically be defined exclusively using either ingress or egress shaping, which is why all of the corresponding standards have a relatively large range of functions. In standardized TSN usage[18], however, it is agreed that a profile with both ingress and egress mechanisms makes the most sense and that some optional features of the standards do not need to be used.

5.6.1 IEEE 802.1Qbv

In addition to time synchronization, IEEE 802.1Qbv[19] is the core of TSN and the best-known standard. IEEE 802.1Qbv is also referred to as scheduled traffic and specifies the time-aware shaper (TAS). The TAS essentially defines timed-controlled opening and closing of switch queues on an egress port of a switch. These are referred to as Qbv windows that are configured for queues. The TAS precedes the actual selection of the frame to be transmitted and can therefore be used with familiar selection mechanisms such as strict priority (see section 5 "TSN") and credit-based shaper (see section 5.6.2 "IEEE 802.1Qav").

Without TAS, frames intended for a certain egress port are routed to the corresponding queue for delivery. If several queues have frames waiting, they are taken and sent based on their priority. Using TAS, a cycle is introduced within which the queues can be opened or closed individually (gate control list, GCL). If several queues are open (set to "1" in the GCL), the available frames are taken and sent based on their priority.



If only one queue is open at a time, this is referred to as "exclusive network access". This gives traffic in the respective queue reserved bandwidth on the corresponding port without potential interference from traffic in the other queues on the same port (inter-class interference). If the queues of this exclusive traffic class are aligned accordingly on all switches on a network, frames of this traffic class can be transferred without additional delay (intra-class interference), which makes it possible to achieve hard real-time capability and the shortest possible latency.

Information:

IEEE 802.1Qbv is used in the B&R system to enable isochronous real-time traffic without affecting other traffic. In addition, IEEE 802.1Qbv is used to reserve bandwidth for other traffic types.

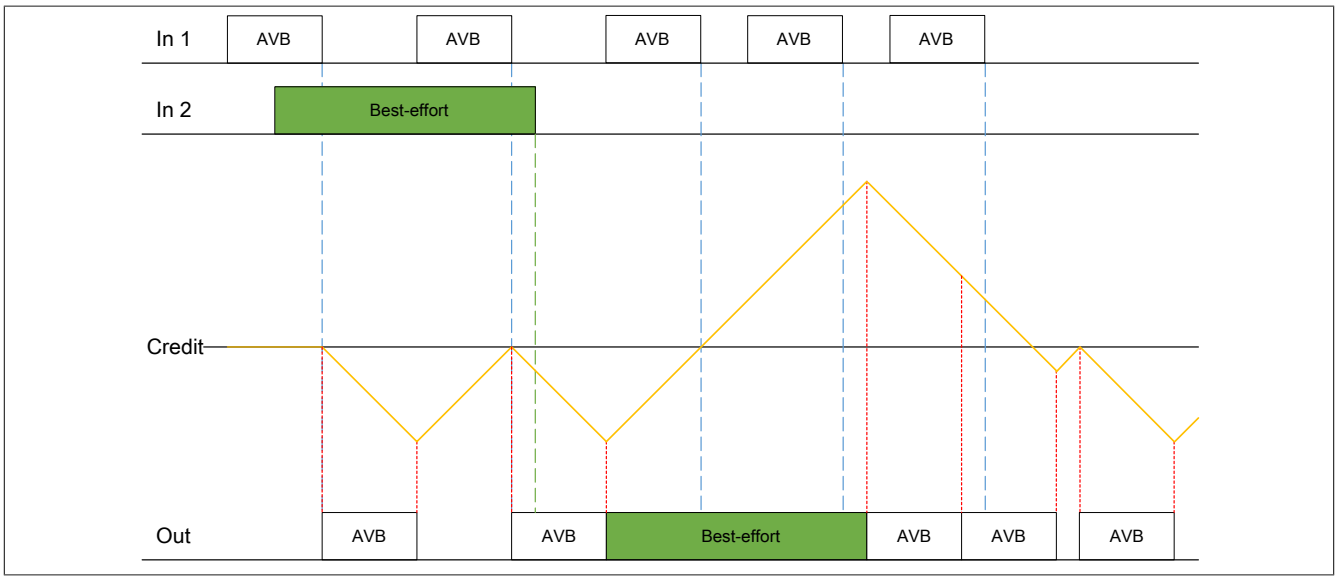
[18] TSNIEC/IEEE 60802 and OPC Field Level Communications Initiative
[19] Integrated into the 802.1Q standard and not available separately

5.6.2 IEEE 802.1Qav

IEEE 802.1Qav[20)] was developed by the AVB working group back in its time. This standard specifies the credit-based shaper (CBS). CBS can be used to achieve several goals:

- Bandwidth limitation of a transmitter
- Guaranteed maximum latency for sent frames
- Defined burst behavior of a transmitter

The credit-based shaper is implemented in such a way that a certain amount of credits is required for sending a frame depending on its size (≥0). If no more credits are available (< 0), a frame is not permitted to be transmitted. If no frames are transmitted, the egress port "earns" new credits up to a limit of 0. If the egress port is currently blocked by another transmitted frame and Qav frames are in the queue (i.e. cause a jam), the CBS "earns" additional credits on the port (up to another maximum) so that it can reduce the backlog of frames in its queue and utilize its allocated bandwidth over time. In addition to the maximum frame length, this maximum defines how long a CBS queue is permitted to continuously occupy the network (burst). Typical use cases for IEEE 802.1Qav are audio/video and events.



Information:

IEEE 802.1Qav is available in the B&R system for scheduling audio, video or other traffic on the network in a customized manner.

20) Integrated into the 802.1Q standard and not available separately

5.6.3 IEEE 802.1Qci

IEEE 802.1Qci[21)] is a comprehensive standard that can be used to define filter rules for discarding incoming traffic. IEEE 802.1Qci is also referred to as "ingress policing". IEEE 802.1Qci defines rate-based and time-based filtering on a stream and port basis.

If corresponding egress port mechanisms are additionally configured in a network, IEEE 802.1Qci is "only" used as a mechanism for increasing robustness and diagnostic capability. IEEE 802.1Qbv can be configured on a switch on the egress port, and IEEE 802.1Qci can be configured on the next switch directly connected on the ingress port. Setting the filter rules in such a way that the function of IEEE 802.1Qbv is checked on the neighbor switch, i.e. frames that do not meet the requirements are filtered, is referred to as "reverse Qbv".

IEEE 802.1Qci can also be used without IEEE 802.1Qbv to check incoming traffic, for example with "reverse Qbv". The aim is to protect its egress queues from unwanted traffic. The difference is that with IEEE 802.1Qci, frames that meet the criteria of a filter rule are discarded. With IEEE 802.1Qbv on the egress port, the frames remain in the queue if the gate is closed and wait for the next gate to open. This can theoretically cause problems over many cycles.

If all connected devices in the Qbv-Qci example above always show correct behavior, the IEEE 802.1Qci rules will not be triggered. Assuming that a device can send 2 frames (instead of one) in one cycle due to an error, however, the Qbv windows would need to be dimensioned accordingly larger in order not to keep the invalid frame permanently in the queue and disturb subsequent cycles. With IEEE 802.1Qci on the next switch, the excess frame transmitted can be detected and discarded. Another use case for IEEE 802.1Qci is bandwidth limitation. This can protect the network from too much traffic from outside or an invalid station or a station without its own bandwidth limitation. DoS attacks can also be mitigated or prevented altogether in this way.

Information:

IEEE 802.1Qci is used in the B&R system to make isochronous real-time traffic more robust against errors and to be able to specifically localize errors. In addition, IEEE 802.1Qci is used to limit the bandwidth of otherwise unconfigured (open) ports in the system.

[21)] Integrated into the 802.1Q standard and not available separately

5.6.4 Configuration (IEEE 802.1Qcc)

IEEE 802.1Qcc defines 3 different configuration models for TSN networks: centralized, distributed and hybrid. The distinction is based on where on the network the knowledge about the reserved streams is stored.

In the distributed model of the direct evolution of the configuration of AVB streams, both talkers and listeners report their needs for streams to their respective neighbors. This information is propagated across the network until at some point the requests of at least one talker and listener match. Each switch then provides information about whether it can additionally reserve the requested stream with the requested QoS parameters on this path. If all switches confirm, the stream is established and can be used. This method is easy and efficient to implement, especially for CBS traffic (see section 5.6.2 "IEEE 802.1Qav"). For other traffic shapers, the distributed configuration reaches its limits, which is why the other two models were developed.

The centralized model introduces a central instance, the central network configuration (CNC), that knows the topology of the network and the current stream configuration. An additional required stream can be requested from the CNC, which has all the information to confirm or deny the request and reconfigures all infrastructure components. The CNC is especially suitable for scheduling frames with hard real-time requirements on the network. In addition, the CNC can also efficiently manage other mechanisms across the network, for example preemption (see section 5.7.2 "Preemption") and per-stream filtering and policing (see section "IEEE 802.1Qci" on page 29).

Direct communication with a CNC for each application protocol is not efficient since there may be many different mappings of the application's QoS understanding of the network QoS available. This is why in addition to the CNC, the role of the central user configuration (CUC) was introduced in Qcc. The CUC is specific to the application protocol used on the network. For OPC UA PubSub communication, the PTCC was specified as a TSN CUC, for example. Each CUC can be addressed via the usual configuration mechanisms of "its" application protocol (the PTCC via OPC UA client/server) and translates these requests into requests to the CNC[22)].

The hybrid model combines the central instance of the CNC with stream requests to the neighboring switches. This approach is not used in practice.

Central configuration is used in the B&R system, but the network configuration is calculated by Automation Studio and the result is distributed across the network at runtime by Automation Runtime.

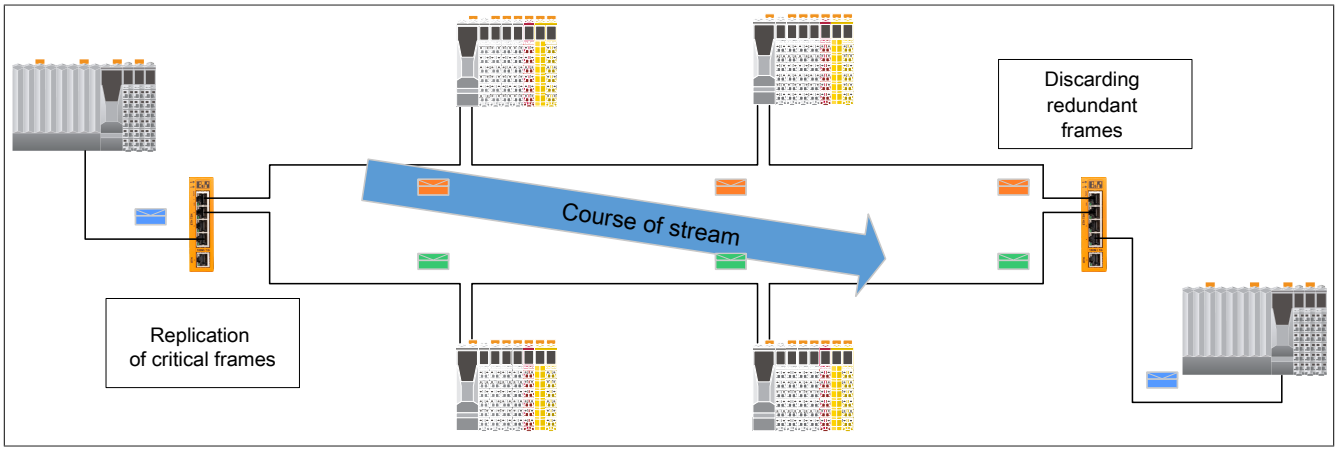
Information:

IEEE 802.1Qcc is used in the B&R system to configure the TSN parameters in the devices.

22) To configure a PubSub frame, publisher and subscriber are configured directly. To configure the "same" PubSub over a TSN stream, the same request (extended with the QoS requests) is transmitted to the CUC. It requests the necessary network resources from the CNC, which are then also reserved by the CNC. The CUC then configures the publisher and subscriber, each extended by the required QoS parameters (e.g. stream ID).

5.6.5 IEEE 802.1CB

IEEE 802.1CB defines mechanisms for duplicating streams and merging redundant duplicates to enable bumpless, redundant transfer of streams on a network. The mechanisms are transparent to any devices that do not support this standard, i.e. any switch on the network that does not support this standard will forward the redundant copies in the same way as normal streams. A simple example is a central switch to which a controller is connected on one port and a ring of field devices on 2 other ports. The switch can replicate all (configured) streams from the controller to both ports, and the field devices will each process the first copy received and discard the second (if a second is received). Conversely, the field devices can also replicate their streams to both ports, and the central switch can then forward the first copy to the controller in each case (and discard further copies). Since only duplication and merging are configured in each case, a user can define any redundancy topologies and extend them with switches. The image shows an example with 2 infrastructure switches configured with CB.



The advantage of CB lies in its use with streams and its preconfiguration, i.e. there are no switchover times in the event of ring breakage or recovery. In addition, only those messages can be transferred redundantly for which availability without packet loss is more important than the additional bandwidth required for the duplicates on the network. Non-stream traffic can/must be handled using other mechanisms. In industrial applications, the switchover times of spanning tree variants are completely sufficient except for cyclic process data. Short-term packet loss (until a new spanning tree is built) is compensated on the transport layer of the TCP by repeated transmission.

In special applications, other data in addition to the process data can have high availability requirements (events, configuration). In these cases, these streams can be designed redundantly. CB therefore provides a minimally invasive, strategic mechanism for redundancy.

In case of redundant connections, cyclic process data is transferred redundantly by default in the B&R system.

Information:

IEEE 802.1CB is used in the B&R system for bumpless redundancy of critical process data.

5.7 Optimizations

5.7.1 Cut-through

Ethernet is designed in such a way that frames are completely received, analyzed, traffic-shaped and forwarded accordingly by a switch. This behavior allows each switch to analyze the checksum (at the end) of a frame to discard messages corrupted by bit errors. Since the frame size is not known until a frame is completely received, many IEEE 802.1Qci filters can only be operated meaningfully with this information.

This store/forward behavior, however, induces a latency in each switch that corresponds approximately to the transfer time of one frame on the cable. A maximum Ethernet frame requires approx. 127 µs at a bandwidth of 100 Mbit or approx. 12.7 µs with Gigabit Ethernet to be completely received by a switch. If the frames on the network are all of comparable size, the bandwidth can also be used very well with this behavior (since one frame can be forwarded and another received at the same time). If the frames differ noticeably in size, however, shorter frames "bump into" longer frames on the network or, conversely, gaps are created by switches forwarding shorter frames faster than longer frames.

For many closed-control loop applications in large industrial networks, the accumulated latencies in the store/forward operation of the network would conflict with the targeted cycle time or response time.

There is therefore the alternative "cut-through" operating mode, where a switch makes the forwarding decision as early as possible, i.e. after receiving the destination address or TSN stream ID, and immediately passes on the frame to the egress port for forwarding. The accumulated latency in this operating mode is minimal, but it may be that corrupt frames are forwarded or that specific filter rules are violated in individual cases (with bandwidth filters). The network must be designed accordingly to handle the disadvantages of cut-through switching in a different way.

Cut-through switching[23)] is used in the B&R system for deterministic real-time communication. Corrupt frames may spread on the network, but are discarded at the receiver. Qci is used for this traffic type to check the transmitting or forwarding time. The frame length information is therefore not needed. All other traffic types are handled via store/forward. The latency of B&R devices is <800 ns for Gigabit Ethernet and <3 µs for 100 Mbit.

Information:

Cut-through switching is used in the B&R system for minimum latency with isochronous real-time traffic <800 ns per hop[24)] with Gigabit Ethernet.

[23)] Cut-through switching is implemented by the B&R field devices, not by the TSN switch that operates in store/forward.
[24)] The specified <800 ns includes PHY delays and applies to B&R field devices, not to the TSN switch.

5.7.2 Preemption

IEEE 802.1Qbu for switches in conjunction with 802.3br for devices (single ports) is referred to as preemption and defines a method for forwarding high-priority frames through the network faster by interrupting low-priority frames on the egress port. In case a short frame "bumps into" a longer one as described above, the short frame, if configured accordingly, would not experience the longer latency of the longer frame on every switch but could overtake it. Preemption is configured between every 2 network neighbors and can therefore be used very specifically on the network.

Each of the 8 queues on an egress port is configured either as "express" or as "preemptive". If a preemptive frame is being transmitted and an express frame arrives at the egress port for delivery, the preemptive frame can be interrupted (and marked accordingly) every 64 bytes and the express frame can be forwarded promptly. The started fragment is stored on the neighbor until all further fragments have been received. Only then is it processed further.

Preemption can be used to achieve real-time behavior without stream-based network scheduling in an easy way since one frame per switch can experience a maximum additional latency of 127 bytes (the largest non-interruptible frame) on its path if only one express traffic type is used.

Preemption is not used in the B&R system since real time is ensured via IEEE 802.1Qbv without additional latencies. B&R devices with TSN switches, however, offer preemption so that they can be used in other environments.

Information:

Preemption is available in the B&R system for customized use.

6 OPC UA field-level communication

Sections 6.1 to 6.3 have been taken from the website of the OPC Foundation[25)].

6.1 Background

The goal of digitalization is to foster the integration of IT technologies with OT products, systems, solutions and services across their complete value chains which stretches from design and production to maintenance. Once implemented, digitalization opens the doors to new business opportunities like the digitalization of products and systems, new and enhanced software solutions, and new digital services.

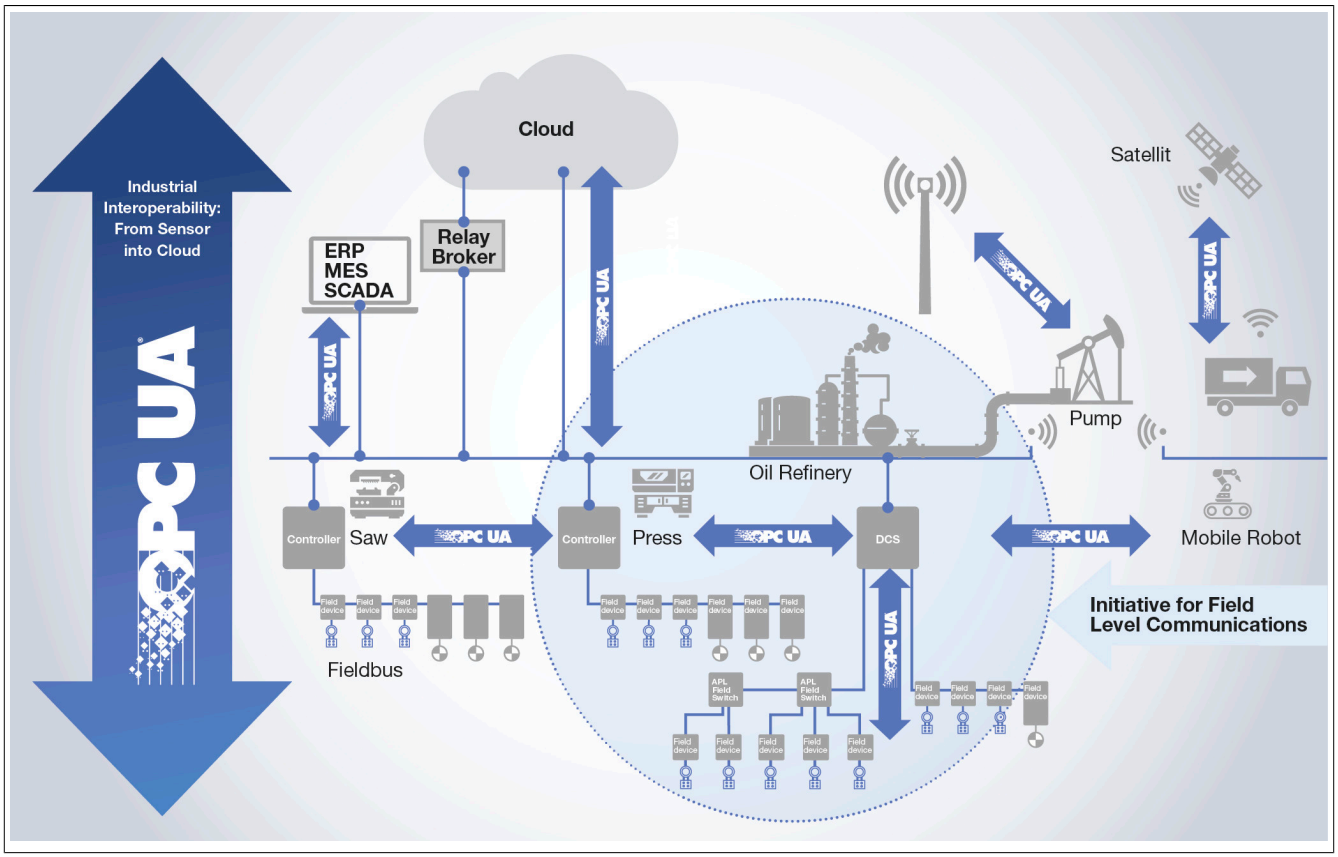
The Internet of Things (IoT) brings together a broad range of technologies that have traditionally not been connected via today's near ubiquitous IP-based networks. While Ethernet provides the ability for things to "reach" each other, they still need a common way to communicate. Standardized data connectivity and interoperability addresses this need.

In simple terms, with standardized data connectivity at its core, the Industrial IoT (IIoT) can be looked at from two perspectives: horizontal and vertical data connectivity. An example of horizontal communications is Machine to Machine (M2M) data connectivity between shop floor systems. An example of vertical communications is device-to-cloud data transfer.

In both cases, the OPC UA standard from the OPC Foundation provides a secure, reliable and robust foundation to facilitate standards-based data connectivity and interoperability. For years many companies and partner organizations have openly worked together under the umbrella of the OPC Foundation to make this a reality and will keep doing so as it continues to expand its collaboration activities.

A key aspect of improving horizontal and vertical data connectivity is network convergence supporting a common network for IT- and OT-related communication. With Ethernet Time-Sensitive Networking (TSN) according to IEEE 802.1 not only communication with bounded latency and jitter is being supported. In addition, various data streams and traffic types can be transmitted over a common network infrastructure, while at the same time guaranteeing the various bandwidth, latency, jitter and reliability requirements of the different applications. Therefore, TSN plays a key role in supporting the convergence of IT and OT. The Ethernet Advanced Physical Layer (APL) is another key technology to drive network convergence as APL delivers seamless Ethernet connectivity to sensors and actuators in process automation – including hazardous areas.

25) https://opcfoundation.org/wp-content/uploads/2020/11/OPCF-FLC-Technical-Paper-C2C.pdf



6.2 FLC Initiative

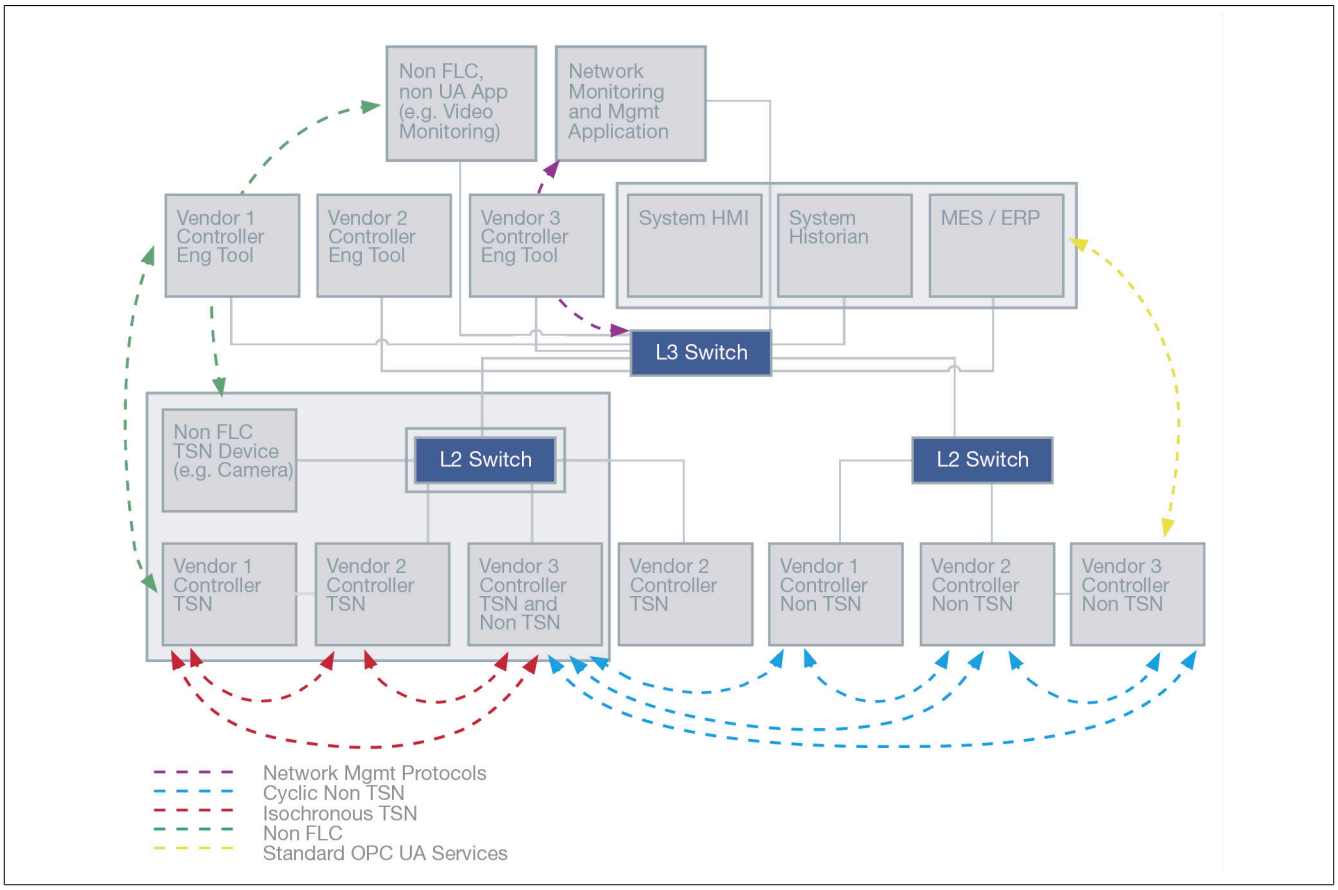
At the SPS IPC Drives Fair 2018 in Nuremberg, Germany the OPC Foundation officially launched the Field Level Communications (FLC) initiative. This initiative aims at extending OPC UA to field level to achieve an open, unified, standards-based IIoT communication solution between sensors, actuators, controllers and cloud addressing all requirements of factory automation and process automation (see figure in section "Background" on page 34). Consequently, the OPC Foundation vision of becoming the worldwide industrial interoperability standard is advanced by integrating field devices and by extending OPC UA to the field level. Vendor independent end-to-end interoperability into field level devices is provided for all relevant industrial automation use cases including real-time, functional safety and motion, requiring secured information exchange.

6.3 System Architecture Outline

The FLC-related technical work includes the following topics:

- Definition of a base model for automation components that are common to all FLC conformant controllers and devices
- Definition of system behaviors and sequences for common functionalities e.g. bootstrapping, connection establishment, etc.
- Harmonization and standardization of application profiles like IO, motion control, functional safety, system redundancy
- Standardization of OPC UA information models for field level devices in online and offline scenarios, e.g. device description, diagnostics…
- Support of Ethernet TSN for deterministic communication and IT/OT convergence
- Mapping of application profiles related to real-time operations on Ethernet networks including TSN
- Definition of conformance units/profiles that can be tested to guarantee interoperability across vendors
- Definition of certification procedures

In the first specification release (Version 1), the focus is on the Controller-to-Controller (C2C) use case which includes exchanging both standard and safety real-time data using OPC UA Client/Server and OPC UA PubSub in combination with a peer-to-peer application relationship and basic diagnostics. The target network architecture is shown in Figure.


Non FLC, non UA App (e.g. Video Monitoring)
Network Monitoring and Mgmt Application
Vendor 1 Controller Eng Tool
Vendor 2 Controller Eng Tool
Vendor 3 Controller Eng Tool
System HMI
System Historian
MES / ERP
L3 Switch
Non FLC TSN Device (e.g. Camera)
L2 Switch
L2 Switch
Vendor 1 Controller TSN
Vendor 2 Controller TSN
Vendor 3 Controller TSN and Non TSN
Vendor 2 Controller TSN
Vendor 1 Controller Non TSN
Vendor 2 Controller Non TSN
Vendor 3 Controller Non TSN
Network Mgmt Protocols
Cyclic Non TSN
Isochronous TSN
Non FLC
Standard OPC UA Services