HIRSCHMANN
A BELDEN BRAND

White Paper

TSN – Time Sensitive Networking

Dr. René Hummen – CTO Office,
Hirschmann Automation and Control GmbH

Stephan Kehrer – CTO Office,
Hirschmann Automation and Control GmbH

Lukas Wüsteney – CTO Office,
Hirschmann Automation and Control GmbH

Time-Sensitive Networking (TSN) adds a level of determinism to Ethernet-based data transmissions that was previously not possible with conventional Ethernet technology.

Ethernet networks are now able to provide:

- Predictable, guaranteed end-to-end latencies
- Highly limited latency fluctuations (jitter)
- Extremely low packet loss

From a standardization point of view, key TSN features already are an integral part of basic Ethernet according to IEEE 802.1 and IEEE 802.3 for quite some time now. In addition, this next step in Ethernet evolution is increasingly becoming available in current FPGA IP cores and ASICs. Therefore, now is the right time to get an overview of the most important TSN functions and their advantages in demanding automation networks. The goal of this white paper is to provide exactly that.

Table of Contents

Be certain.
Belden.

Real-time communication today and in the world of the Industrial IoT

Latency guarantees are a basic prerequisite for the transmission of process data with real-time requirements in a number of application fields. These include, in particular, synchronized drive technology, automation of control technology in power generation and transmission and transportation networks. In these application fields, the cycle times for the transmission of time-critical data are, in some cases, significantly less than 1 ms. In order to achieve such cycle times with corresponding latency guarantees, usually real-time communication methods such as EtherCAT, Profinet IRT or SERCOS III are used. Although these methods are based on conventional Ethernet, they all extend it separately in an incompatible form to guarantee latency boundaries. The value proposition of TSN is a uniform physical layer and data link layer that is standardized by the IEEE. It can be used by the entire real-time communication market with corresponding economies of scale and reduced efforts on the part of the implementers and users.

In addition to these fields of application with "firm" real-time requirements, other areas, such as process automation, also benefit from TSN. In these cases, cycle times are usually significantly longer than, for example, in synchronized drive applications. However, applications in this field also frequently require guaranteed end-to-end latency or dedicated bandwidth. In current networks, these guarantees are typically implemented through flat overprovisioning. With TSN, it becomes possible to move away from such approximate solutions and to provide and guarantee precisely the required bandwidth, based on the process data that needs to be transferred. Thus, TSN is the tool that permits planning and dimensioning of future automation networks according to the actual bandwidth requirements and latencies to be expected.

In general, it can be observed that multiple parallel networks, installed and used for different applications in a production facility are converging into singular multi-service networks. While in current plants, time-critical control information is often transmitted via dedicated networks, customers are increasingly expressing the wish that this critical process data, together with "best-effort" data (e.g. configuration and monitoring information) and data with "soft" real-time requirements (e.g. video data from surveillance cameras) should be transmitted via a single common network. TSN offers a suitable solution for converged network infrastructure with high bandwidth requirements on the one hand and hard and soft real-time requirements on the other hand.

When having a look at the automation networks of the future, the pivotal importance of TSN is also evident here. Even today, industrial automation is undergoing a transformation that is driven by the idea of building production facilities that are much more flexible, intelligent and able to cope with more dynamic changes. The terms frequently used in this context are "Industry 4.0 (I4.0)" and "Industrial Internet of Things (IIoT)". They describe the idea of an intelligent production environment, in which production machines, conveyors and work pieces are constantly communicating with each other to automatically guide and support the production process. This is made possible by increased inter-networking of the sensors and actuators involved in production and by an increased integration of the (local) cloud infrastructure. Through this, virtual programmable logic controllers (PLCs) are directly integrated into the production processes. These changes also affect the basic models on which automation networks are developed and planned today. As shown in Figure 1, for example, the well-known automation pyramid is, from a network perspective,

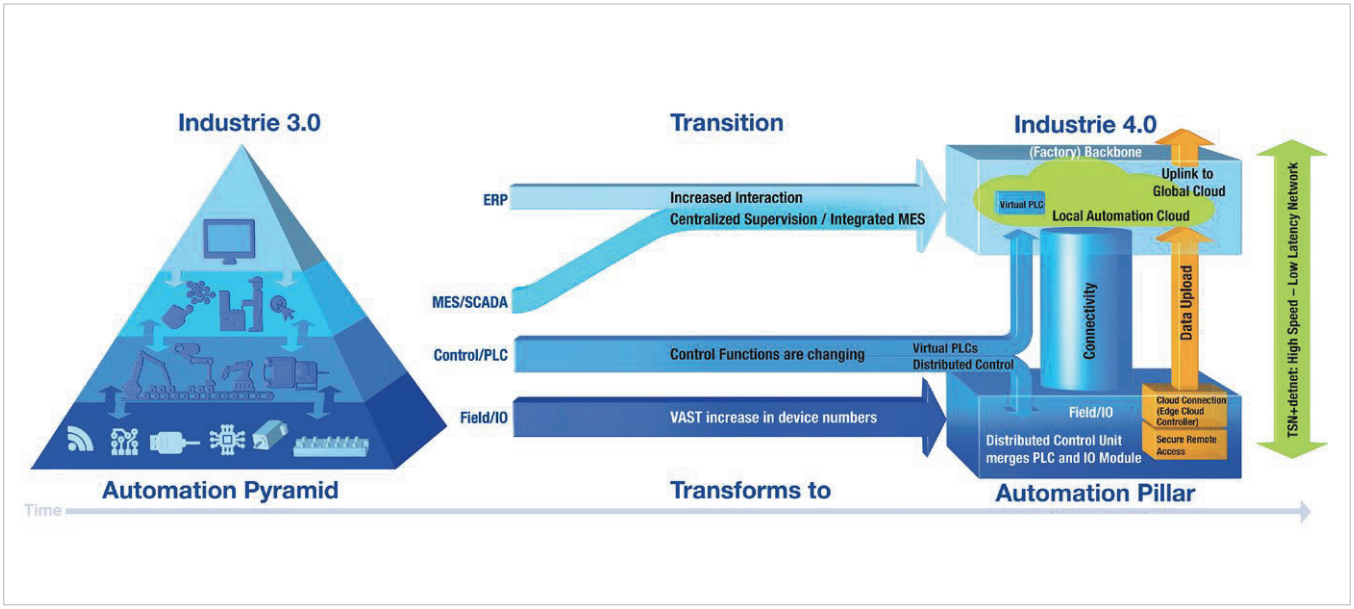

Figure 1: Transformation from the automation pyramid to the automation pillar in future automation networks

+65 6879 9800 | belden.com

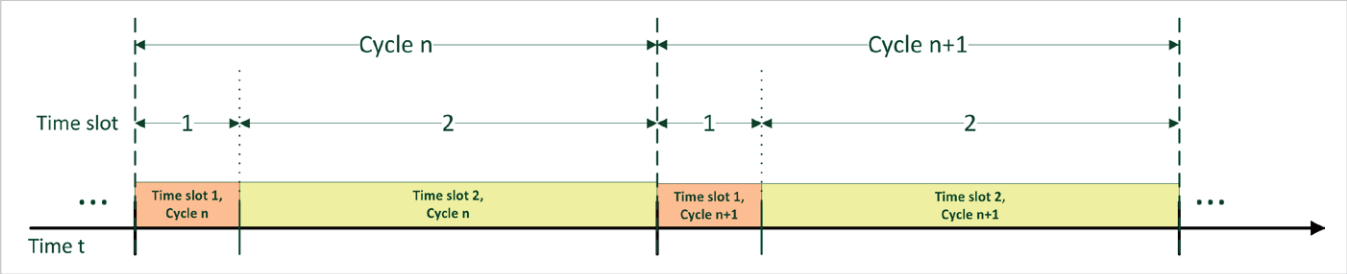

Figure 2: Time division multiplexing permits the reservation of time slots within a cycle in order to enable the timely transmission of periodic real-time data

developing into an automation pillar. This is a continuous long-term transformation process. In the context of the automation pillar, the full value of TSN becomes apparent: It enables the demand-oriented scaling of real-time mechanisms. It also allows for planning of the network topology and the bandwidth and, at the same time, retains investment security in the networks due to the backwards compatibility to the proven, existing Ethernet technology.

TSN – Mechanisms and interdependencies

TSN extends Ethernet-based data communication with novel mechanisms to provide determinism. This is done on such a level of performance that TSN can satisfy even the most demanding requirements of modern control networks, e.g. in industrial automation, automotive manufacturing and in-vehicle networks. Even today, it is foreseeable that some of TSN's target markets will differ significantly from each other concerning their requirements. For example, a deterministic and, at the same time, fault-tolerant data transmission may be absolutely necessary in one application, but in another case, the fault tolerance requirements may only be of secondary importance. To cater to these different requirements, TSN is designed as a modular system in which the exact characteristics of deterministic data transmission – and the associated hardware and software requirements – can be adapted to the respective requirements. Following this logic, TSN is not specified in one single standards document, but rather through a family of international standards that have been in development since 2012 at the IEEE 802 and its TSN Task Group [1]. By now – with only a few exceptions – all mechanisms of the TSN family are available as finished standards documents. In the following sections of the white paper, we will provide an overview of the most important TSN mechanisms and how these mechanisms interact with each other.

Prioritization based on timing with the Time-Aware Shaper

Until now, Class of Service (CoS) mechanisms such as the strict priorities according to IEEE 802.1Q have not been able to guarantee the forwarding of time-critical data traffic at a fixed point in time: A low-priority Ethernet frame already in the sending queue of a switch can delay the transmission of other data frames, even when they are tagged with the highest priority (i.e. priority 7). This is possible in each Ethernet switch along the transmission path. As one of the central components of TSN, the "Enhancements for Scheduled Traffic" introduce the possibility of prioritizing the data transmission of conventional Ethernet frames on a temporal level. For the first time with Ethernet, this

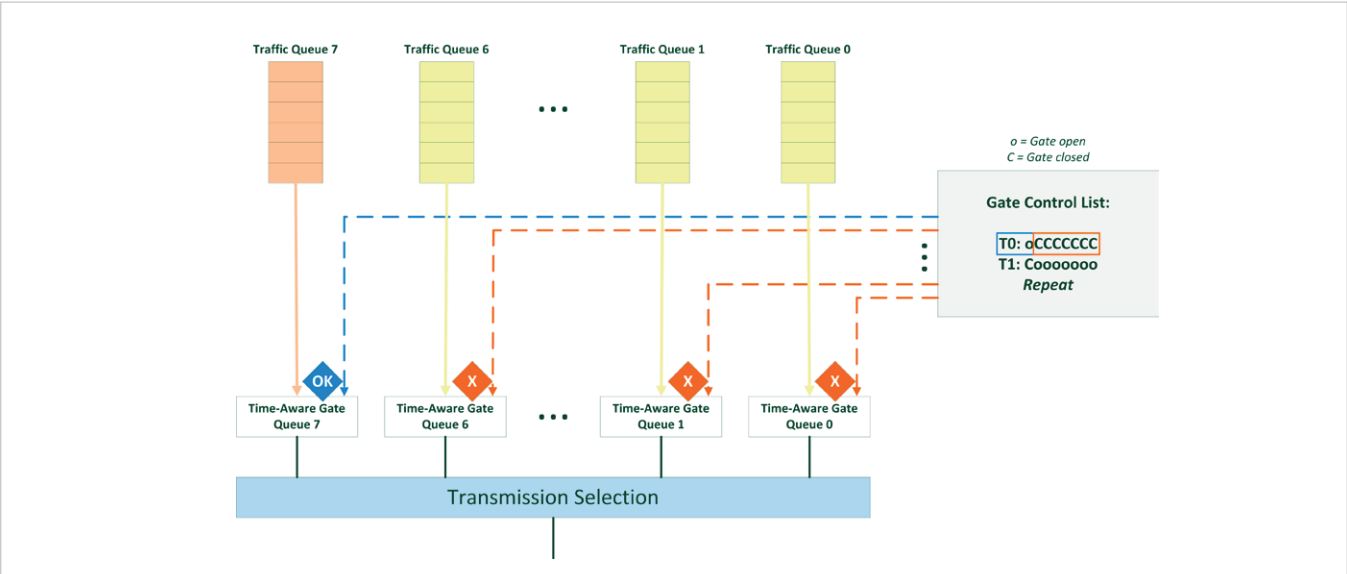

Figure 3: The Time-Aware Shaper (TAS) implements time-based prioritization via the newly-introduced Time-Aware Gates that are located between the CoS queues and the selection function for the packets to be sent

allows to guarantee forwarding at a fixed point in time. In this document, this mechanism is referred to as the Time-Aware Shaper (TAS).

The core idea of this TSN mechanism (IEEE 802.1Qbv-2015 [2]), which was published in March 2016, is to divide time into discrete sections. These sections of equal length, so-called cycles, can be further broken down according to the time slot method (TDMA – Time Division Multiple Access) as shown in Figure 2. This enables the allocation of data packets with real-time requirements to dedicated time slots within these cycles. In other words, the TAS can be used to temporarily interrupt the transmission of conventional best-effort Ethernet traffic in order to forward time-critical data traffic within specifically reserved time slots. The TAS thus enables a temporal preference of periodic real-time data over conventional best-effort data traffic (see "Time slot 1" in Figure 2).

Similar to the strict priorities that are common in Ethernet switches today, the TAS relies on the CoS Priorities (PCP - Priority Code Point) which are encoded in the VLAN tag of the Ethernet header. Ethernet frames are initially processed unchanged in the Ethernet switch until they reach the queues (traffic queues) at the output port. At this point, the TAS intervenes in packet processing through the newly introduced Time-Aware Gates, as shown in Figure 3. To be more precise, the selection of the next Ethernet frame to be transmitted is no longer performed exclusively based on a strict ranking of the queues when using TAS, but the state of the respective Time-Aware Gate is also taken into account. This can be either open or closed. Based to this time-dependent state, Ethernet frames in the associated queue are taken into account when queued packets are selected for transmission. For example, at the point in time that is shown in Figure 3, only the queue associated to Priority 7 is served.

The Gate Control List (GCL) determines which traffic queue is allowed to send at a certain point in time within the cycle. In addition to the states of the Time-Aware Gates, the Gate Control List specifies the time period for which a particular entry is active. Going back to the example in Figure 3 on the right hand side, this GCL reflects the cycle from Figure 2, consisting of a phase with time-prioritized data traffic (Time slot 1) and a best-effort phase (Time slot 2).

The necessity of guard bands

Due to the very poor predictability of the traffic patterns of Best-Effort data transmissions, it is normally not foreseeable when such data packets will be transmitted. Accordingly, as shown in Figure 4, the transmission of a best-effort packet could begin during time slot 2, right before the opening of time slot 1. The transmission of this frame must be completed before the transmission of the time-critical frames in time slot 1 can be started. Figure 4 shows that despite the use of the TAS in this case, the best-effort packet would extend into the time slot 1 of the following cycle. This would delay the transmission of the time-critical real-time data and could result in a violation of the end-to-end latency guarantees.

In order to avoid such situations, so-called guard bands are introduced with TAS, together with the time slots. These guard bands prevent the forwarding of frames for the length of a maximum-size Ethernet frame, right before the transition from one time slot to the next. This is done by explicitly configuring an additional time slot in which all gates are closed – for as long as it would take to transmit the maximum-size frame. Thus, the guard band can prevent the transmission of a

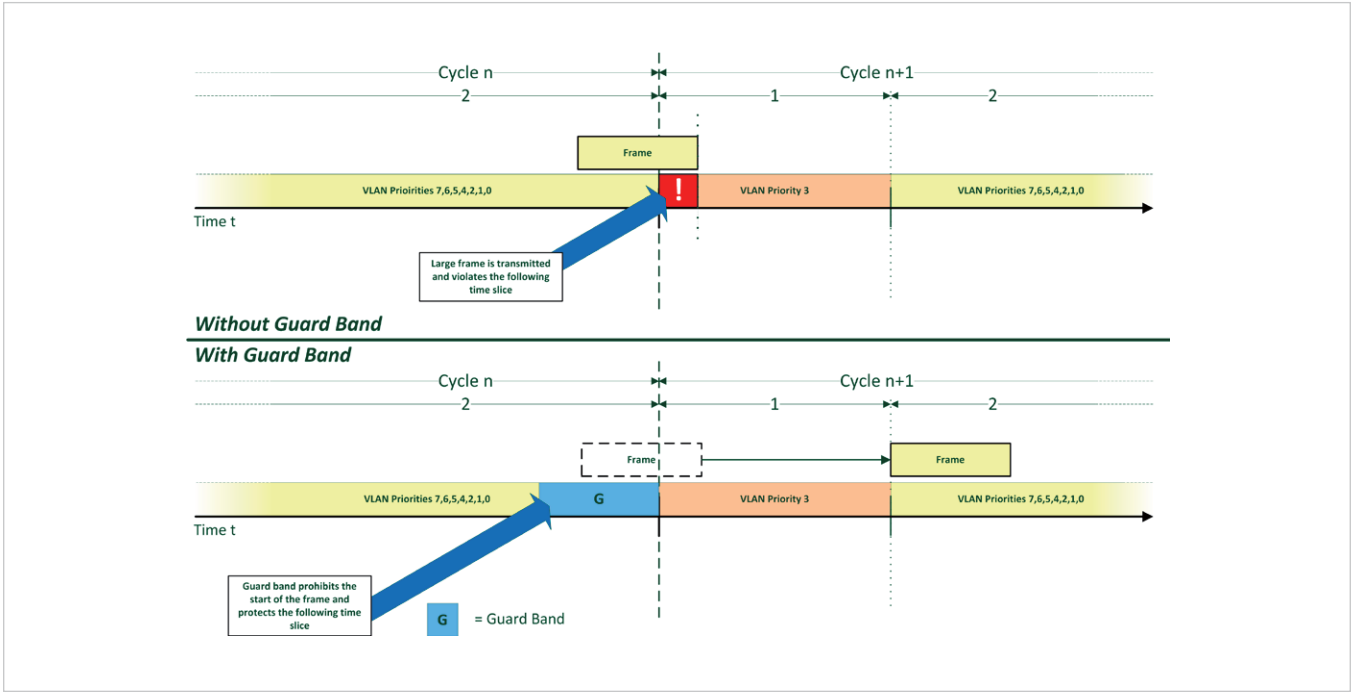

Figure 4: The guard band in TSN prevents best effort frames from extending into a time slot that is reserved for real-time data, but it decreases the available bandwidth

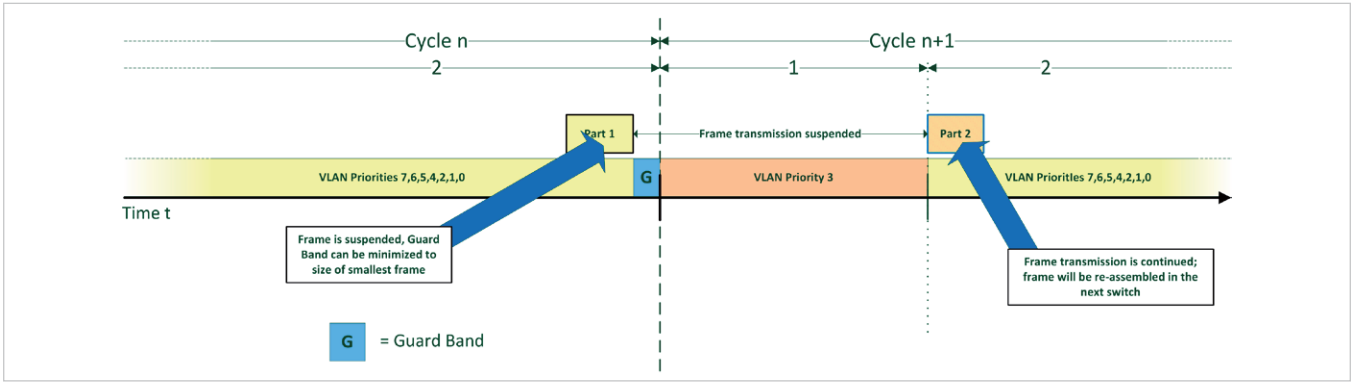

Figure 5: With the method of Ethernet frame pre-emption, the guard band size can be reduced from the maximum size of an Ethernet frame to the size of a partial packet

best-effort Ethernet frame and can stop it from violating the subsequent time slot, as shown in Figure 4. At the same time, however, a guard band inevitably leads to unwanted downtime in the network, as nothing can be transmitted, and thus to a waste of bandwidth.

As an alternative to the explicit guard band, the Transmission Selection mechanism (see Figure 3) can additionally take the packet length of an Ethernet frame into account. This works as long as the size of the packet is known at the point in time when the transmission decision is taken. The transmission decision therefore depends on whether the next packet in the queue still "fits" and can be completely transmitted – before the transition to the next time slot. Still, the situation may arise where a packet can no longer be transmitted if it is too large or if the remaining time in the time slot is too short. In this case, the time-critical traffic will not be disturbed, as the best-effort packet will be buffered until the next best-effort time slot starts, but it would experience a delay in transmission and the bandwidth that can effectively be used would be restricted.

Interruption of Ethernet frames

In order to maximize the bandwidth utilization for best-effort Ethernet frames, the IEEE 802 working group also developed a method for Ethernet frame pre-emption (IEEE 802.1Qbu-2016 [3], IEEE 802.3br-2016 [4]), which was completed in June 2016. Using this method, conventional Ethernet frames can be divided into small packet fragments of at least 64 bytes and each fragment can be transmitted separately. As shown in Figure 5, this enables the transmission of an Ethernet frame to be started within a best-effort phase, despite insufficient remaining time to transmit the whole frame. The transmission of the frame can then be interrupted and suspended at a multiple of the 64 byte limit before completion of the current time slot. The transmission is then resumed and completed at the beginning of the next best-effort phase. Frame pre-emption allows the guard band to be limited to the maximum size of the Ethernet fragments, with 64 byte as the minimum. With Fast Ethernet networks, for example, this results in a reduction in the "dead time" of up to 0.12 ms per transition from a best-effort phase to a phase with time-critical data traffic. This subsequently results in significantly less wasted bandwidth.

As a component of the TSN toolkit, Ethernet frame pre-emption can, of course, also be used separately from the TAS mechanism. In this case, for example, the CoS priority assigned to the time-critical control traffic can be configured as "express" – effectively as preferred traffic that is not to

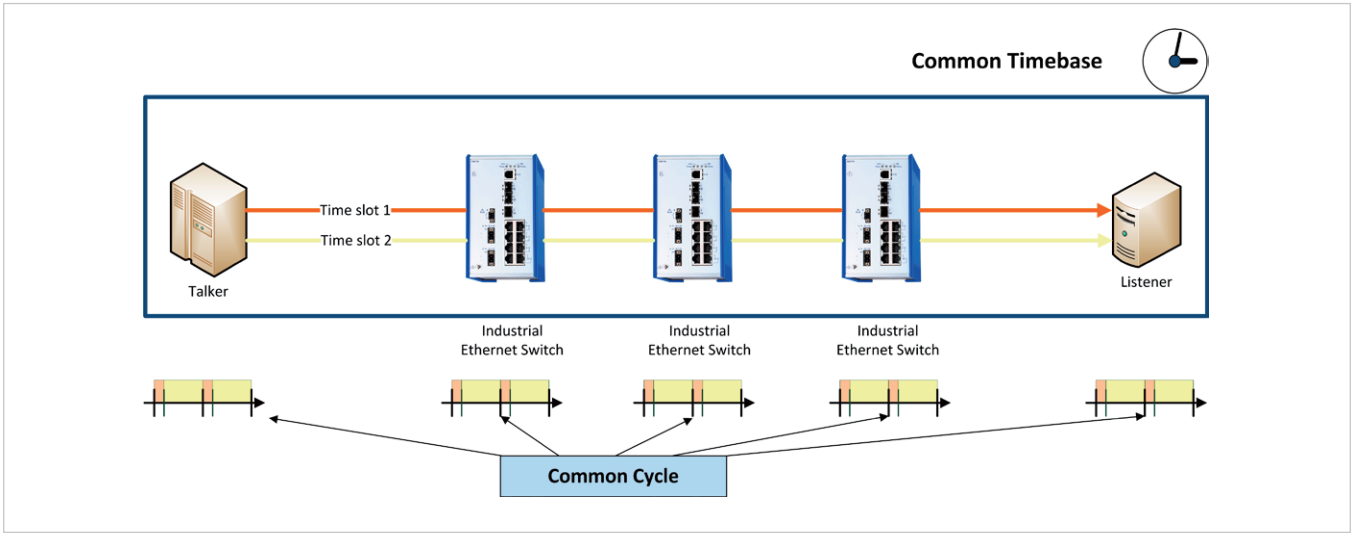

Figure 6: Precise time synchronization is a prerequisite for the TSN Time-Aware Shaper

be fragmented. The other priorities can similarly be configured as "pre-emptible", effectively data traffic that can be fragmented. As a result, a switch configured in this way interrupts the transmission of best-effort traffic as required when control traffic is present, thus reducing the transmission latency of time-critical control data. However, in this scenario, an increased volume of control traffic on the network decreases the prediction quality of the transmission latency compared to the method that includes TAS. This is due to the fact that multiple express frames can still interfere with each other, as known from conventional Ethernet.

Due to the fact that Ethernet frame pre-emption necessitates changes in the Ethernet frame structure, it is important to note that this mechanism is limited to links where both adjacent devices support the procedure. These two devices (e.g. two Ethernet switches) inform each other about their pre-emption support on the connected Ethernet ports by means of the Link Layer Discovery Protocol (LLDP) (IEEE 802.1AB-2016 [5]). Only with mutual support on both ends, pre-emption is activated on the corresponding end device or switch ports. This ensures backward compatibility with existing Ethernet devices.

Synchronous transmission cycles as a prerequisite

TAS operates only based on local device information – in other words the information available within a network device (bridged end-device or Ethernet switch). This information includes, for example, the cycle length and time slot periods. To ensure that data streams can actually be transmitted with guaranteed latencies over an end-to-end connection, unplanned waiting times must be avoided in all devices on the transmission path. To ensure this, the frames must "hit" the correct time slots on each device as they move through the network. In addition to the operation of TAS, this requires a close coordination between the devices in the network that participate in the transmission (see Figure 6). In particular, this means that all network participants must have a common notion of time. The participants must know when a cycle begins and which time slot is active during which period. To ensure this, the use of a time synchronization protocol such as the Precision Time Protocol (PTP) according to IEEE 1588 (IEEE 1588-2008 [6]) or the IEEE 1588 variant IEEE 802.1AS (IEEE 802.1AS-2011 [7]) is mandatory.

Both IEEE 1588 and IEEE 802.1AS enable the synchronization of distributed clocks within a network with an accuracy of 1 µs and below. When implemented with hardware support, accuracies in the range of a few nanoseconds can be achieved (Hirschmann PTP Whitepaper [8]). In contrast to protocols known from the IT environment, such as the Network Time Protocol (NTP), IEEE 1588 often does not aim for global time synchronization, for example, the synchronization with an atomic clock. Instead, the Best Master Clock (BMC) algorithm is used to determine the network subscriber with the most accurate, free-running clock. This device serves as a reference clock (grandmaster clock) against which the other network participants synchronize. With respect to TSN, this means that the time on all clocks in a network must be synchronized. The actual time, on the other hand, only plays a secondary role.

IEEE 802.1AS is a relatively new time synchronization protocol in the automation environment. It follows the same basic synchronization model as IEEE 1588 and originally was developed to limit the high number of configuration options of IEEE 1588 to exactly those parameters that are relevant in local area networks (LANs). IEEE 802.1AS, for example, limits the choice of transport technology and encapsulation to Ethernet, while IEEE 1588 provides additional UDP/IP encapsulation for use in wide area networks. In the course of TSN standardization, the existing IEEE 802.1AS protocol will be extended through the work in IEEE 802.1AS-Rev [9] by some additional mechanisms from IEEE 1588 that are required for use in automation networks. This is the case, for example, regarding the support for multiple synchronized time domains. This feature currently is available in IEEE 1588, but not in 802.1AS, and allows network participants to synchronize against a global clock (as with NTP) as well as a second time source for a working clock. This, for example, offers the possibility to use the globally synchronized clock for unique event logging, while the working clock can be used for the TAS. This has the advantage of not having to adjust to global time conventions – such as the leap second – in the working clock, which could interfere with synchronized local operations.

Since the current version of IEEE 1588 was already specified in 2008, this technology for time synchronization has already established itself in many markets and application areas. In some cases, special profiles have been developed for distinct fields of applications such as the energy market. In such cases, of course, IEEE 802.1AS is not a requirement the use of TSN. Rather, the TSN mechanisms do not mandate the use of a specific mechanism for time synchronization. Depending on the application, the use of IEEE 1588 with or without profiles can therefore continue to be preferred over 802.1AS as the solution for time synchronization. Regardless of the synchronization protocol used, however, it is necessary that the selected time synchronization mechanism offers a high degree of accuracy such that time slots of the TAS mechanism start and end at the right time on all devices in the network.

Traffic shaping with imprecise transmission timespans

In application fields such as process automation, periodic control processes are often executed. These will only sporadically lead to data transmissions, for instance in the case of an event occurring. Such event-based data transmissions can e.g. occur for the communication of state transitions of a process or for the transmission of aggregated measurement values when certain pre-determined time or value

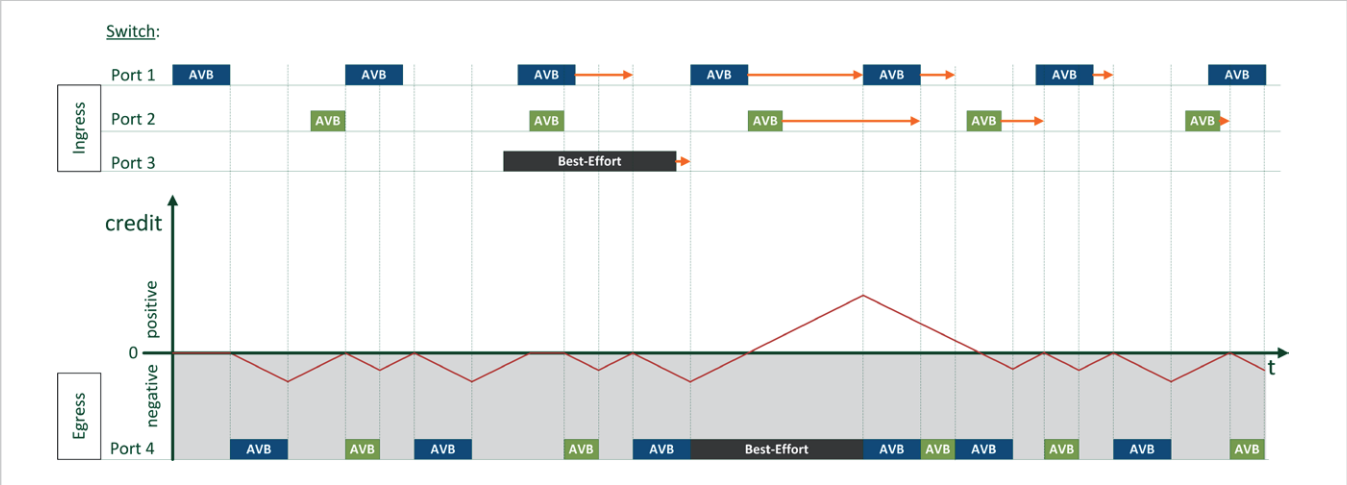

Figure 7: With the Credit-Based Shaper, data streams with reserved bandwidths are handled with higher priority than best effort traffic, as long as positive transmission credit is available

threshold are exceeded. Accordingly, it is not always possible to predict the exact time of communication in these scenarios. During the data transmission however, clearly defined latency limits should typically be adhered to in order to ensure a control decision based on current information. But the TAS mechanism can only provide low latency and jitter guarantees if the end device meets predictable transmission times. Therefore, this mechanism is only suitable to a limited extent for such traffic patterns to ensure prioritization of the process data.

In addition to the TAS mechanism, TSN therefore offers further prioritization mechanisms, so-called traffic shapers. These afford the reservation of worst-case bandwidth requirements for time-critical control data based on defined observation intervals (e.g. 250 µs). The forwarding of such reserved traffic is then prioritized by the respective traffic shaper in a way that ensures that certain latency bounds for the time-critical data can be achieved. However, a compromise for the flexibility in prioritization gained in this way is a conceptually lower accuracy in the achievable latency and jitter guarantees in comparison to the TAS mechanism.

As part of the standardization activities at the IEEE, three different traffic shapers are currently being considered for use with TSN:

- Credit-Based Shaper (CBS; IEEE 802.1Qav-2009 [10])
- Cyclic Queuing and Forwarding (CQF; IEEE 802.1Qch-2017 [11])
- Asynchronous Traffic Shaping (ATS; IEEE P802.1Qcr [12])

The Credit-Based Shaper was already developed in 2009 by the IEEE 802.1 working group for the predecessor technology of TSN, Audio/Video Bridging (AVB). As the name suggests, this technology primarily serves audio/video and comparable applications. The aim of the Credit-Based Shaper is to secure the maximum bandwidth required for an audio/video transmission in a defined time interval without noticeably interrupting the best-effort data traffic that shares the same network. To achieve this, the Credit-Based Shaper allocates transmission credit to data streams with reserved bandwidth. Initially this credit is 0.

As long as the transmission credit is in the positive range (≥0), data packets with reserved bandwidth are transmitted preferentially (see, for example, the transmission of the first AVB frame marked in blue in Figure 7 on the left). Through any such preferential transmission, the transmission credit

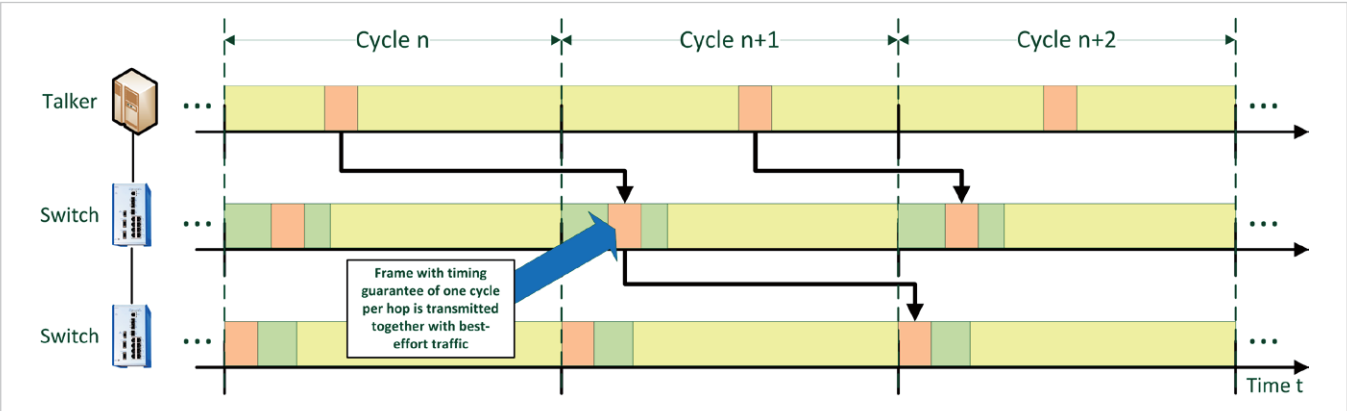

Figure 8: Using Cyclic Queuing and Forwarding, data streams with reserved bandwidth are transmitted intermittently by one hop in the direction of the receiver with each cycle

decreases until it finally reaches a negative value. After finishing the transmission of a packet and while the credit for transmission is negative, data packets with reserved bandwidth are temporarily held back and are no longer being transmitted. During these brief recesses in preferential transmission, best-effort traffic is served. While the AVB packets are waiting, their transmission credit recovers until the credit reaches 0 again. If the forwarding of AVB packets is delayed by longer best-effort packets, the transmission credit of the corresponding data stream increases above the "0" mark (see Transmission of the black marked Best Effort Frame in Figure 7). As a result, after the best-effort packet is finished, the AVB packets that were temporarily held back can be transmitted back-to-back following the best-effort transmission, for as long as the transmission credit lasts. This allows time-critical frames and especially the reserved bandwidth to statistically catch up.

Due to this prioritization behavior, the Credit-Based Shaper is ideally suited for the preferred transmission of audio/video data, for example in video surveillance of production processes. This is of particular importance if limited buffering of data at the receiving end-station is advantageous. However, it has been shown [13] that the maximum end-to-end latency of 2 ms (AVB Class A) or 50 ms (AVB Class B) over 7 hops that is specified in the standard cannot be met in worst case scenarios. This prevents the Credit-based Shaper from being used in application areas such as process control, where fixed guarantees with regard to maximum end-to-end latency are absolutely required. For this reason, the IEEE has been and is currently developing additional traffic shapers that are able to guarantee end-to-end latencies without restriction with regard to network topology and communication patterns.

One, as of 18 May 2017 readily published, traffic shaper is the Cyclic Queuing and Forwarding procedure (IEEE 802.1Qch-2017), which is based on the TAS mechanism. As shown in Figure 8, the core idea of the Cyclic Queuing and Forwarding procedure is to collect the data packets that are received within a cycle with reserved bandwidth and to send them prioritized at the beginning of the next cycle. This allows the maximum end-to-end latency to be clearly determined by the number of hops on the transmission path and the configured cycle time. These properties may make Cyclic Queuing and Forwarding an appropriate mechanism e.g. for process automation applications.

Due to the similarities with TAS, it is clear that Cyclic Queuing and Forwarding also requires a common time understanding of the network participants and thus a time synchronization mechanism must be in place. The third Traffic Shaper, Asynchronous Traffic Shaping, differs in this detail from Cyclic Queuing and Forwarding: It does not require a time synchronization mechanism. Asynchronous Traffic Shaping is suitable for the preferred transmission of telemetry and monitoring data, for example, for manual or non-real-time monitoring. It focuses on the prioritization of traffic bursts. These are passed through the network cyclically and not synchronized as bundles of packets. At the time of writing (Q3 2019), the Asynchronous Traffic Shaping process is still in the process of standardization, such that no final statement can be made about the concrete implementation and performance of this mechanism.

Combined use of Traffic Shapers

The use of the different traffic shapers is always linked to the assignment of one of the eight CoS priorities from the VLAN tag to a specific shaping algorithm. If, for example, a device supports the Time-Aware Shaper according to IEEE 802.1Qbv, the Cyclic Queueing and Forwarding Traffic Shaper according to IEEE 802.1Qch as well as the strict priorities according to IEEE 802.1Q, the various CoS priorities can be assigned to these shaping mechanisms in the device configuration. Priorities 4 and 5, for example, can be assigned to the Cyclic Queueing and Forwarding Shaper and Priority 7 to the TAS to implement communication with soft and hard real-time requirements. In this way, different traffic classes can coexist in the same network and be prioritized by the appropriate mechanism. The basic prerequisite for this, however, is that all devices in the network observe CoS priorities and support all the shaping mechanisms that are utilized.

Preventing interfering traffic with ingress filtering and policing

In a system in which all participants behave as expected, the TSN standards that are described so far already offer all the mechanisms necessary for deterministic data transmission. However, the methods described so far require complete packet reception and (partial) packet processing in a switch or end station. Incorrectly configured devices or malicious network participants can significantly disrupt the functioning of TSN mechanisms such as the TAS by sending data packets with incorrectly assigned CoS priorities or by overusing the resources that the network provides, e.g. bandwidth. To counteract this, the IEEE 802.1 working group has defined Ingress Filtering and Policing mechanisms (IEEE 802.1Qci-2017 [14]) as part of the TSN standardization activities.

These new mechanisms offer the possibility to carry out filtering and policing decisions at different granularity levels. These levels range from a port level to the granularity of individual data streams. In a migration scenario from non-TSN to TSN networks, for example, port-level policing allows the traffic of a non-TSN network subscriber to be restricted to ensure that it cannot use all bandwidth for best-effort traffic. Policing at the data stream level, on the other hand, ensures that a network participant cannot use more than the bandwidth that was reserved by it for that stream. Ingress Policing is not only limited to bandwidth monitoring, but also offers the possibility to allow or disallow packet reception in a device – with the above mentioned granularity levels – based on a certain

time window. In other words: while TAS acts on the output side for time-controlled sending, Ingress Policing offers similar functionality on the input side upon reception, i.e. before the packet is processed by the switch or end device. Due to implementation complexities associated with time-based policing, it is not yet completely clear if and in which scenarios this full version of the Ingress Filtering and Policing mechanism will be used. Bandwidth monitoring, on the other hand, is commonly used today in switches for automation networks. Last but not least, TSN can also be used with existing Layer 2 security mechanisms such as MACsec (IEEE 802.1AE [15]). In this way, the authenticity of the sender can be checked and only correctly verified Ethernet frames can be forwarded. In this way, it is possible to handle a large number of attacks and scenarios with incorrectly configured network participants.

Better safe than sorry: fault tolerance in communication paths

In addition to incorrectly configured network subscribers or those with malicious intent, the failure of a network element or a communication line can also lead to interference in deterministic data transmission. In order to avoid the packet loss associated with such a disruption, a fault-tolerance procedure was developed at the IEEE with IEEE 802.1CB-2017 [16], which works similarly to the already established, seamless redundancy mechanisms High Availability Seamless Redundancy (HSR) and the Parallel Redundancy Protocol (PRP) according to IEC 62439-3, and was kept compatible with them. IEEE 802.1CB is therefore a static fault-tolerance procedure in which the redundant transmission paths are permanently active. In the event of an error, the switchover times from one path to the other path can be avoided.

In order to achieve seamless redundancy according to IEEE 802.1CB, the Ethernet frames are replicated at the beginning of a redundant transmission path and then transmitted through the network via the different redundant paths. Usually, the frame replication takes place either directly at the sending end device or, if the end device does not have a redundant network connection as shown in Figure 9, at the first network element (e.g. a switch) on the transmission path. At the destination, the first replicated data packet is forwarded to the application layer. Subsequently received packet duplicates, on the other hand, are detected via a new redundancy field in the Ethernet header, the so-called R-TAG, and rejected. This ensures that redundant data transmission according to IEEE 802.1CB is transparent for higher-level layers in the network stack and does not have to be considered separately there.

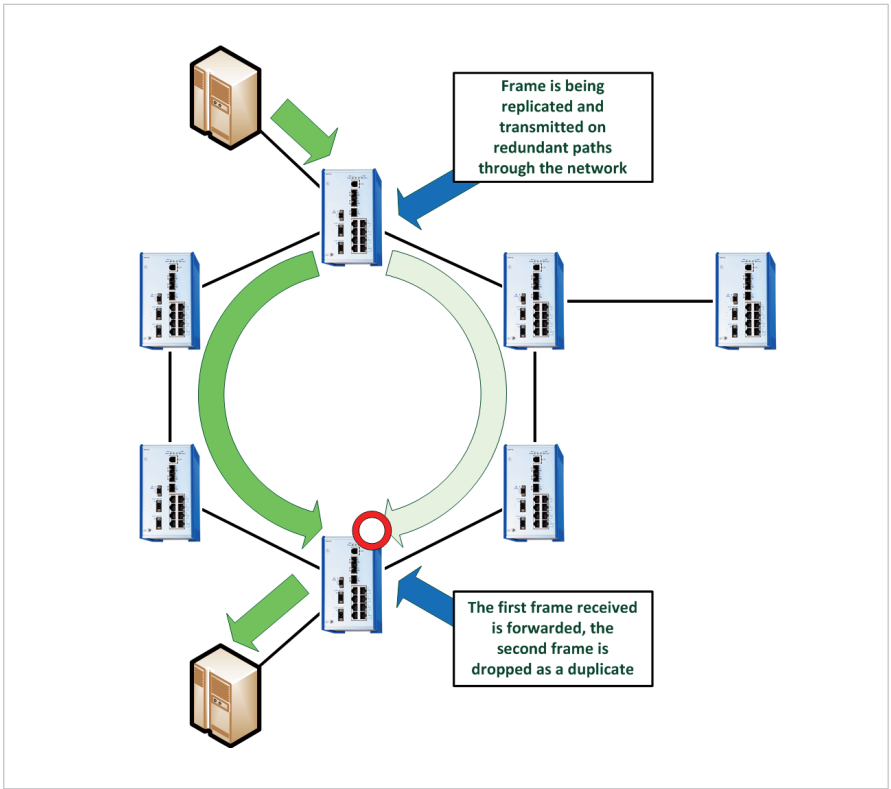

Figure 9: In the case of the seamless redundancy protocol IEEE 802.1CB, Ethernet frames are replicated

The redundancy mechanisms developed as part of IEEE 802.1CB offer a significant advantage over HSR and PRP: they can be used on any topology. IEEE 802.1CB is not limited to the otherwise mandatory ring topologies or topologies with completely independent parallel networks. In addition, IEEE 802.1CB supports redundant transmissions over more than two path in order to further reduce the probability of packet losses and, thus, eliminates this limitation of existing redundancy mechanisms. However, it must be ensured that all redundant paths can meet the required latency guarantees. The management of requirements and comfortable configuration of TSN networks is therefore an important part of a functioning TSN ecosystem that consists of both network devices and network management.

Configuration of the entire TSN network

As explained at the beginning of this white paper, TSN consists of a series of standards and mechanisms that serve the various requirements of deterministic data transmissions. In order to use these mechanisms together in a network and to be able to parameterize them across different network participants from different manufacturers, a standardized configuration method is required. This configuration method must allow the use of TSN mechanisms such as Ethernet frame pre-emption or seamless redundancy according to

IEEE 802.1CB to be activated as required on the network devices. On the other hand, the TSN mechanisms such as TAS must be parameterized consistently for proper functioning, including the configuration of cycle times, CoS priorities and time slots for the time-prioritized transmission of real-time data.

Three different models (IEEE 802.1Qcc-2018 [17]) have been developed at the IEEE for the configuration of TSN: a centralized, a distributed and a combined/hybrid approach. All three approaches share the requirement that the configuration should be largely automated in order to ensure easy handling of TSN. It should be possible for end devices to announce the communication mechanisms that they support, for example TAS, and their data transmission requirements. Network devices subsequently should then be configured automatically according to these announced capabilities and requirements.

The general configuration sequence of a TSN network is as follows: First, the TSN mechanisms supported within a network are determined and activated as required. Subsequently, the transmitting end device, the so-called talker, announces information about the data stream it is about to transmit.

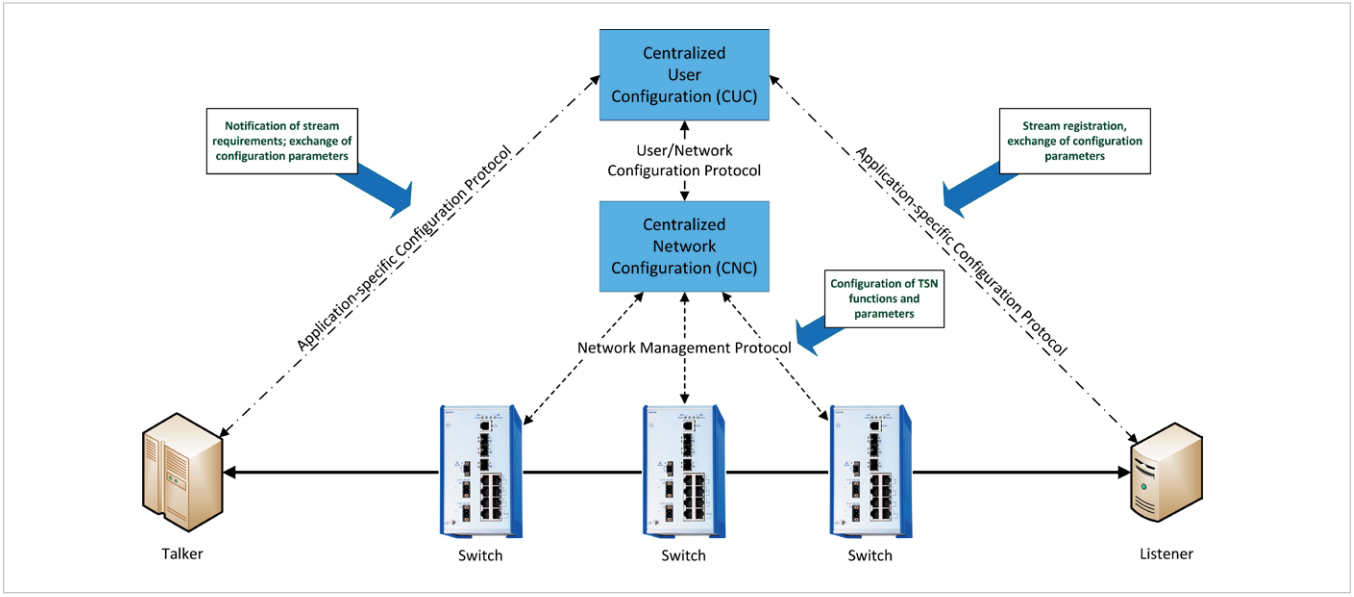

Figure 10: In the centralized TSN configuration approach, the end devices communicate directly with a central configuration instance

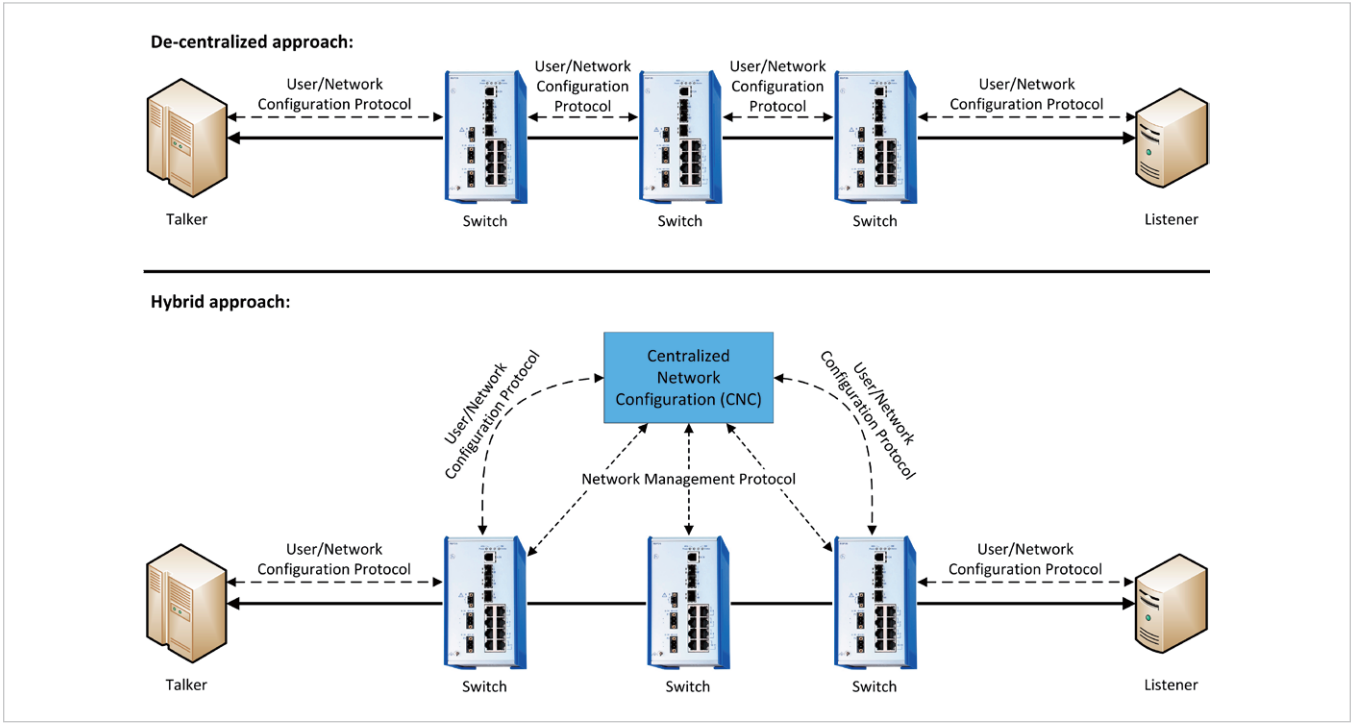

Figure 11: The decentralized and hybrid approaches offer a configuration interface to the end devices that is agnostic to the configuration model

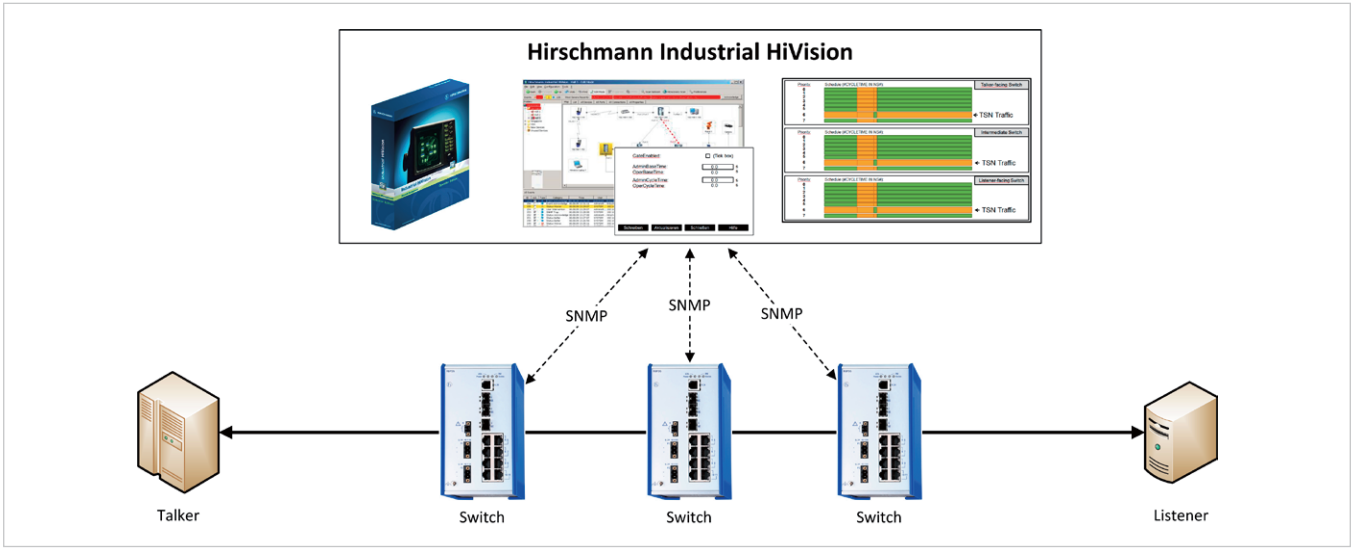

Figure 12: Hirschmann Industrial HiVision enables convenient manual engineering and monitoring of TSN networks

This information includes characteristics such as the cycle time and bandwidth requirements. End devices that are interested in a data stream that is announced, so-called listeners, can then use this information to register for a specific data stream. After registration, they will receive the data packets belonging to the stream.

The three planned configuration models differ in how the device and stream requirements are communicated and processed in the network. With the centralized approach, talkers and listeners communicate via a direct end-to-end connection with a (logically) centralized end device configuration instance, the Centralized User Configuration (CUC), as shown in Figure 10.

The CUC creates the combined requirements for the data streams from the Talker and Listener information and passes them on to the Centralized Network Configuration (CNC), which is also the logical network configuration instance. The CNC calculates the time slot for a new data stream, for example, based on the information available regarding the network topology and the resource reservations already allocated, and configures the network participants (e.g. switches) accordingly. Application-specific protocols such as OPC UA can be used for the information exchange between Talker or Listener and CUC. The switches are configured using existing management protocols such as SNMP (Simple Network Management Protocol) or through YANG models with protocols such as NETCONF. To interface CNC to the CUC, IEEE 802.1Qcc already defines some basic structures in the form of a YANG model. Within the scope of a new standardization project (IEEE P802.1Qdj), these basic structures will now be extended to a complete interface definition. This will, for example, enable a uniform interface based on RESTCONF in the future.

In contrast to the centralized approach, the distributed approach propagates the end device requirements in the network (see Figure 11) and determines a common configuration of the TSN mechanisms to be used, based on the locally available information in each device. The Stream Reservation Protocol (SRP) that was originally developed for the TSN predecessor technology AVB operates based on this principle. However, it does consider the TSN configuration requirements, as TSN was developed later. Therefore, IEEE 802.1 is currently developing the Link-local Registration Protocol (LRP, IEEE P802.1CS [18]) and the Resource Allocation Protocol (RAP, IEEE P802.1Qdd [19]) to specify a new distributed protocol at the IEEE that considers the new TSN mechanisms.

Finally, the hybrid approach combines the centralized and decentralized approaches. Like with the distributed approach, the end devices announce their requirements via the distributed protocol. However, as shown in Figure 11, the actual TSN configuration takes place centrally. One advantage of this approach is that end devices only have to support a single configuration protocol, but the network can be either managed centrally or configured in a decentralized fashion, depending on the network operator's preference. SRP in IEEE 802.1Qcc has already been extended for this purpose.

Even though all three configuration mechanisms described here are still under development, it is already possible today to configure the presented TSN mechanisms. Configuration interfaces are available via standardized protocols such as SNMP. This enables, for example, manual engineering of the cycle times and time slots of the TAS mechanism using network management tools such as Hirschmann Industrial HiVision (see Figure 12).

Summary and outlook

With TSN, deterministic data transmissions with Ethernet according to IEEE 802.1 and 802.3 become possible for the first time. TSN's range of functions allows it to be used in a wide variety of application scenarios, some with very different requirements in terms of transmission latency, jitter and reliability. The "TSN Profile for Industrial Automation" (IEC/IEEE 60802) will therefore be of particular importance for automation networks. Its aim is to pick a selection of mechanisms from the TSN set of standards to form a baseline specification for automation networks. This profile is currently being developed in cooperation between the IEC and the IEEE 802.1 TSN Working Group. In addition, IEEE naturally continues to develop complementary mechanisms and enhances existing TSN standards as required. Central mechanisms of the TSN family have already been completed for some time and have been successfully demonstrated on several occasions. These mechanisms, such as the Time-Aware Shaper, are already integrated in various products, so that their advantages can already be used productively. The IEEE 802 standardization process also ensures complete backward compatibility: TSN networks already installed today can be used in the future as well. The time of TSN has come!

References

1. https://1.ieee802.org/tsn/
2. https://ieeexplore.ieee.org/document/7440741
3. http://ieeexplore.ieee.org/servlet/opac?punumber=7553413
4. http://ieeexplore.ieee.org/servlet/opac?punumber=7900319
5. http://ieeexplore.ieee.org/servlet/opac?punumber=7433913
6. http://ieeexplore.ieee.org/servlet/opac?punumber=4579757
7. http://ieeexplore.ieee.org/servlet/opac?punumber=5741896
8. Hirschmann IEEE 1588 White Paper
9. https://1.ieee802.org/tsn/802-1as-rev/
10. http://ieeexplore.ieee.org/servlet/opac?punumber=5375702
11. http://ieeexplore.ieee.org/servlet/opac?punumber=7961301
12. https://1.ieee802.org/tsn/802-1qcr/
13. http://www.ieee802.org/1/files/public/docs2010/ba-boiger-bridge-latency-calculations.pdf
14. http://ieeexplore.ieee.org/servlet/opac?punumber=8064219
15. http://ieeexplore.ieee.org/servlet/opac?punumber=11085
16. https://ieeexplore.ieee.org/document/8091139
17. https://ieeexplore.ieee.org/document/8514112
18. https://1.ieee802.org/tsn/802-1cs/
19. https://1.ieee802.org/tsn/802-1qdd/

About Belden

Belden Inc., a global leader in high quality, end-to-end signal transmission solutions, delivers a comprehensive product portfolio designed to meet the mission-critical network infrastructure needs of industrial, enterprise and broadcast markets. With innovative solutions targeted at reliable and secure transmission of rapidly growing amounts of data, audio and video needed for today's applications, Belden is at the center of the global transformation to a connected world. Founded in 1902, the company is headquartered in St. Louis, USA, and has manufacturing capabilities in North and South America, Europe and Asia.

For more information, visit us at www.belden.com and follow us on Twitter @BeldenIND.