AUTOWARE

Wireless Autonomous, Reliable and Resilient Production Operation Architecture for Cognitive Manufacturing

D2.2a AUTOWARE Deterministic Ethernet Communications

Document Owner	TTTech Computertechnik AG		
Contributors	-		
Reviewers	SmartFactoryKL		
Dissemination level	Public	Dissemination nature	Report
Date	18/07/2018	Version	1.0

European Commission

Version History

Nr.	Date	Author (Organization)	Description
0.1	22/05/2018	TTT	Deliverable structure and ToC
0.2	08/06/2018	TTT	Overview Ethernet Standards
0.3	22/06/2018	TTT	Description reconfiguration concept and network tool
0.4	25/06/2018	TTT	Version ready for review
0.5	17/07/2018	SFKL	Review
1.0	18/07/2018	TTT	Updated based on review and final version

Project partners

Software Quality Systems	SQS
Asociación de Empresas Tecnológicas Innovalia	INNO
Technologie Initiative SmartFactoryKL e.V.	SmartFactory
Josef Stefan Institute	JSI
TTTech Computertechnik AG	TTT
Consiglio Nazionale Delle Ricerche	CNR
imec	imec
Robovision	Robovision
Universidad Miguel Hernández	UMH
Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V.	FhG
Blue Ocean Robotics	BOR
Fundación Tekniker	Tekniker
SMC Pneumatik GmbH	SMC

Table of Contents

List of Figures

List of Tables

No table of figures entries found.

Executive Summary

This deliverable D2.2a provides an overview of the work that is being performed within the AUTOWARE project in the field of Deterministic Ethernet Communication and dynamic reconfiguration under consideration of deterministic requirements and changing environments with real-time constraints. This deliverable is related to the work performed in Task 2.2 – Deterministic Ethernet Communication and CPS control/networking, which is led by partner TTTech.

Deterministic Ethernet refers to a wired networked communication technology that uses time scheduling to bring deterministic real-time communication on one Ethernet network. Time-scheduled traffic is partitioned from all other network traffic and is therefore immune from disturbance. The document will give a short introduction in Ethernet-based communication and present the Time-Sensitive Networking (TSN) concept which will form the basis for the work in this document. Additionally, first steps towards the reconfiguration concept of communication networks will be presented together with a configuration tool for wired Ethernet networks.

The work in this document is not the final work with respect to Deterministic Communication inside the AUTOWARE project, but will be further developed in the course of the project. In the coming months, further development will take place in implementing TSN and upholding the TSN standards that are still under development. Additionally, the reconfiguration concept introduced in this chapter will be further developed together with developments of a network configuration tool for TSN communication networks.

Keywords

Deterministic communication, Deterministic Ethernet, IEEE 802.1, Time-Sensitive Networking (TSN), Reconfiguration

Acronyms

AFDX	Avionics Full Duplex Ethernet
AVB	Audio/Video Bridging
CA	Consortium Agreement
CBS	Credit Base Shaper
CNC	Centralized Network Configuration
CPPS	Cyber-Physical Production System
CUC	Centralized User Configuration
HSR	High-availability Seamless Redundancy
IETF	Internet Engineering Task Force
IIoT	Industrial Internet of Things
NETCONF	Network Configuration Protocol
OEM	Original Equipment Manufacturer
PRP	Parallel Redundancy Protocol
SRP	Stream Reservation Protocol
TAS	Time-Aware Shaper
TSN	Time-Sensitive Networking
UNI	User Network Interface
VLAN	Virtual Local Area Network
YANG	Yet Another Next Generation

1 Introduction

Communication is the key element of the emerging paradigm of the Industrial Internet of Things (IIoT), and especially the Real-Time IIoT, as more and more systems will be interconnected [Atzori, 2010]. Another trend that is currently happening is that products must have shorter product production life cycles, constantly changing customer requirements and with respect to that, manufacturing systems need to be extremely flexible [Lafou, 2016], especially on the interaction/communication between all the connected devices inside a manufacturing system. Communication of real-time networks is one of the most challenging demand of the Real-Time IIoT, where the network has to be deterministic and yet flexible enough to adapt to changes through its life cycle.

The scope of Work Package 2 is to provide heterogeneous communications and networking architecture to support connectivity and data management in Cyber-Physical Production Systems (CPPS), and to develop novel Deterministic Ethernet technology to efficiently and reliable support flexible reconfiguration of future industrial systems. The latter is the main focus of the deliverable, where Time-Sensitive Networking (TSN) will be the main Deterministic Ethernet technology being considered. The work presented in this deliverable is mainly related to the work being performed in Task 2.2 – Deterministic Ethernet Communications and CPS control/networking.

The upcoming sections of the deliverable are organized as follows. Section 2 gives as introduction an overview of existing Ethernet Standards and how the different standards have been developed and where they are used. Section 3 provides a description of two Deterministic Ethernet technologies, namely SAE6802 Time-Triggered Ethernet (TTEthernet) and IEEE802.1 Time-Sensitive Networking (TSN), which are the communication concepts that are of main focus to the research of the AUTOWARE partners. For the work being performed in the AUTOWARE project, the TSN standard will be used and therefore, the focus will mainly be on this standard. Following, section 4 will introduce the concept of dynamic reconfiguration and provide first concepts and approaches that are being developed to investigate the concepts of dynamic reconfiguration under fill consideration of deterministic requirements and changing environments with real-time constraints. Additionally, a network (re-)configuration tool will be introduced that can support the configuration of deterministic communication network. The deliverable is finally concluded with section 6, which provides a small conclusion and a look out to the upcoming work and the final deliverable of this task.

2 Ethernet Networking

2.1 Ethernet Standards

Figure 1 shows different Ethernet networking standards with their target applications (industry niches) and use cases (application criteria). Specific network capabilities are added by modifications or enhancements at layers 3-7. The capabilities which support the integration of different functions and traffic isolation for critical applications, are defined at Layer 2. At this level, it is possible to support detailed HW-based control of data flows on every port. The following figure shows an overview of Ethernet standards and protocols which are used in different applications.

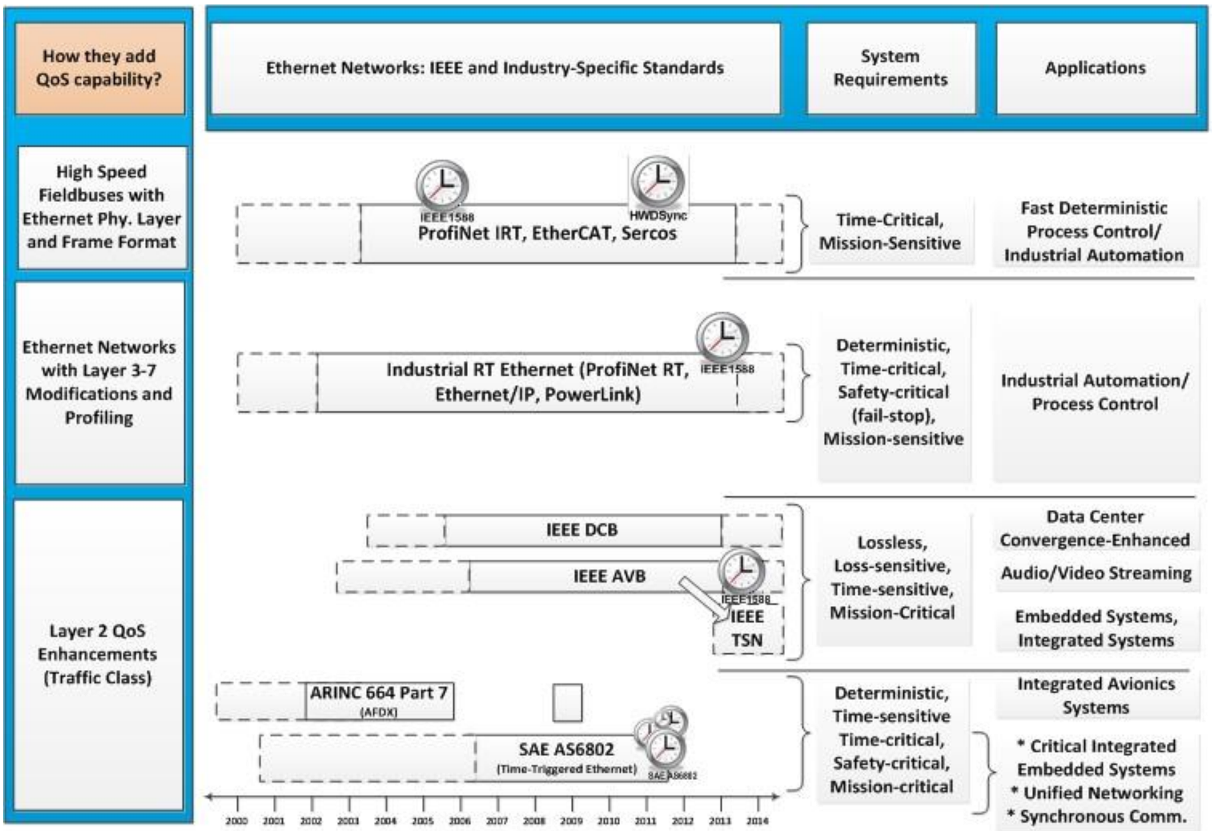

Figure 1: Ethernet variants and protocols addressing different system requirements and application domains

Obviously, it is possible to design real-time systems using the above presented technologies, but very few of their capabilities can be used for design of scalable Ethernet-based integrated architecture which can host mixed-criticality functions in safety-critical systems (up-to level SIL4).

2.1.1 ARINC664

The ARINC664 standards and its core technology was developed between 1998-2004 by Airbus, Boeing and their key avionics suppliers (e.g. Rockwell Collins). ARINC664 [ARINC, 2009] proposes "*profiled networks*", *which are entitled to adapt the IEEE 802.3, 802.1D and "IP" (RFC 1122) standards in order to fulfil specific performance or safety needs. For instance, a subset of profiled networks, called "deterministic networks", is defined for those aircraft network domains where quality of service (including timely delivery) is the objective.*

The idea was to avoid traffic congestion and prevent the overflow of internal switch queues and memory buffer, and minimize frame drops or loss. The frame drop avoidance was not the only driver, as the Avionics Full Duplex Ethernet (AFDX) network does not guarantee a fully lossless service, but it minimizes the packet loss probability. The deterministic networking approach applied in AFDX requires a computing model which is insensitive to the occasional loss of frames. The primary driver was to keep maximum latency under control to provide deterministic latency.

Avionics Full Duplex Switched Ethernet (AFDX) defines the protocol specifications (IEEE 802.3 and ARINC664, Part 7) for the exchange of data between Avionics Subsystems. Airbus has patented the key AFDX mechanisms, which can be licensed by semiconductor component providers.

Same as standard Ethernet, AFDX networks contain the following components:

- AFDX End Stations: Network interface card with AFDX interface to the network, and a host CPU interface to the computer node.
- AFDX Switches: network devices which forward Ethernet frames to their target destinations.

2.1.2 IEC 62439-3-4/5 PRP/HSR

IEC 62439-3 [IEC, 2016] specified two redundancy protocols based on the duplication of the LAN, designed to provide seamless recovery in case of single failure of an inter-switch link or switch in the network. Critical applications may require much faster recovery on link or path faults and recovery periods of 100ms or seconds may be too long for fast processes.

This set of standards is relevant for redundancy management in general switched Ethernet networks (PRP – Parallel Redundancy Protocol) and their linear/circular topology variants (HSR – High-Availability Seamless Redundancy). In the PRP, the objective is to send redundant frames over two independent paths (see Figure 2), and allowing the receiving end station to decide how to handle incoming redundant messages. The sequence number field is attached into the message. Same as in ARINC664, PRP will accept the first available message and discard the second one.

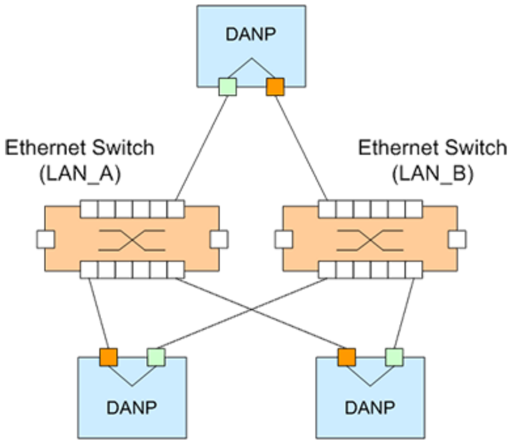

Figure 2: PRP Network with DANP (doubly attached nodes)[PRP, 2018]

HSR uses an equivalent mechanism, but in a daisy chained ring (see Figure 3), by sending frames in opposite directions. HSR adds a new forwards mechanism to the switch, which behaves as a simple end station.

As Layer 2 mechanisms, PRP/HSR can “cooperate” with other mechanisms used in proprietary Ethernet solutions. However, it does not represent a holistic solution for the design of scalable integrated systems with hard RT performance.

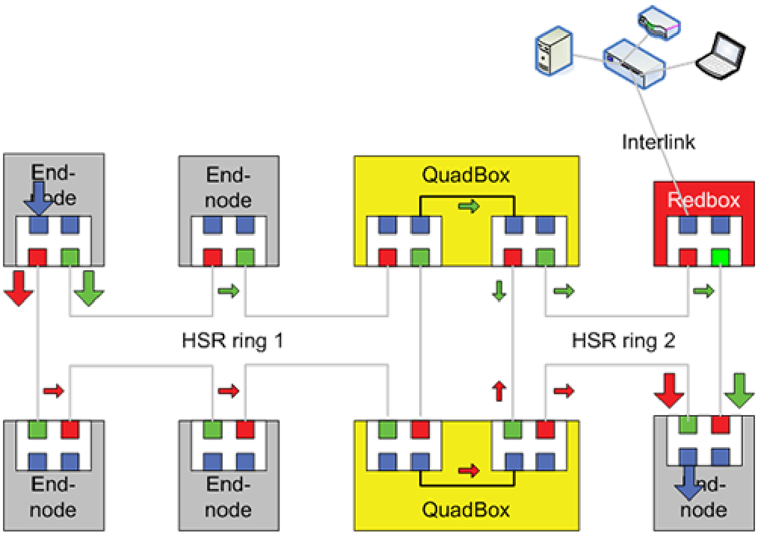

Figure 3: Two HSR rings connected [HSR, 2018]

As a conclusion, ARINC664 and PRP provide similar capabilities for redundancy management. The only difference is the position of the sequence counter in the frame. Both PRP and HSR mechanisms do not have any impact on isolation or temporal boundaries in the system, and represent only one of mechanisms for designing highly available networks.

2.1.3 IEEE AVB

IEEE AVB is a set of protocol services which solve the challenge of periodic communication for audio/video applications (AVB = Audio/Video Bridging). It supports smooth traffic shaping on every switch, and allows up to 25% of best effort traffic, with bandwidth reservation. IEEE AVB includes the synchronization of applications via IEEE 802.1AS. This standard also supports stream reservation protocols suitable for data producer and subscribers within the network.

In simple linear architectures with relatively few data flows, an AVB network can perform better than standard Virtual Local Area Network (VLAN) network architectures. AVB as a set of network protocols relies on traffic shaping inside switches to provide generic temporal boundaries on latency, prevent stream distortion and microbursts. AVB can support 2ms latency with few controlled data flows and NX10 A/V channels per port over 7 hops.

The analysis on AVB latency is based on a simplified analytical model of AVB Ethernet switch without technology latency and occasional jitter in end-station transmissions, but it reveals key mechanisms and considerations used in the design of IEEE AVB.

This analysis claims that the end-to-end latency/delays can be varied effectively by changing link utilization level and shaping period, but AVB focuses solely on two classes of traffic (high-priority A and low-priority B) with different periods of 125 and 250µs, which limits the maximum number of channels and data flows.

The benefit of end-station synchronization helps to avoid congestions and reduce the switch buffer memory resources requirements. By scheduling packet transmissions much better control of network bandwidth use can be accomplished. Similar to other asynchronous packet-switched network, the number of data flows supported will depend on the link bandwidth, number of data flows, and the number of channels per stream, and will be limited by the configuration tool calculation capabilities.

2.1.3.1 IEEE AVB/TSN

New emerging deterministic Ethernet capabilities enable time-multiplexed bandwidth sharing and a number of services defined for embedded system capabilities. The IEEE TSN WG will define a number of amendments to IEEE 802.1Q and 802.3, which will be later adopted as a part of 802.1Q, together with VLANs. Most probably this set of standards will be completed before 2020, with essential part capabilities implemented before this date.

The key players behind the IEEE TSN standardization include IT/networking, telecom, automotive, and industrial OEMs (Original Equipment Manufacturers), as well as all major semiconductor and Ethernet network switching companies. Therefore, it can be safely assumed that such industry support for this standard will lead to a broad availability of components from different suppliers and affordable pricing for network solutions. More detailed information about TSN is presented in Section 3.2.

3 Deterministic Ethernet Networking

In a Deterministic Ethernet network, regular constrained best-effort Ethernet traffic can co-exist with real-time critical traffic flows without altering the guaranteed and deterministic strict delivery timeliness of scheduled traffic flows (see Figure 4). Mixed-criticality requirements from classic industrial deployments can be fulfilled and guaranteed with the support of deterministic scheduled networks and a carefully build distributed communication scheme.

Figure 4: Deterministic Ethernet

To make it possible for the customers to converge real-time traffic with regular best effort traffic on one Ethernet network, in deterministic networks, all the traffic in the network needs to be scheduled. This allows isolating the critical real-time traffic from the remaining traffic. It is therefore immune to disturbances, i.e., critical functions can send messages at scheduled points in time with a guarantee of available bandwidth and message delivery. This isolation of time-critical traffic allows the convergence of many different types of traffic, such as control, data-analytics, operations and enterprise functions on a single physical network. This means that in a Deterministic Ethernet network, latency of critical scheduled communication can be guaranteed. This is called Guarantee of Service.

Deterministic Ethernet is used in a wide range of applications where guaranteed latency is vital, either for reasons of operational efficiency of functional safety. These include autonomous driving, machine-to-machine communication, and aerospace flight control. The configuration of the Deterministic Ethernet is then seen as a one-time event that occurs during the initialization of the network and any change during run-time required often manual reconfiguration. Whereas this characteristic has not been such a problem in the past due to the static nature of Operation Technology networks like automotive or industrial automation, the online reconfiguration of the network is now becoming an urgent need as we are seeing larger and larger networks, transporting heterogeneous traffic that still require real-time guarantees.

To overcome this issue, the vendor-independent communication protocol such as OPC UA protocol is needed to be implemented for the Deterministic Ethernet. One of the main exponents of this integration is the OPC UA over TSN also called OPC UA TSN. The OPC UA over TSN development consists out of two main parts; Time-Sensitive Networking (TSN) and OPC UA. The main functionality of OPC UA over TSN is providing seamlessly shared information between all kinds of machines, devices and sensors in real-time.

3.1 SAE AS6802 – Time-Triggered Ethernet

As described in the SAE AS6802 standard, Time-Triggered Ethernet (TTEthernet) combines the proven fault-tolerance and real-time mechanisms of the time-triggered technology that enables deterministic communication, with mechanisms that allow flexible and free-form communication compatible with legacy Ethernet, and therefore it is suited for different types of applications. Time-Triggered Ethernet combines together the high flexibility of free-form systems and the reliability and determinism of statically configured systems.

In order to support integration of applications with different real-time and safety requirements in a single network, Time-Triggered Ethernet supports three different traffic classes:

- Time-Triggered (TT) traffic: is sent in a time-triggered way, i.e. each Time-Triggered Ethernet sender node has a transmit schedule, and each TTE-Switch has a receive and forward schedule. This traffic is sent over the network with constant communication latency and a small and bounded jitter.
- Rate-constrained (RC) traffic: is sent with a bounded latency and jitter ensuring lossless communication. Each Time-Triggered Ethernet sender node gets a reserved bandwidth for transmitting messages with the RC traffic. No clock synchronization is required for the RC message exchange.
- Best-effort (BE) traffic – traffic with no timing guarantees. The BE traffic class is compatible with the IEEE 802.3 standard Ethernet traffic.

3.2 Time-Sensitive Networking (TSN)

Time-Sensitive Networking (TSN) is the set of IEEE 802 Ethernet sub-standards (see Figure 5) that are defined by the IEEE TSN task group. The new standards describe several mechanisms for improved or even guaranteed real-time delivery of Ethernet traffic. Most prominently, TSN defines the first IEEE standard for time-triggered message forwarding in a switched Ethernet network, and therefore fully deterministic real-time communication within the 802 suite of standards.

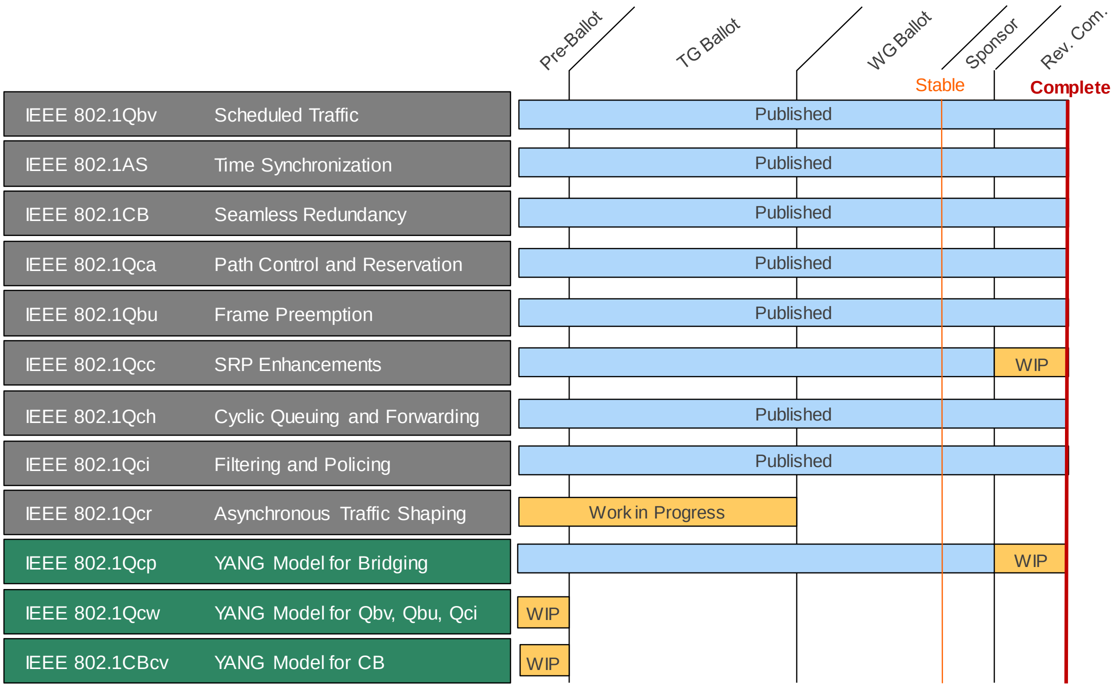

Figure 5: TSN Standardization Status (June 2018)

Currently, the following TSN standards are supported in available components:

- **IEEE 802.1Qbv – Scheduled Traffic:** the core of TSN is a time-triggered communication principle, known in TSN as the "Time-Aware Shaper" (TAS), which deterministically schedules traffic in queues through switched networks (see Figure 6). This is standardized in IEEE 802.1Qbv. The TAS enables to control the flow of queued traffic from a TSN-enabled switch. Ethernet frames are identified and assigned to queues based on the priority of the frames. Each queue operation is based on a schedule, and the transmission of messages in these queues is then executed at the output ports during the scheduled time windows. Other queues and ports are then blocked, thereby removing the chance of scheduled traffic being interrupted by non-scheduled traffic.

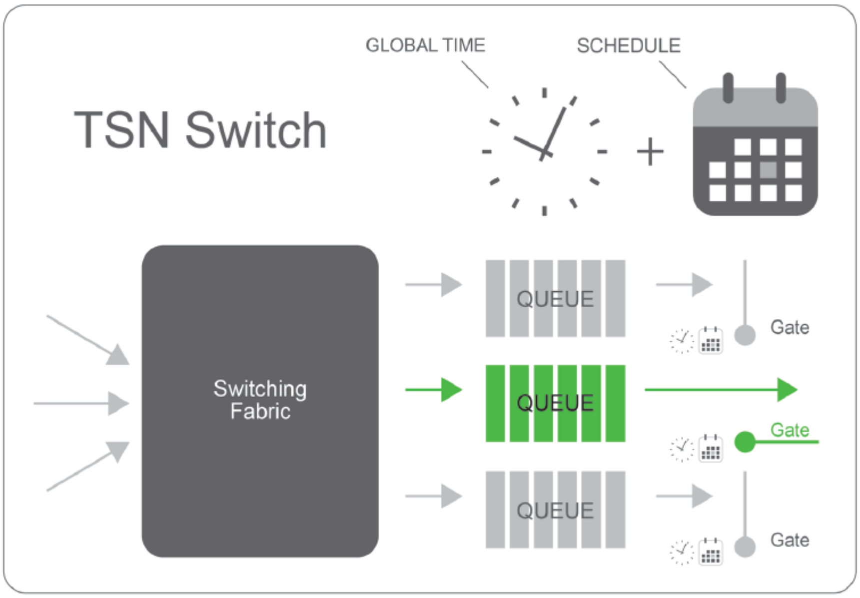

Figure 6: TSN queues and transmission gates

- **IEEE 802.1Qbu – Frame Pre-emption:** This standard describes the concept that the TAS (IEEE 802.1Qbv) avoids transmission jitter by blocking lower priority queues in advance of the transmission point of the critical frame. In cases, where minimal latency for scheduled messages is desired, the TAS mechanism may not be an optimal solution. Therefore, on links where pre-emption as defined by this standard is supported, the transmission of standard Ethernet frames can be interrupted in order to allow high-priority frames to be transmitted, and afterwards resume the transmission of the interrupted frame (without discarding the message).
- **IEEE 802.AS – Timing and Synchronization:** Clock synchronization is a vital mechanism for establishing deterministic communication with bounded message latency in TSN. The IEEE 802.1AS standard creates a profile of the IEEE 1588 PTP synchronization protocol for TSN. This profile will enable clock synchronization compatibility between different TSN devices. Additionally, IEEE 802.1AS standardizes the use of multiple grandmaster clocks as well as the possibility to make connections to these grandmaster clocks. Replication of grandmaster clocks results in shorter fail-over times in cases when a grandmaster clock becomes faulty. Furthermore, IEEE 802.1AS support multiple synchronized clocks, enabling timestamping of events such as production data or measurements, and the synchronization of applications such as sensors, actuators and control units.
- **IEEE 802.CB – Frame Replication and Elimination for Reliability:** The IEEE 802.1CB standard implements a redundancy management mechanism similar to the approaches known from HSR (High-availability Seamless Redundancy – IEC

62439-3 Clause 5) and PRP (Parallel Redundancy Protocol – IEC 62439-3 Clause 4). In order to increase availability, redundant copies of the same messages are communicated in parallel over disjoint paths through the network. This feature has similarities with AFDX integrity checking.

- **IEEE 802.1Qci – Per-Stream Filtering and Policing:** Per stream filtering and policing prevents adverse effects on system communication performance, as a result of faulty end-stations which would otherwise violate the engineered bandwidth use. This fine-grained policing capability allows to better control different data flows in complex systems.
- **IEEE 802.1Qcc – Stream Reservation Protocol (SRP) Enhancements and Performance Improvements:** TSN also provides mechanisms to improve existing reservation protocols such as SRP (Stream Reservation Protocol – IEEE 802.1Qat) in order to meet the configuration requirements of industrial and automotive systems, such as timing, bandwidth reservation, frame pre-emption, synchronization, and redundancy. This standard will enable consistent configuration of Ethernet switches form various vendors. In addition, it will support the implementation of central configuration models for dynamic scheduling of TSN networks.
- **IEEE 802.1Qca – Path Control and Reservation:** This protocol relies of IS-IS and collects topology information form nodes (network discovery), to be able to adapt dynamically on network modifications and failures, and contains the mechanism to specify the path, bandwidth reservation and redundancy for data flows. However, the IS-IS algorithm is complex and in-depth formal analysis for arbitrary topologies have not been carried out partly due to this complexity.

4 Dynamic Reconfiguration

Communication is one of the main elements of the upcoming (Real-Time) Industrial Internet of Things (IIoT). In the future, more and more systems will be interconnected to each other. A trend in industrial automation is towards shorter product development cycles and flexible manufacturing systems, where continuously new or different systems are being added or removed from the manufacturing systems, based on the customer requirements. Networks for those kind of systems need to be prepared to satisfy these new requirements while still being able to provide deterministic guarantees. The network must be capable to adapt to all kind of changes, like e.g. changes in the topology of the network, adding or removing equipment, changes for maintenance purposes, or even changes in the functionality of the network. Within the AUTOWARE project, TTT is investigating the possibility to dynamically reconfigure the network, thereby upholding the deterministic requirements (i.e. timing, latency, etc.) of the communication network. The work that is being presented here, is work that has started inside the AUTOWARE project and will be an investigation of concepts and potentially first tests. A complete working approach is beyond the scope of the AUTOWARE project.

The standard that deals with the (re-)configuration of TSN networks is IEEE 802.1Qcc (see Figure 5). The development of the standard is an ongoing project – at the moment of the execution of the AUTOWARE project, the status of the standard is still a draft format, but the aspects discussed here are considered stable. As mentioned before, IEEE 802.1Qcc is an enhancement of the Stream Reservation Protocol (SRP) (IEEE 802.1Qat) designed for the resource management in networks using the Credit Base Shaper (CBS) (IEEE 802.1Qav). CBS defines two traffic classes A and B and traffic of each class is allowed to be sent as long as there is enough credit for that traffic type. Credits are being accumulated to queues as they wait to transmit. Credits are spent by queues when they are transmitting. Queues with positive credits are eligible for transmitting. In that context SRP was a simple admission protocol in which the talker announces the traffic that it will send and depending on the available bandwidth it will be granted permission to do it, or not. Unfortunately, with the inclusion of more complex traffic shaping mechanism in TSN, such as the Time Aware Shaper (TAS) (scheduled traffic) or frame pre-emption, an update for the SRP was needed.

One of the main elements for the configuration of TSN networks is the User Network Interface (UNI). On the user side of this interface are the talkers and the listeners. On the network side are the bridges. The idea is that the user specifies the requirement for the streams that they want to transmit without having all the details (e.g. end systems, latency, switches, etc.) about the network. The network then analyses this information along with network capabilities and configures the bridges to meet the user requirements. To realize this configuration paradigm, IEEE 802.1Qcc defines three different configuration models with regards to their architecture:

- **Fully Distributed Model**: In this model, the User Network Interface (UNI) is situated between the talker/listener (user) and the bridge to which it is connected (network). The user transmits its requirements and the network propagates them through the relevant paths. The management of the bridges is performed locally just with the information that is available to that bridge. This model can be used to configure the CBS and for that the SRP can be used as a UNI. One limitation of this model is the lack of a centralized view with complete knowledge of the network that makes it not suitable for the configuration of the TA.

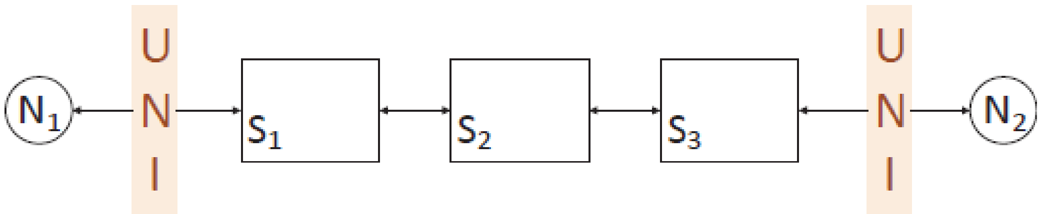

Figure 7: IEEE 802.1Qcc Fully Distributed Model

- **Centralized Network/Distributed User Model:** This model comes to alleviate that limitation of the fully distributed model by introduced the Centralized Network Configuration (CNC). The UNI is still between the talkers/listeners and the bridges, but in this model the bridge communicates the user requirements directly to the CNC. The CNC has a complete knowledge of the network topology as well as the bridges capabilities and that enables it to perform more complex calculations needed to configure the TAS, frame replication and elimination or frame pre-emption. The management of the bridges is performed by the CNC using a network management protocol. The management of end-stations is not performed by the CNC. The CNC can either exist in an end-station or a bridge.

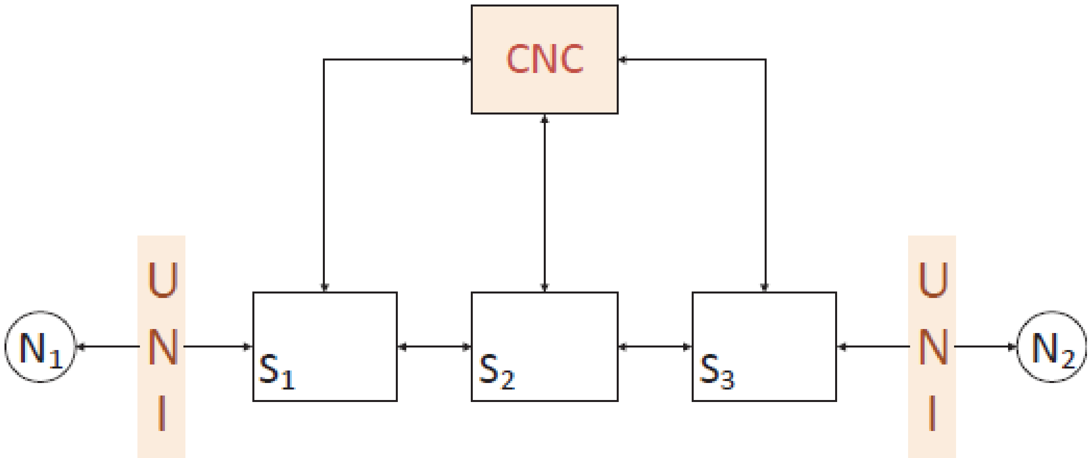

Figure 8: IEEE 802.1Qcc Centralized Network/Distributed User Model

- **Fully Centralized Model:** In the two previous models, the configuration of the end-stations was not addressed. However, there are use-cases for highly critical applications, such as automotive or industrial control, in which there are complex timing requirements that needs extra configuration. For those cases, the notion of the Centralized User Configuration (CUC) is introduced in this model. The talkers/listeners communicate their requirements to the CUC and then the CUC exchanges this information with the CNC through the UNI. The CNC and the CUC can be implemented in the same device or in separate devices. The definition of the communication protocol between the CUC and the end-stations is considered to be out of scope of the IEEE 802.1Qcc.

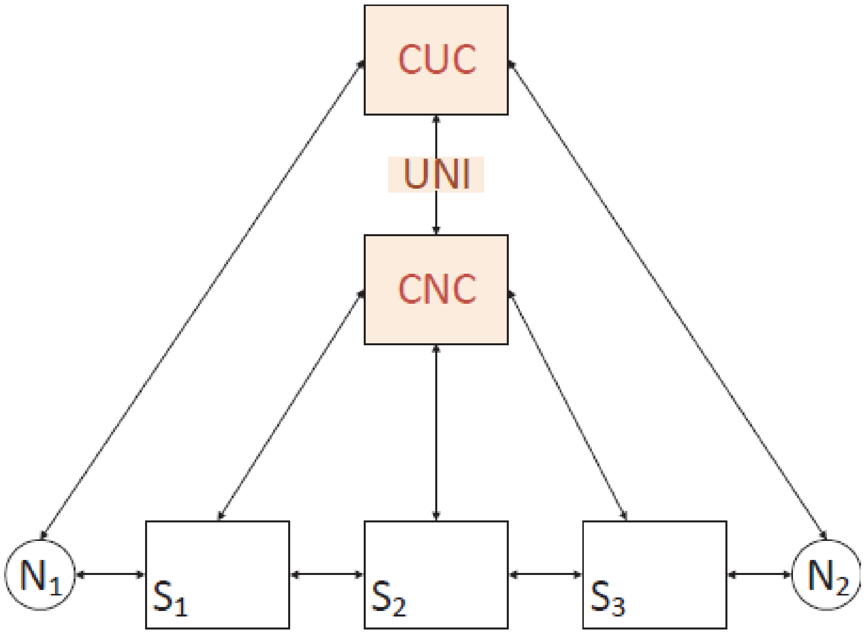

Figure 9: IEEE 802.1Qcc Fully Centralized Model

These three models represent three different approaches how configuration can be handled in a TSN network. To manage the configuration aspects of each functionality defined in the standards, other protocols are needed to manage the configuration aspects. For the purpose, NETCONF and OPC UA are introduced. The detailed functionality of each of them are described in the following sections.

4.1 NETCONF

Despite the protocol independent design of IEEE 802.1Qcc, there is an ongoing standardization effort to model managed using YANG (Yet Another Next Generation). YANG is a data modelling language standardized by the Internet Engineering Task Force (IETF) [Bjorkland, 2010] to be used with the Network Configuration (NETCONF) protocol. NETCONF has been developed and standardized by IETF [NETCONF, 2011] with the aim of creating a configuration management protocol that alleviates some of the most common problems with existing network configuration protocols. Most of these protocols are vendor-specific and operate as command line interfaces which translates into low efficiency and low reliability. The main goal has been to move from the "network is the record" paradigm in which operators directly modify the configuration of devices to the "generate everything" approach in which a global knowledge of the network is used to generate new configurations that are pushed to the devices.

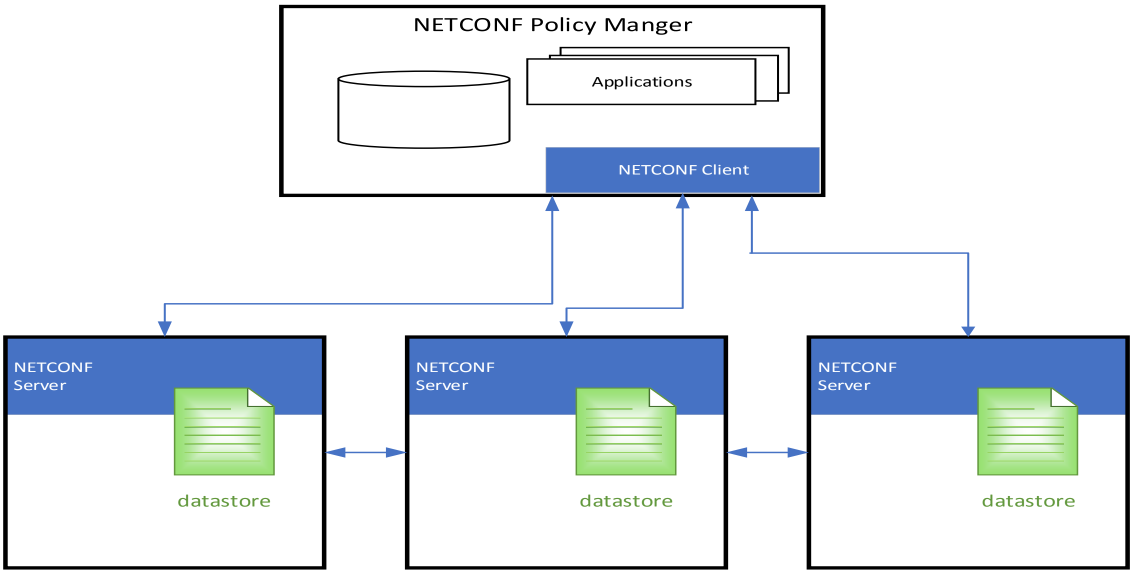

Figure 10: NETCONF deployment

A NETCONF deployment is depicted in Figure 10. The centralized architecture is needed to have a global knowledge of the network and the Policy Manager acts as the network manager. With regards to the NETCONF protocol we can see two roles: servers and clients. The network devices act as the configuration servers and the Policy Manger as the configuration client. NETCONF represents the configuration of a network device as a structured document that is called the "datastore". Datastores are stored in each network device and can be retrieved and edited. NETCONF supports different types of datastores: *running* datastore that contains the current configuration of the device. It should always be present. The other two datastores are optional: *start up* datastore with the initial configuration of the device, and *candidate* datastore that is a draft configuration for the device that can be edited for a later commit.

To interact with devices configuration NETCONF provides a series of operations such as *edit-config, get-config, copy-config* and/or *delete-config*. For large configuration datastores NETCONF also provides filtering mechanisms that allow the client to modify or retrieve just a subset of the whole configuration. NETCONF sees configuration changes as network-wide transactions, this means that if there is an error in the configuration, for example some inconsistency, the transaction will fail. The existence of those consistency and the all-or-nothing semantics guarantee a safe-way of updating the configuration.

4.2 OPC UA

OPC UA is a cross-platform service-oriented architecture and data exchange technology that enables safe and reliable vendor- and platform communication and is widely used in the industry.

OPC UA has been supporting the Client/Server approach for communication. In this situation, a client makes a request and receives an answer from a server (response). Unfortunately, this system will reach quickly its boundaries when the network becomes too large, meaning having too many participants that are sending data over the network. For each communication, a client-server link has to be established. In the case of Industry 4.0 and IIoT, where everything is connected with everything, this will lead to an explosion of the amount of links and the Client/Server approach will not suffice. The Publish/Subscribe (Pub/Sub) model, on the other hand, enables one-to-many and many-to-many communication. A server sends the data in the network (Publish) and each Client can receive this data (Subscribe). The difference is depicted in Figure 11.

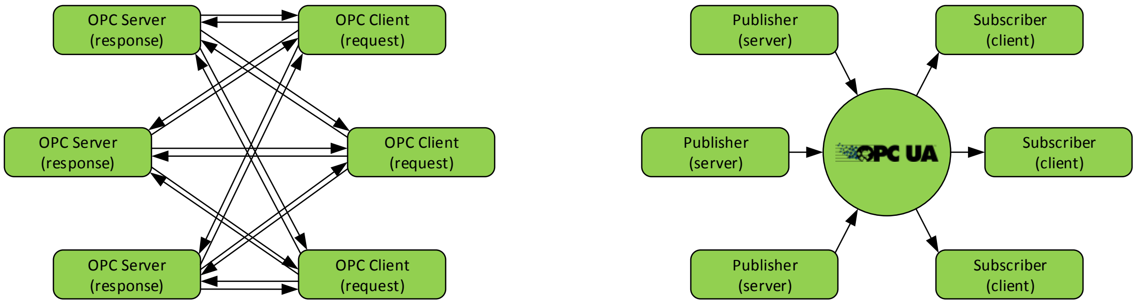

Figure 11: Client/Server vs Publish/Subscribe

4.2.1 Merging OPC UA and TSN

The merging of OPC UA Pub/Sub and with TSN is more than just adding the two parts together. Currently, the available OPC UA Pub/Sub stacks are not TSN aware and therefore cannot make use of the substantial advantages of TSN (e.g. synchronization, convergence, robust delivery, etc.). To solve this, modifications of the applied OPC UA stacks are mandatory, which is one of the biggest challenges identified within the TSN developments with respect to OPC UA over TSN.

Additionally, TSN is constantly under development. As mentioned before, TSN is an 802 suite of standards. These standards are unfortunately still under development and have not been finalized yet. The further development of the TSN solutions is one of the priorities of TTTech within the project and the results of these developments will directly flow into their product portfolio.

4.3 Reconfiguration Approach

The solution that is being proposed to establish dynamic reconfiguration under full consideration of deterministic requirements and changing environments with real-time constraints is to introduce intelligence into the network. This is done by creating an autonomous system that learns the characteristics of the network through continuous monitoring and analysis of the network. The system will be able to update the configuration of the network preserving the original real-time requirements (e.g. maximum transmission latency and jitter) and optionally improve them (e.g. less latency and jitter).

The system will be made up of the following functional elements:

- **Monitor:** Observes the network's traffic and gathers measurements to identify traffic patterns. The goal is to recreate from the identified parameters the original real-time constraints defined by the currently running applications.
- **Extractor:** Derives traffic parameters based on the traffic patterns observed by the Monitor and previous knowledge of the network and applications. This part is the learning phase of the reconfiguration.
- **Scheduler:** Uses the traffic parameters obtained by the Extractor to generate a schedule for the network, that maintains and improves the deterministic guarantees.
- **Reconfigurator:** In charge of updating the network configuration to follow the new communication schedule generated by the Scheduler.

Figure 12 gives an overview of the reconfiguration system, where the Monitor and Reconfigurator elements directly interact with the system, for either gathering information from the network, or updating the configuration with the newly generated schedule.

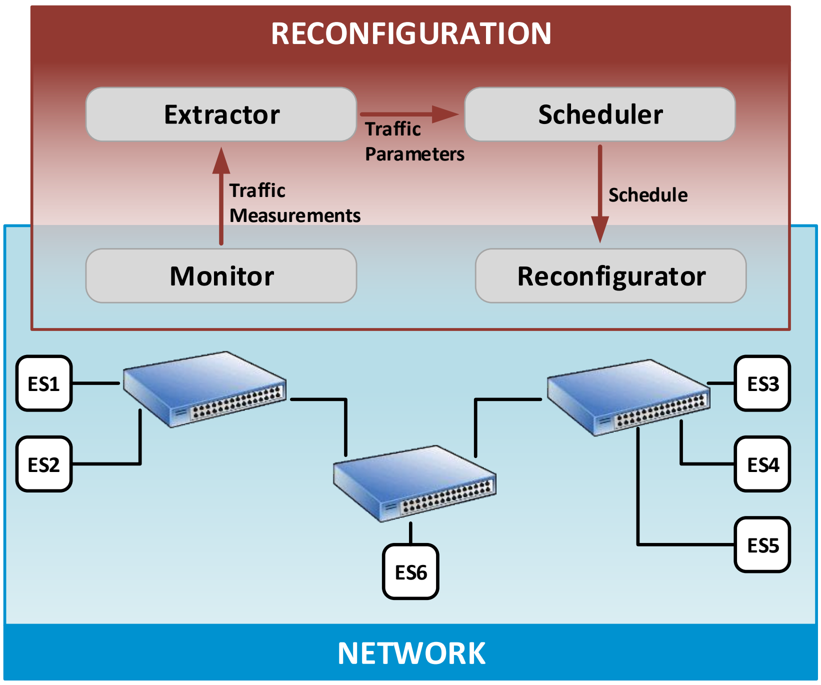

Figure 12: Reconfiguration System - Identifying changes in the network

The approach that is being targeted sees the network reconfiguration as a continued process. This means that after the network has been (re-) configured according to the new schedule, the Monitor will start again to gather information from the network because the traffic parameters might have changed or will change again.

One possible example of a reconfiguration can be the transformation from unsynchronized traffic to synchronized traffic, which is actually very aligned with the latest developments of the IEEE 802.1 TSN task group in which a form of time-triggered communication is being standardized in IEEE 802.1Qbv. Furthermore, TSN is also looking into new ways of networking reconfiguration and for that the concept of Software Defined Networking (SDN) is emerging. One of the main features of SDN is the centralized management of the network which makes it a perfect architecture for the inclusion of the presented reconfiguration concept.

At the time of document delivery, this is still work in progress. In the second version of this deliverable, which is deliverable D2.5 (D2.2b) and due in M30, more detailed information regarding the reconfiguration concept will be presented.

5 Deterministic Communication Network Configuration Tool

The network devices have multiple scheduling modes based on the NETCONF/YANG standard. A TSN configuration tool is under development that will alleviate the user by automatically configuring TSN communication networks. The configuration tool is a browser-based, user-friendly software that makes it easy to model topologies, create schedules and deploy configurations for TSN networks. Offline network configuration is made possible by an intuitive GUI which provides a topology view or table-based editor for managing components and data streams. Schedules are calculated and generated with just one click via TTTech's own-developed built-in scheduling engine [Oliver, 2018], and network components are configured using open, standard YANG models.

As depicted in Figure 13, in the first approach the inputs for the configuration of the network are provided as input manually to the configuration tool. These inputs consists among other out of device capabilities, network topology descriptions, stream requests, etc. With the TSN configuration tool (containing the GUI and internally, the TSN scheduler) a new schedule can be generated that upholds the deterministic requirements that are expected.

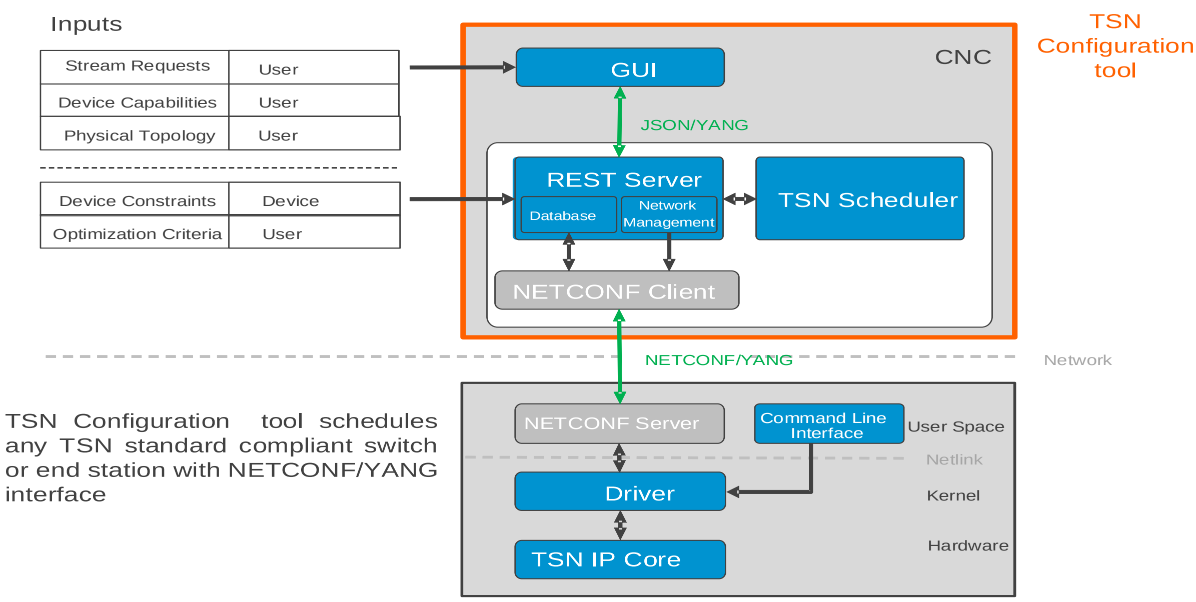

Figure 13: TSN Configuration Tool – Engine

The graphical view enables the designer to model small and medium sized network topologies (see Figure 14). New network components (i.e. switched, end systems, etc) can easily be added by dragging and dropping them into the topology. Data streams can also be represented as logical components between the different components.

A table editor is designed for modelling large communication network topologies and supports mass editing of these network. Components, data streams and other parameters can be configured and inputted into different tables and afterwards viewed in the before mentioned graphical mode.

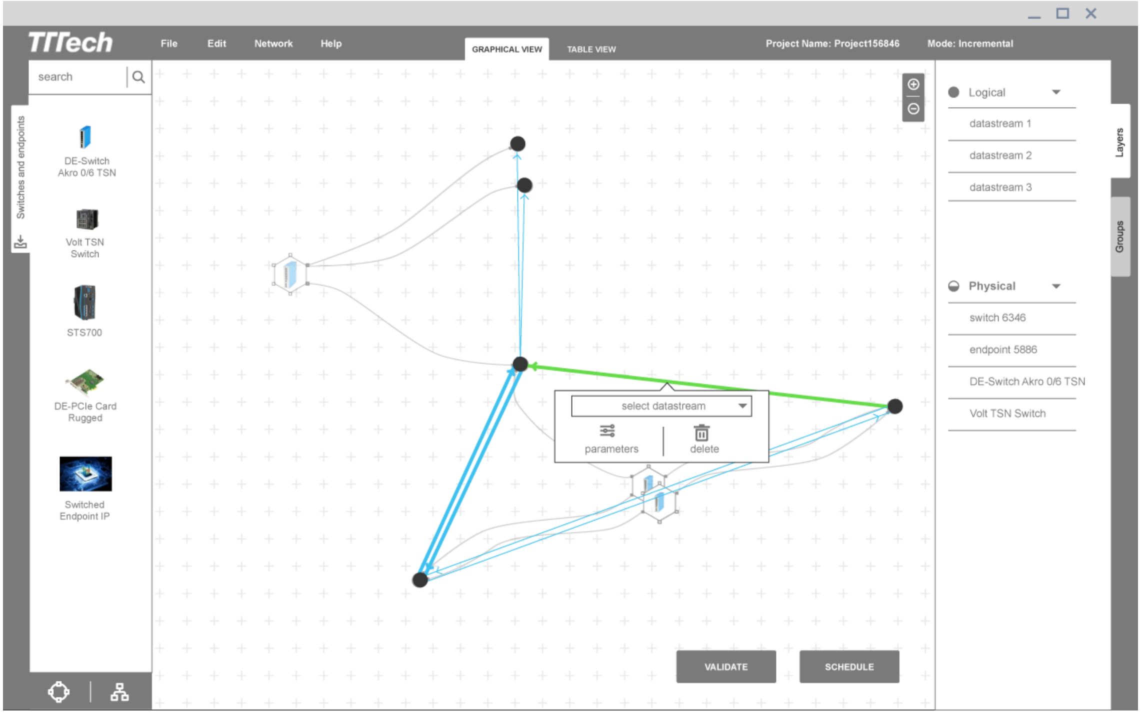

Figure 14; TSN Configuration Tool – User Interface

The tool schedules IEEE 802.1Qbv and 802.1Qbu traffic as well as supporting cut-through data streams. Scheduling can be adjusted to optimize the bandwidth for non-scheduled traffic.

This is currently heavy work in progress. In the final version of this deliverable, D2.2b – Deterministic Ethernet Communication (D2.5) which is due in M30, will give a more elaborated overview of the tool. The final goal of the tool, which goes beyond the scope of the project, is a complete tool which completely automatically generates new schedules based on the changes in the network, without any human interferences.

6 Conclusions and Next Steps

This deliverable provides two major topics in the AUTOWARE project:

- Deterministic communication for future flexible reconfigurable industrial systems
- Investigation and first concepts for a dynamic reconfiguration approach under full consideration of deterministic requirements and changing environments with real-time constraints.

First, the document gives a high-level overview of different Ethernet communication standards that are on the market and used in different areas (like e.g. manufacturing or aerospace). This provides an introduction towards deterministic communication. Two different Deterministic Ethernet standards are then introduced, namely TTEthernet and TSN, where the latter is the main area of focus of the deliverable and the work performed inside the AUTOWARE project.

The IEEE 802.1 TSN (Time-Sensitive Networking) standard is introduced in more detailed, which also provides the introduction to the dynamic reconfiguration concept investigated inside the project and first concepts being described in this deliverable. The focus is on which standards from the TSN set of standards are targeted by the work in this area. The reconfiguration concept is introduced with the autonomous entity that gather the required information from the network to uphold the deterministic requirements. Finally, a network configuration tool is introduced, which is work in progress, which will integrate the reconfiguration concepts for modifying changed communication networks.

Next steps in the AUTOWARE project is continuously updating the TSN concepts, as the set of standards is still under development. Here it needs to be assured that the available technologies will uphold the IEEE 802.1 standard. With respect to the reconfiguration approach, further tests and developments will be undertaken to make a complete framework for reconfiguration of deterministic communication networks based on TSN. Finally, the configuration tool will also be further updated to finally support a reconfiguration process without human interference. Follow-up results will be presented in the upcoming deliverable (which is D2.5) and is due in project Month 30.

References

[ARINC, 2009] ARINC, "664P7-1 Aircraft Data Network, Part 7, Avionics Full-Duplex Switched Ethernet Network," SAE ITC, 2009

[Atzori, 2010] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey", *Computer Networks*, 54(15):2787-2805, October 2010

[Bjorkland, 2010] M. Bjorkland, ,"YANG – A Data Modeling Language for the Network Configuration Protocol (NETCONF)", Internet Engineering Task Force, October 2010

[HSR, 2018] Flexibilis, "High-availability Seamless Redundancy (HSR)", [Online], http://www.flexibilis.com/technology/high-availability-seamless-redundancy-hsr/, last accessed: June 2018

[IEC, 2016] IEC, "IEC 62439-3:2016 Standard," 2016

[Lafou, 2016] M. Lafou, L. Mathieu, S. Pois, M. Alochet, "Manufacturing System Flexibility, Sequence Flexibility Assessment", 49th CIRP Conference on Manufacturing Systems (CIRP-CMS 2016).

[NETCONF, 2011] "Network Configuration Protocol (NETCONF)", Internet Engineering Task Force, June 2011

[Oliver, 2018] R.S. Oliver, S.S. Craciunas, W. Steiner, "IEEE 802.1Qbv Gate Control List Synthesis using Array Theory Encoding", IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS), Porto, Portugal, 11-13 April 2018

[PRP, 2018] Flexibilis, "Parallel Redundancy Protocol (PRP)", [Online], http://www.flexibilis.com/technology/parallel-redundancy-protocol-prp/, last accessed: June 2018