DETERMINISTIC6G

Final Report on DETERMINISTIC6G Architecture

A Dependable Network Architecture for 6G

The DETERMINISTIC6G project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement no 1010965604.

Final Report on DETERMINISTIC6G Architecture - A Dependable Network Architecture for 6G

Grant agreement number: Project title: Project acronym: Project website: Programme:	101096504 Deterministic E2E communication with 6G DETERMINISTIC6G Deterministic6g.eu EU JU SNS Phase 1
Deliverable type:	Report
Deliverable reference number:	D1.4
Contributing workpackages:	WP1
Dissemination level:	Public
Due date:	M30
Actual submission date:	30 June 2025
Responsible organization:	EDD
Editor(s):	Joachim Sachs
Version number:	V1.0
Status:	Final
Short abstract:	The report describes a dependable network architecture for E2E dependable communication with 6G.
Keywords:	Network architecture, system architecture, dependable communication, TSN, 6G, DetNet, Industry 5.0, time-sensitive communication, extended reality, exoskeleton, adaptive manufacturing, smart farming, cyber-physical systems

Contributor(s):	Joachim Sachs (EDD) Edgardo Montes de Oca (MI) Huu Nghia Nguyen (MI) Mahin Ahmed (SAL) Jose Costa Requena (CMC) James Gross (KTH), Gourav Prateek Sharma (KTH) Frank Dürr (USTUTT) Simon Egger (USTUTT) Lucas Haug (USTUTT) János Harmatos (ETH) János Farkas (ETH) Dávid Jocha (ETH) Balázs Varga (ETH) Marilet De Andrade Jardim (EAB)

	Christer Holmberg (LMF) Oliver Hoeftberger (B&R) Ines Alvarez Vadillo (ABB) Ognjen Dobrijevic (ABB) Drissa Houatra (OR) Giulia BIGONI (IUVO) Francesco Giovacchini (IUVO) Filippo Dell'Agnello (SSSA) Emilio Trigili (SSSA)

Reviewers:	James Gross János Harmatos

Disclaimer

This work has been performed in the framework of the Horizon Europe project DETERMINISTIC6G co-funded by the EU. This information reflects the consortium's view, but the consortium is not liable for any use that may be made of any of the information contained therein. This deliverable has been submitted to the EU commission, but it has not been reviewed and it has not been accepted by the EU commission yet.

Executive summary

Dependable, time-critical communication is poised to become a key technology enabler for future 6G networks. This necessity stems from the demand to facilitate a wide range of indoor and outdoor applications with high availability for time-critical services, across various application domains, such as adaptive manufacturing, smart farming, extended reality (XR) and occupational exoskeleton. While the 5G system architecture already includes functional components and solutions to support use cases with low latency requirements, it also reveals certain shortcomings. Consequently, there is a need for additional enablers in 6G to efficiently support these emerging, visionary use cases.

This deliverable provides a comprehensive overview about a system architecture tailored to dependable, time-critical communication, and offers a detailed description of the architecture's deployment for realizing several time-critical use cases. It describes how observability for performance monitoring can be embedded into the 6G network design enabling new forms of performance predictions. Those are valuable to, first, allow for dependable end-to-end traffic management when integrating 6G into time-sensitive or deterministic networks (TSN/DetNet), and second, assure that the network connectivity is fulfilling the performance needs required by the application. By integrating edge computing tightly into the 6G network fabric, also a dependable compute platform can be provided for virtualizing application design, while ensuring service operation for networked time-critical applications.

Contents

1. Introduction

Digital transformation of industries and society is resulting in the emergence of a larger family of time-critical services with needs for high availability which present unique requirements distinct from traditional Internet applications like video streaming or web browsing. Time-critical services are already known in industrial automation; for example, an industrial control application that might require an end-to-end (E2E) “over the loop” (i.e., from the sensor via the controller to the actuator) latency of 2 ms and with a communication service requirement of 99.9999% [3GPP19-22261]. In the same way, with the increasing digitalization similar requirements are appearing in a growing number of new application domains, such as extended reality, autonomous vehicles, and adaptive manufacturing [DET23-D11]. The general long-term trend of digitalization leads towards a Cyber-Physical Continuum where the monitoring, control and maintenance functionality is moved from physical objects (like a robot, a machine or a tablet device) to a compute platform at some other location, where a digital representation – from now on, digital twin – of the object is operated. Such cyber-physical system (CPS) applications need a frequent and consistent exchange of information between the digital and physical twins. Several technological developments in the information and communications technology (ICT) sector drive this transition. The proliferation of (edge-) cloud compute paradigms provides new cost-efficient and scalable computing capabilities that are often more efficient to maintain and evolve compared to embedded compute solutions integrated into the physical objects. It also enables the creation of digital twins as a tool for advanced monitoring, prediction, automation of system components, and improved coordination of systems of systems. New techniques based on Machine Learning (ML) can be applied in application design that can operate over large data sets and profit from scalable compute infrastructure. Offloading compute functionality can also reduce spatial footprint, weight, cost, and energy consumption of physical objects, which is particularly important for mobile components, like vehicles, mobile robots, or wearable devices. This approach leads to an increasing need for communication between physical and digital objects, and this communication can span over multiple communication and computational domains. Communication in this cyber-physical world often includes closed-loop control interactions which can have stringent E2E Key Performance Indicators (KPI) (e.g., maximum packet delay and packet delay variation) requirements over the entire loop. In addition, many operations may have high criticality, such as business-critical tasks or even safety relevant operations. Therefore, it is necessary to provide dependable time-critical communications which provide service-assurance to achieve the agreed service requirements.

1.1. DETERMINISTIC6G approach

In the past, time-critical communication has mainly been prevalent in industrial automation scenarios with special compute hardware like Programmable Logic Controllers (PLCs), and has been based on a wired communication system, such as POWERLINK and EtherCAT, which is limited to local and isolated network domains configured according to the specific purpose of the local applications [ECAT] [PLNK]. With the standardization of Time-Sensitive Networking (TSN) and Deterministic Networking (DetNet), similar capabilities have been introduced into the Ethernet and IP networking technologies, which thereby provide a converged multi-service network allowing time critical applications in a managed network infrastructure aiming for consistent performance with zero packet loss and guaranteed low and bounded latency [IEEE-TSN] [IETF-DETNET]. The underlying principles are that the network elements (i.e., bridges or routers) and the PLCs can provide a consistent and known performance with

negligible stochastic variation, which allows to manage the network configuration according to the needs of time-critical applications with known traffic characteristics and requirements.

Nonetheless, it turns out that several elements in the digitalization journey introduce characteristics that deviate from the assumptions that are considered as baseline in the planning of deterministic networks. There is often an assumption for compute and communication elements, and applications, that any stochastic behavior can be minimized such that the time characteristics of the element can be clearly associated with tight minimum/maximum bounds. Cloud computing offers efficient and scalable computing resources, but it introduces uncertainty in execution times. Wireless communications provide flexibility and simplicity, however they contain inherently stochastic components that lead to significant packet delay variations compared to those found in wired counterparts. Additionally, emerging applications incorporate novel technologies (e.g., ML-based or machine-vision-based control) where the traffic characteristics deviate from the strictly deterministic behavior of old-school control [SPS+23]. In addition, it is expected that there will be an increase in dynamic behavior, where characteristics of applications and network or compute elements may change over time in contrast to a static behavior that does not change during runtime. It turns out that these deviations of stochastic characteristics make traditional approaches to planning and configuration of E2E time-critical communication networks such as TSN or DetNet fall short regarding service performance, scalability, and efficiency. Instead, a revolutionary approach to the design, planning, and operation of time-critical networks is needed which fully embraces the variability but also dynamic changes that come at the side of introducing wireless connectivity, cloud compute and application innovation. The objective of DETERMINISTIC6G is to address these challenges: including the planning of communications and compute resource allocation for diverse time-critical services E2E over multiple domains, while providing efficient resource usage and a scalable solution [SPS+23].

DETERMINISTIC6G takes a novel approach towards converged future infrastructures for scalable CPSs deployment. With respect to networked infrastructures, DETERMINISTIC6G advocates (I) the acceptance and integration of stochastic elements (like wireless links and computational elements) with respect to their stochastic behavior captured through either short-term or longer-term envelopes. Monitoring and prediction of KPIs, for instance latency or reliability, can be leveraged to make individual elements plannable despite a remaining stochastic variance. Nevertheless, system enhancements to mitigate stochastic variances in communication and compute elements are also developed. (II) Next, DETERMINISTIC6G attempts to manage the entire E2E interaction loop (e.g., the control loop from the sensor to the controller to the actuator) with the underlying stochastic characteristics, especially while embracing the integration of compute elements. (III) Finally, due to unavoidable stochastic degradations of individual elements, DETERMINISTIC6G advocates allowing for adaptation between applications running on top of such converged and managed network infrastructures. The idea is to introduce flexibility in the application operation such that its requirements can be adjusted at runtime based on prevailing system conditions. This encompasses a larger set of application requirements that (a) can also accept stochastic E2E KPIs, and (b) that possibly can adapt E2E KPI requirements at run-time in harmonization with the networked infrastructure. DETERMINISTIC6G builds on a notion of time-awareness, by ensuring accurate and reliable time synchronicity while also ensuring security-by-design for such dependable time-critical communications. Generally, we extend a notion of deterministic communication, where all behavior of network and compute nodes and applications are pre-determined, towards dependable time-critical communication, where the focus is on ensuring that the communication (and compute)

characteristics are managed in order to provide the KPIs and reliability levels that are required by the application. DETERMINISTIC6G facilitates architectures and algorithms for scalable and converged future network infrastructures that enable dependable time-critical communication E2E, across domains, including 6G.

1.2. Objective of the document

DETERMINISTIC6G has described several use cases and their requirements [DET23-D11] and developed a dependable network service design for time-critical applications [DET25-D13]. Functionality for time-awareness based on robust time synchronization [DET23-D22] [DET25-D24], 6G capabilities for dependable communication [DET23-D21] [DET25-D23] and approaches for E2E dependable time-critical communication including fixed and wireless domains [DET23-D31] [DET24-D34] [DET25-D35] including security solutions [DET23-D32] [DET24-D12] have been proposed. This includes the integration of edge computing for time-critical applications, based on tight coupling of application execution with time-sensitive network design [DET24-D33] [DET25-D36]. The objective of this document is to describe a 6G network architecture in an E2E context that integrates the functionality listed above and provides dependable time critical communication E2E, building on an architecture framework developed in [DET24-D12].

An architecture can be described with different purposes [RÖT+23]. A *functional architecture* describes functional blocks and their relationships and interactions. It is often the baseline for standardization. An *implementation architecture* describes how functionality is realized in a real system. Often different design choices exist, on how functional blocks are grouped and how they are implemented. The functional architecture should provide sufficient freedom for implementation choices and optimizations. A central part of the functional architecture is to define interfaces where system components of several different vendors are integrated, and where open standardized interfaces enable commercially relevant system realizations. A *deployment architecture* describes how a network is practically deployed in a specific environment. The functional and implementation architectures shall allow for flexible deployments, so that it can efficiently realize the use cases envisioned in the deployment area. The focus of this document is on a functional architecture description, but it also addresses some deployment aspects.

1.3. Relation to other work packages and deliverables

This deliverable is part of Work Package 1 and has linkages with other technical work packages, as presented in Figure 1.1. The deliverable is based on the use cases developed in [DET23-D11], which are utilize edge computing in their application design and require dependable network services as described in [DET25-D13]. The deliverable builds on the architecture framework developed in [DET24-D12].

The architecture developed here embraces the technology components developed in work packages WP2 and WP3. This includes:

- 6G capabilities for performance monitoring and ML-based performance prediction, packet delay correction, and TSN aware traffic management [DET23-D21] [DET25-D23],
- Robust time synchronization [DET23-D22] [DET25-D24],
- Wireless-friendly E2E traffic management in TSN and DetNet [DET23-D31] [DET24-D34], and including multiple network domains [DET25-D35],

- Security mechanisms based on in-network telemetry, threat analysis and mitigation [DET23-D32] [DET24-D12] [DET25-D24],
- Integrated dependable edge computing [DET24-D33] [DET25-D36],
- Situational awareness being provided by digital twins [DET24-D33] [DET25-D36].

Technology components in WP2 and WP3 have been developed and validated by making use of performance and concept validation tools, that have been developed in WP4, and include:

- A simulation framework for performance evaluation of E2E traffic management in TSN networks, which has been enhanced with a model for a 5G and 6G mobile network, network time synchronization, and includes a model for edge cloud compute characteristics [DET23-D41] [DET25-D44] [DET25-D45],
- A network emulator that allows to test and validate time-critical applications over 6G and TSN networks [DET25-D44] [DET25-D45],
- A network emulator for security threat analysis [DET25-D45],
- A latency measurement framework for packet delay measurements in 5G trial networks, and in-depth RAN latency characterization [DET23-D42] [DET25-D43] [DET25-D45].

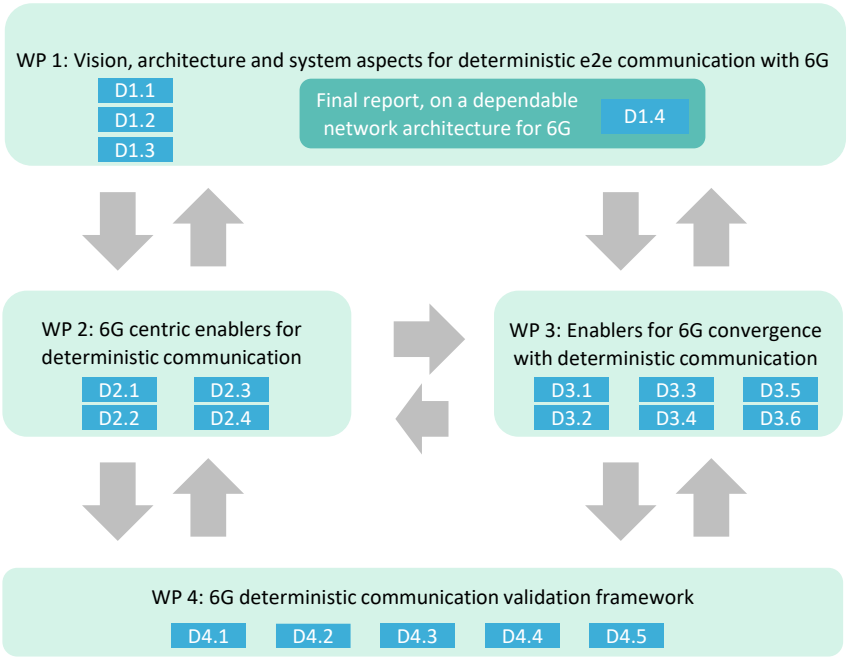

Figure 1.1: Relationship to other work packages.

1.4. Structure of the document

The report starts in chapter 2 to describe the E2E framework for realizing networked time-critical applications in local areas or over wide areas, based on dependable edge computing and dependable networks. Chapter 3 provides an overview of the network architecture principles in 5G that are relevant for a dependable mobile network design and summarizes architecture principles investigated in the research community for 6G. Chapter 4 proposes a dependable 6G mobile network architecture, including the integration with E2E time-sensitive and deterministic networks (TSN/DetNet), as well as a dependable edge compute infrastructure. Also, a migration path from a 5G network towards 6G is discussed. In chapter 5, the realization of use cases for time-critical applications with a 6G network

are described, including use cases in wide area and local area deployments. In chapter 6 the work is concluded.

2. End-to-end connectivity for critical applications

2.1. General

Critical networked applications are distributed applications connected over a network, where a guaranteed minimum performance is required from the connectivity service provided by the network for the application to function correctly, see Figure 2.1. In several use cases, like control, the application operates in a loop, comprising two connectivity services in opposite direction, one that provides timely status reports from the system to the controller, and another which provides control commands from the controller to the system. The critical application has a certain *criticality*, which is expressed in minimum performance requirements for critical connectivity KPIs. Supporting this criticality, by meeting the minimum performance requirements for all critical KPIs, is the necessary condition for the application to work at a desirable level. In addition, a critical application is associated with a certain *severity*, which describes the damage that is perceived, in case that the critical KPIs of the connectivity are not met and the application fails in its operation. For a control application, the damage can be due to loss of stability of the control operation, which can lead to undesired results and in some cases may lead to safety risks or an emergency stop. It can also mean that a certain action cannot be performed, leading to a financial or reputational loss. The severity describes the reliance of the application on the network connectivity service and the importance of the network meeting the critical KPIs. Critical KPIs can be a maximum network latency and / or packet delay variation that is acceptable for the application; or it can be a minimum data rate. In order to trust on a satisfactory operation, a critical application (as service consumer) may want to request a *service level agreement* (SLA) from the network (as service provider), which defines the required connectivity performance for the defined application traffic, and potential penalties if the performance is not met. In other words, the SLA describes a promise by the network for a certain connectivity service, and the severity indicates the financial value of this promise. An SLA also contains the method how KPIs are calculated, measured and proven towards the customer. Technically, a required *connectivity service availability* can describe the promised probability of the network to fulfil the requirements over a time span. The SLA can be restricted to certain geographic areas and time periods.

Generally, a network needs to be deployed appropriately to be able to support applications sufficiently. Further, the network needs to be configured and operated with service assurance to be able to commit to connectivity and performance guarantees. If very high levels of *connectivity service availability* are desired, the network must also account for risks of unforeseen events, such as failures of network equipment or power outages. In this case additional approaches for network robustness and resilience are required [VHC+21] [KSR+24] [BSB+25].

In many cases critical applications can contain multiple modes of operation and support different levels of operation. An application can switch between different levels of operation and each of those can be associated with their corresponding connectivity requirements [DET25-D13]. Typical operation modes that provide higher value to the application, have tighter KPI requirements (like lower guaranteed latency) and / or create an increase in traffic load. For example, a control system can operate faster by doubling the frequency of the control loop, which also doubles the traffic of control commands and sensor reports. In computer vision a scene analysis algorithm can provide better performance if the image/video quality uses a higher resolution or uses a higher frame rate. A

dependable networks ensures that the network connectivity service meets the requirements of the current operation mode of the application. It further enables the application to use the most valuable operation mode when the according performance can be met, but it enables the application also to switch to another mode of operation in case that unfavorable situations challenge the reliability at which the connectivity performance is provided.

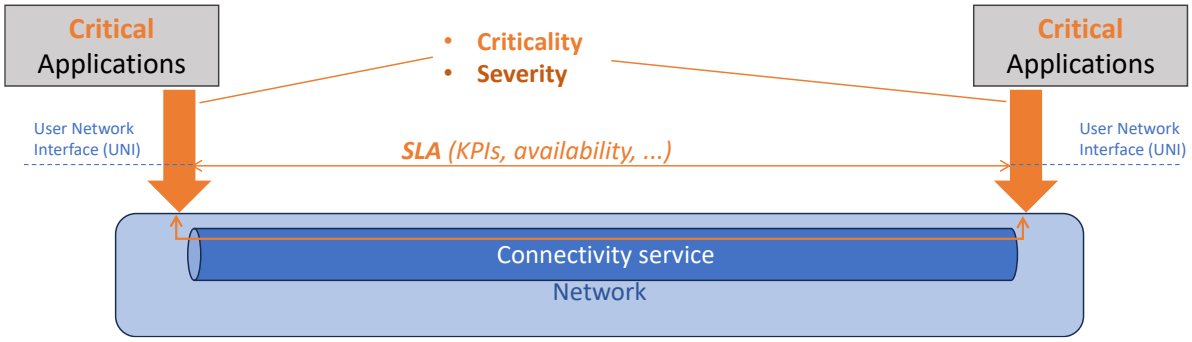

Figure 2.1: Critical application connected over a network.

2.2. Local-area scenarios

For fixed local area networks, the dominating network technology is Ethernet, as specified in IEEE 802.3 and in IEEE 802.1 standards. In general, Ethernet deployments do not allow to provide performance guarantees, for example, a maximum latency. Such performance guarantees are however required, for example in industrial automation use cases. To this end, a set of various fieldbus technologies have been developed and deployed over the last decades, that can provide performance guarantees, like delay bounds. Some fieldbus technologies are based on modifications of standard Ethernet. These different fieldbus technologies are often targeted towards particular applications; they are also limited in their scalability. Practical local network deployments can contain multiple nested fieldbus segments. These different network segments become administrational domains of their own. This heterogeneity of network technologies and deployments leads to high system integration efforts and complexity of network operation and configuration.

TSN has been standardized in IEEE 802.1 as a set of features for standard Ethernet that can support delay-critical applications with deterministic and bounded latency behavior. This allows to deploy local networks based on Ethernet, that can support time-critical and best-effort connectivity and provide a converged network for all applications. Over time, TSN-enabled Ethernet is expected to become the dominating local area network technology for industrial automation [BSB+19].

In local area networks, there is typically a single entity in charge of deploying and operating the local network and support applications in the local domain.

For wireless connectivity, Wi-Fi is a wireless communication technology for local area networks. Even if Wi-Fi can provide very low latencies on average, it cannot provide latency bounds. In particular, at high load situations, large latency tails are perceived [KPB+24]. Mobile networks have been developed for wide-area coverage of mobile and wireless devices. With the introduction of 5G private networks (typically denoted as *non-public networks*), also small-scale deployments of mobile networks in local areas become possible. With 5G, support for ultra-reliable low latency communication (URLLC) has been specified, which enables to provide upper bounded latency with high reliability. While the

standard supports bounded latency down to millisecond level, commercial deployments are primarily targeting latency bounds in the order of 10 ms.

2.3. Wide-area scenarios

For wide-area scenarios, networks are dominantly based on the Internet Protocol (IP) and include often connectivity provided by public networks. IP allows to provide an E2E data networking over several individual networks – called autonomous systems [MRW14] [Toz16] [Nur21] [Wik25] – which are separate administrative domains that are operated by different network operators. What is known as the *public Internet* is the interconnection of approx. 25.000 autonomous systems based on IP [Wik1-25]. In particular, in-transit backbone networks and networks providing virtual private network (VPN) service, IP is often complemented with multi-protocol label switching providing more efficient packet forwarding, virtualization, traffic separation, guaranteed quality of service (QoS) and enabling traffic engineering.

By default, public IP networks provide best-effort connectivity. Distributed congestion control is applied to prevent excessive network congestion (which may result in a severely degraded network performance). Transmitting end nodes participate in congestion control by reducing their sending rate when queues build up in the network, which can be noticed by packet drops or congestion notifications. Enhancements have been proposed to support QoS in IP networks, in particular, for prioritized transfer of streamed media. One approach is based on *integrated services* (IntServ) [BKM+98] [Wik2-25], where bandwidth reservations can be made in IP routers for individual flows. For this, a reservation of resources is made prior to a session on all routers along the E2E path. This allows, in principle, to provide guaranteed data rates for a specific data flow. IntServ has several challenges. It requires that all routers maintain state information for all reserved traffic flows, which limits its scalability and adds router complexity and network costs. Furthermore, in a public network the provision of improved QoS – and performance guarantees – is a value-adding function. To provide incentives for a network operator to support IntServ it needs to be linked to a business support function (to commercially QoS provisioning, e.g. via an SLA) and a policy function (to validate the entitlement of a flow for a requested QoS parameter set). The IntServ reservation is initiated from the sending application and is not linked to a policy and business support function. Furthermore, as Internet paths typically traverse several IP networks (autonomous systems and administrative domains), each of those domains needs to support IntServ and typically lack an incentive to do so. In practice, the IntServ paradigm has never been implemented in public IP networks.

Another approach to improve QoS has been made for IP networks based on differentiated services (DiffServ). DiffServ allows to classify packets to certain traffic classes and mark this in the packet headers in the so-called differentiated services code point. Based on their traffic headers, routers can apply different *per-hop behavior* by appropriate packet queuing and forwarding mechanisms. A set of specific traffic classes has been defined with corresponding recommended per-hop behavior. DiffServ is used in networks for traffic differentiation and traffic engineering. It is used within a single administrative domain and can be extended over multiple domains via bi-lateral agreements between the network operators. However, in a loaded network, DiffServ cannot provide by itself performance guarantees as needed by critical applications as described in section 2.1. It rather facilitates to improve QoS in the network from *best-effort* to *better-effort* for traffic classes of higher importance.

A further improvement of IP networks for QoS and guaranteed performance is DetNet, which has been standardized during the last years, and which is described in [RFC8655] [5GAC24b]. DetNet is in

particular addressing QoS for time-critical applications with bounded loss, delay and packet delay variation. DetNet is targeted towards *confined wide-area networks* provided by a single, or some few coordinated administrative network domains. It applies a software-defined networking (SDN) approach, where a control-plane SDN controller receives information about performance needs for individual traffic flows and configures the traffic-handling data plane nodes accordingly. An SDN-based approach can easily integrate a policy function and connect to a business support function (to commercialize performance guarantees, e.g. via SLAs).

2.4. Impact of distributed compute

In classical networked applications, applications are executed on *end hosts* and the network provides connectivity between end hosts as shown in Figure 2.2. The span of the communication depends on the distance between the end hosts. The end-to-end network may comprise multiple interconnected network domains, in particular for wide-area scenarios.

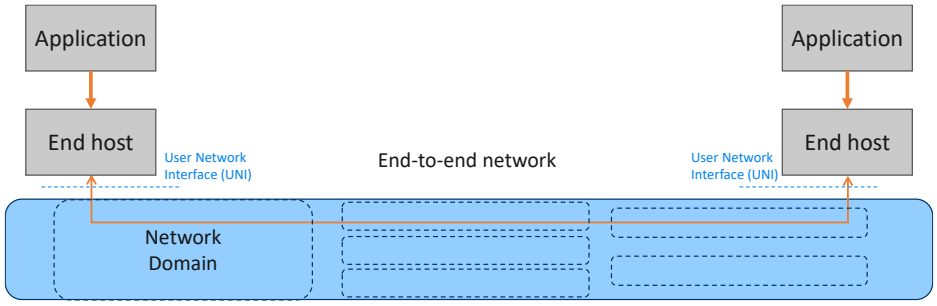

Figure 2.2: End-to-end network connectivity via multiple network domains.

Application design is increasingly embracing the concept of cloud computing, where the application is not installed as is on a particular end host, but it is executed in the compute infrastructure of a data center. This is in particular beneficial for data intensive or compute intensive operations. It has many advantages for application developers, by utilizing cost-efficient scaling of compute workloads and data storage; further, cloud computing can provide compute robustness, e.g. by placing workloads in a geo-redundant way over multiple compute clusters. Cloud computing also impacts how communication and networking between these application components are realized, as illustrated in Figure 2.3. Cloudified applications can be hosted and executed on any suitable data center with the matching cloud execution environment and capabilities. These can be large and centralized data centers or distributed smaller regional data centers. Data centers may be operated by large hyper-scale cloud providers, national or specialized cloud providers or can be private cloud installations. Cloud computing inherently builds on connectivity – connecting the user of an application to the cloud infrastructure hosting the application. Therefore, every data center is directly connected to one or more communication networks. The owners of these communication networks provide connectivity to the cloud. Cloud computing can also be integrated with the communication network infrastructure, where some of the distributed network sites are hosting small data centers. In this case, the network operator in addition to the connectivity service may either provide and operate the compute infrastructure, or the network operator may collaborate with public or private cloud providers for compute infrastructure hosting and integration. For the latter, the network operator can host and co-locate a data center of a cloud provider at the site of the edge gateway of the network.

Cloud computing enables new device and application design principles, as compute capabilities can be expected to be ubiquitously available. This means that a host/device does not need to embed all

its functional elements. It may rather apply ***functional offloading***, where the device manufacturer decides to embed only a subset of desired functionality on the device, and move other functional elements into a remote compute environment. This can have multiple benefits. The device hardware's compute part can be simplified, requiring less processing and storage capabilities. This can reduce the physical footprint, costs, weight, and power consumption of devices. It can further simplify coordinated or collaborative actions among multiple devices, e.g. in a fleet of mobile robots [5GS22-D44] [DET25-D13]. Functional offloading implies that a single application embedded in a device may become a set of networked application components, where the functionality is split among those multiple components, of which one remains on the physical device and the other is executed on some connected compute infrastructure. Cloud computing facilitates and provides benefits in the operation of CPSs [JPK+25], with cyber-physical interactions enabled by the connecting network. While commercial use of cloud computing has grown massively, it is still a challenge to apply cloud computing for time critical operations, since the execution times of functions in a cloud compute platform are typically challenging to guarantee. Real-time dependable computing in cloud environments is an active area of research. A comprehensive overview of how dependable guarantees can be provided by the various components of a cloud ecosystem is provided in [ARS+25]. Offloading of functions from a device to an edge computing execution environment also makes the device operation dependent on sufficient network performance and availability and the performance of the edge computing environment – at least if functions that are critical for the device operation are offloaded. Therefore, functional offloading creates in many cases *critical networked applications* as according to section 2.1.

In many cases at least some component of an application is located on a physical end host. This can be, for example, the automated machine, the physical exoskeleton or XR glasses worn by a person. The other connected components of the application are executed on a remote compute data center. Cloud computing allows the flexible deployment of the cloudified application components in a suitable data center. This can gain benefits of reduced latency, e.g. by choosing a data center in proximity of the user. For mobile networks this is denoted as edge computing, choosing a compute data center in proximity of the user location, as depicted in Figure 2.3. However, in itself, the edge computing paradigm cannot circumvent the latency variation caused by the uncertainties in the compute domain. To mitigate the stochastic nature of cloud deployments and to provide guarantees for the time-bounded execution of time-critical applications, the coordinated use of various real-time support features of the cloud ecosystem is essential. This includes the proper orchestration of application components, considering the capabilities of the virtualization and hardware environment, appropriate configuration of resource allocation and isolation, task (CPU) scheduling and the use of the real-time operating system features (e.g., deterministic task scheduling). In some cases, optimized deployment of the virtualization platform (e.g., bare metal), as well as the use of hardware accelerators (e.g. GPU) and hardware offloading may also be required to achieve performance levels comparable to native execution for real-time applications. Furthermore, instead of traditional, best-effort traffic handling, time-aware scheduling methods must be applied in order to ensure timing guarantees for packet forwarding in virtualized networking of the data center hosts. Different scheduling schemes — such as TAPRIO[1]-based time configuration or ePBF[2]-based packet timing propagation — can be applied, depending on the operating system and the used virtualization technique. In this way, ***dependable***

[1] https://man7.org/linux/man-pages/man8/tc-taprio.8.html
[2] https://ebpf.io /

compute services can be provided by designated cloud premises, going beyond the traditional best-effort compute services.

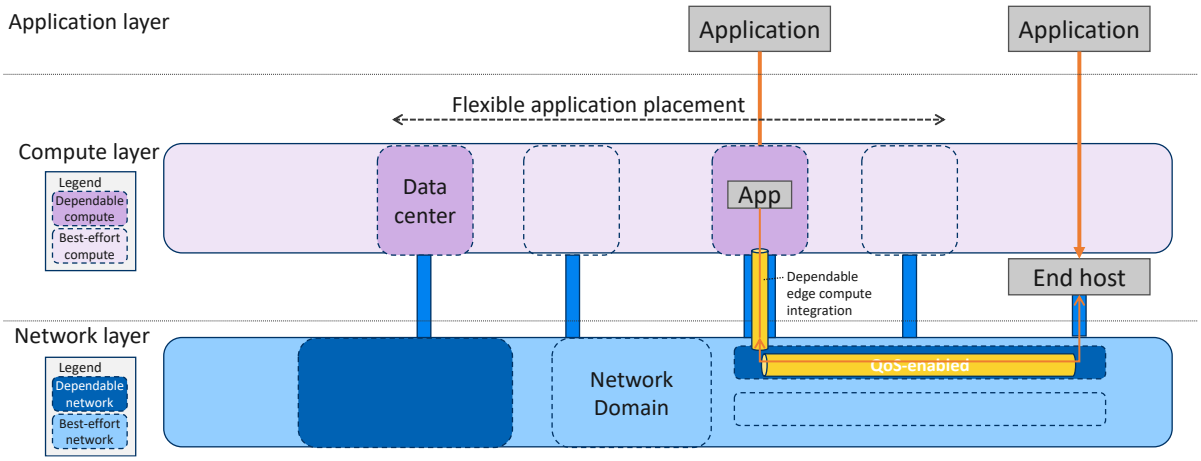

Figure 2.3: Dependable E2E service delivery with a distributed compute infrastructure: a dependable compute location is chosen within reach of a dependable network domain, which is integrated with the network.

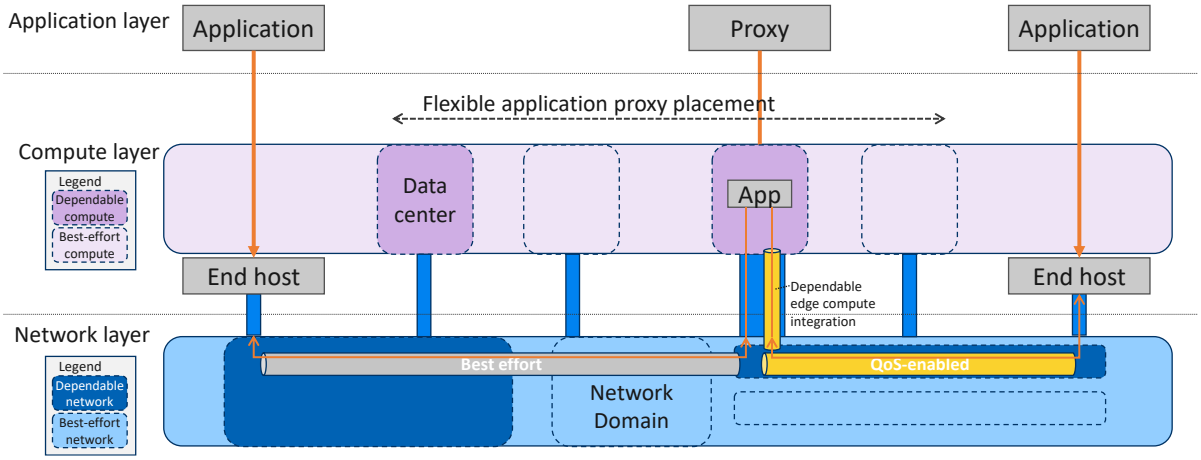

Figure 2.4. Example of remote control over large distance, with a tele-operation proxy being connected in a dependable fashion with the tele-operation a dependable compute location is chosen within reach of a dependable network domain, which is integrated with the network.

There are some situations where cloudification of the end application is not possible. One example is tele-operation, where e.g. some machine is remote controlled over a long distance. In this case, the physical execution components of the applications are on end hosts. This can be a remote controlled machine at a particular location on one side, and the remote operation control center at the other side, from where a human operator is controlling the machine. Alone the speed at which signals are transferred over large physical distance, like in-between countries or continents, can introduce latencies that can exceed 100 ms. Furthermore, QoS support with guaranteed performance can practically not be achieved over such distances (see section 2.3) and variable performance has to be expected. For networked control operations with performance variations, model-based control operations have been developed that enable to compensate for network performance uncertainties. For example, in model-mediated remote control, a control-proxy is added to stabilize the control

[SAA+19], which can be located to provide guaranteed performance to the controlled machine, as shown in Figure 2.4.

3. Review of mobile network architecture concepts for 6G

The system architecture of DETERMINISTIC6G proposed later in chapter 426 builds on earlier architecture work, which is summarized in this chapter. Already the 5G system architecture comprises components that are relevant for DETERMINISTIC6G and are expected to be reused in the migration from 5G towards 6G. Furthermore, other technology components have been developed and are expected to become integrated into a 6G system architecture. This section builds on and extends the architecture concepts described in [DET24-D12].

3.1. Relevant 5G system architecture components

In this section we review those architecture concepts of 5G that are relevant for a dependable network architecture.

3.1.1. Support for non-public networks

In many typical cases the need for dependable communication appears within a specific system of distributed entities. This distributed system contains a well-defined number of end hosts and functions, and it operates by interactions within this closed group of end hosts and potentially their respective cloud compute environment. It is typically desirable to isolate communication of the distributed system. This can have many reasons: to increase the security of the distributed system, by isolating it from other entities that are not part of the distributed system; to manage performance in the inter-connection of the distributed system entities, by limiting the resource management to a limited number of known or anticipated interactions; or to define a specific business agreement (e.g. an SLA) for the interconnection for the distributed system. The network dedicated to the interconnection of the distributed system is sometimes denoted as *private network*, or *dedicated network*. In the standards for mobile networks specified in 3GPP, they are denoted as *non-public networks*[3] (NPN).

NPNs can have various spatial dimensions. Some NPNs may be local, covering only a confined space in which the distributed system is operating. A typical example is an NPN covering an industrial plant or a factory, a mine, or a port. Other NPNs can cover very large areas. Examples include mission critical networks provided to national security and public safety agencies, such as police and fire brigades. Another example is the future railway mobile communication system, which includes use cases from communication among rail personnel to automated train operation [CFL+21].

NPNs may potentially target some public network services. By design, NPNs are intended for private application categories. This can be, e.g., the automation systems in an industrial installation. But in some cases, it may be beneficial to enable even public network services via the NPN. For example, in a mine, it may be beneficial to not only provide automation and remote operation of mining machinery in the mine, also public network services such as voice and data, including emergency calls, may be useful within the mine.

[3] Mobile networks are by their origin defined as *public networks*. Any device that supports the appropriate mobile network standard and obtains a subscription to a particular mobile network, can use the network. With 5G, it has been introduced that also mobile network standards can support *dedicated network* services, resulting in the term *non-public network*.

To this end, 5G has specified two NPN variants, which allow a wide range of configurations of the NPN. The standalone NPN (SNPN) creates an isolated network that is dedicatedly deployed for the NPN user. The SNPN is standalone, i.e. separated from public network installations, and often uses local licensed or leased spectrum [PHB+25]. An SNPN allows the option of RAN sharing with a public network, which would improve the public network service coverage and capacity in the SNPN area and can improve coexistence by reducing potential interference between public and private networks [5GS20-D14] [5GS21-D15] [CAS+22] [CAS+23]. An SNPN is often the preferred type of installation for local industrial deployments. Even if SNPNs are separated from public networks, it is not uncommon that public mobile network operators provide SNPNs to end users, due to their experience in deploying and operating mobile networks. In contrast, a public-network integrated NPN (PNI-NPN) is a NPN that partly reuses public-network infrastructure in its provision of the NPN service. One example to realize the NPN is to establish a *network slice* of the public network, which is dedicated to private network services for the closed group of private devices. PNI-NPNs are in particular relevant, the more the targeted service area of a private network overlaps with public network service coverage, and where an existing public network thus provides a suitable infrastructure foundation for the private network service. In particular, this holds if the private network services are temporary (i.e. nomadic) and an installation of a dedicated network infrastructure creates large overhead. Examples for private network services relevant for PNI-NPN are connected construction sites, smart farming [DET23-D11] [DET24-D12]. Other examples include media production, when a group of reporters need to connect their cameras and production equipment temporarily at the location of some event.

Figure 3.1 depicts options of NPNs. More information on NPNs can be found in [DET24-D12, section 2.1.1] and also in [5GAC21b] [GLS+22] [5GS20-D52] [5GAC24a] [3GPP18-23501].

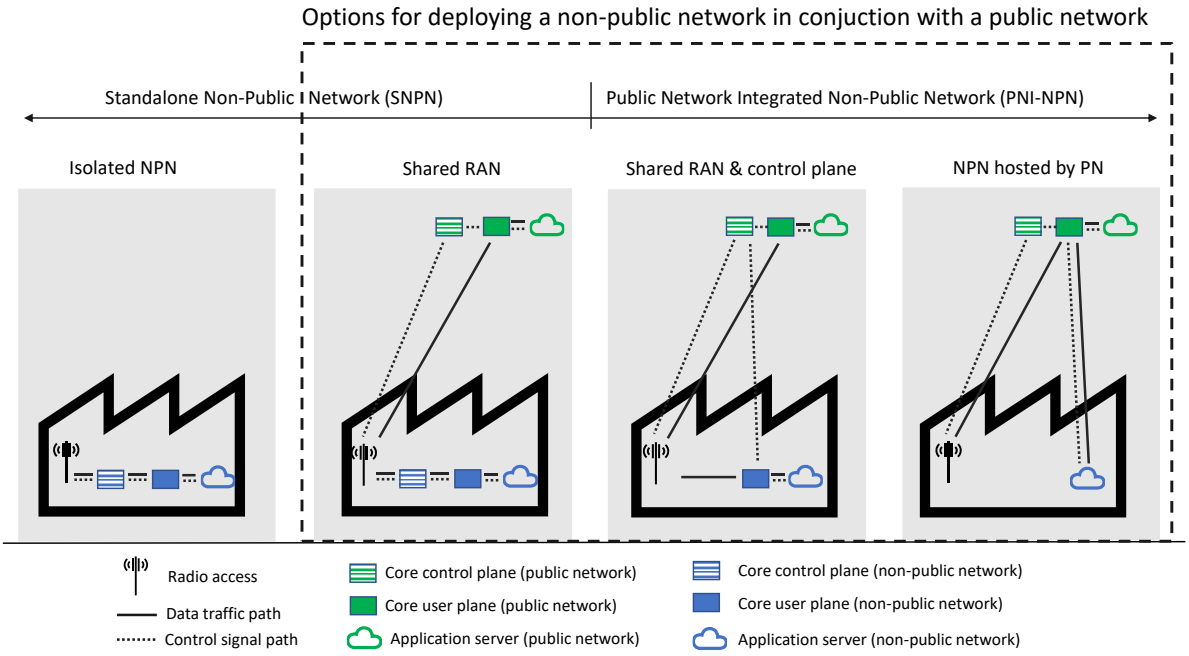

Figure 3.1: Non-public networks (based on [FHB+24]).

3.1.2. Support for TSN and DetNet in 5G System

3GPP introduced the time-sensitive communications (TSC) framework in 5G in order to support deterministic type of communications. This is, in particular, TSN, as IEEE-specified enhancement to

Ethernet in local area networks, and DetNet, as IETF-specified enhancement to IP in local and wide-area networks. The main enabler to support these technologies in the RAN is URLLC which guarantees a bounded delay, delay-critical type of QoS, and reliable communication [LSW+19] [SWD+18] [AKP+21].

With the TSC framework the 5G system is modelled as a virtual node which mimics the behavior of a regular fixed node that supports time-sensitive communication [5GS20-D51] [3GPP18-23501] [5GAC21c] [5GAC24b]. Figure 3.2 shows an example of how a 5G logical TSN bridge is placed in a TSN network. Specifically, the 5G logical TSN bridge is connected to other TSN bridges or TSN end stations (such as Talker or Listener). The 5G logical bridge also interfaces an external control plane management entity, in this case the TSN controller (aka Centralized Network Configuration (CNC) element). Note that in the case of a fully centralized architecture in TSN, applications in the Talker or Listener (aka TSN end stations) communicate with a Centralized User Configuration (CUC) element which oversees user configurations, which then are passed in terms of communication requirements to the CNC, the network controller. After the CNC has collected all TSN flows' requirements from the CUC and all bridge components' capabilities, the CNC sets the configuration for all the network components (TSN bridges and TSN end station Network Card Interface (NIC)) [IEEEQcc] [IEEEQdj] using a configuration protocol such as NETCONF [RFC 6241] or RESTCONF [RFC 8040].

To interface the 5G system with the other TSN nodes, 3GPP introduced TSN translators (TT) at the device side (DS-TT) and at the network side (NW-TT), as shown in Figure 3.2. The translators provide the TSN Ethernet interface and mimic the expected behavior of a number of TSN features, without the need to radically modify the functionalities of the 5G nodes and functions. To interface with the CNC, a TSN application function (TSN AF) was defined. The TSN AF obtains the 5G bridge capabilities and provides them to the CNC, while the CNC configures the 5G bridge via the TSN AF. The TSN AF translates the capabilities to the parameters that the CNC handles and translates the configuration from the CNC into a flow set up in the 5G system. Finally, the TSN AF also handles the typical time synchronization data sets that can be configured by the CNC.

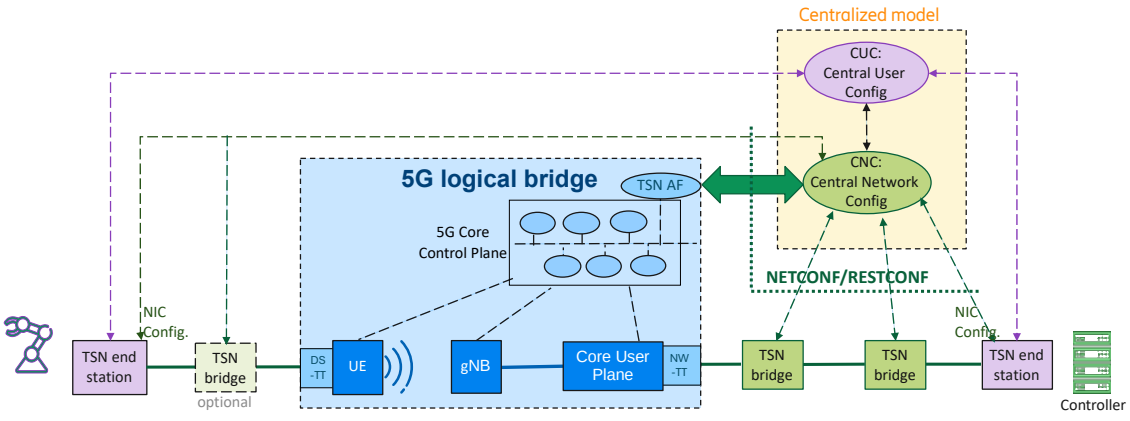

Figure 3.2: 5G system acting as a TSN bridge.

Note that TSN is a specific case of TSC. In general, TSC would involve the use of a similar entity to the TSN AF, known as the TSC and Time Synchronization Function (TSCTSF). TSCTSF performs similar tasks as the TSN AF but with a generalized exposure interface that is proxied via the Network Exposure Function (NEF) as shown in Figure 3.3, unless the external Application Function (AF) is a trusted entity

for the 5G system, in which case the NEF is not needed. In the general case, any AF can request a deterministic service via 5G and the 5G system is considered a node.

Figure 3.3. TSN as a special case of the 5G Time-Sensitive Communications (TSC).

A similar technology to TSN has been standardized in IETF, namely DetNet, which is implemented in layer 3 of the OSI reference model supporting IP-based communications. Similarly, 3GPP has modeled 5G system as a logical DetNet node, just as illustrated in Figure 3.4.

In the case of DetNet, the general TSC framework was reused where the TSCTSF interacts with the DetNet controller (instead of AF) without the need for NEF since the controller is considered a trusted entity. The information used in this interface TSCTSF-DetNet controller uses the defined YANG models for DetNet. TSCTSF translates the configuration from the DetNet controller into requirements towards the 5G system, similarly as in the case of TSN AF. With DetNet there is no requirement for a DS-TT since the interface is IP-based which is already supported in 5G, and no additional translation is needed. If a time synchronization service is required, then the DS-TT will be required.

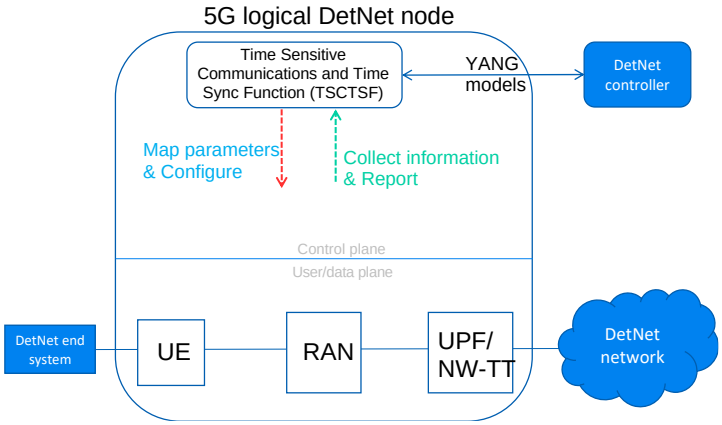

Figure 3.4. 5G logical DetNet node.

3.1.3. Network Exposure

Network exposure provides a means for configuring and monitoring the network and the communication services by network external functions. It is based on Application Programming Interfaces (API) to provide a level of network programmability that allows to customize the network services to the desired use cases. [5GAC21a] has defined the requirements on network exposure for 5G non-public networks to be able to address industrial use cases. Network exposure requirements are grouped into capabilities. One group is related to device management, which includes the device connectivity management and connectivity monitoring that allow to establish communication services to different applications that then provide the required QoS [5GAC21d]. Another group of exposure requirements is focused on network management. [5GS21-D55] [SK23] [GSD+22] [KDS+23] describe how network exposure can be applied in industrial control use cases. It is also worthwhile mentioning ongoing 3GPP efforts on 5G network exposure APIs, namely 5G NEF [3GPP23-29522] and 5G Service

Enabler Architecture Layer for Verticals (SEAL) [3GPP23-23434], to address some specific aspects of TSC. The latter exposure features relate to, e.g., providing application QoS requirements to the network and establishing a TSC session in 5G networks.

Network programmability via APIs is not only relevant for non-public networks, but it also plays a role in wide-area and public mobile networks. These APIs allow to make network capabilities consumable by end users [SKM+21]. Requesting differentiated capabilities from the network can include commercial agreements to be initiated via network exposure [FMK23] [OOA+25]. In recent times, new public network exposure initiatives have started that specify industry-aligned APIs to provide application developers easy ways to make public network usage flexible and fit for the purpose of applications. For example, *the Telco Global API Alliance* (CAMARA) is working on standardized network APIs that allow to configure dynamically a *dedicated network* that is only accessible by a configured set of devices in a certain geographic area and for a specific time [CAM24a]. It is also possible to request specific QoS support for an application [CAM24b].

3.1.4. Support for edge computing in 3GPP

Edge computing leverages the distributed computing paradigm and provides an ecosystem where the execution environment (e.g., compute and resource storages) is closer to the location where the task is invoked compared to the traditional cloud computing paradigm. The proximity of edge premise results in reduced latency between a client and the server application, so edge computing is able to support use cases where low latency is a crucial requirement. Hence, edge computing enables to realize use cases where time-critical applications are moved to the virtualized environment, instead of using dedicated, specialized hardware. Furthermore, the cloudification makes possible the further evolution of the applications by leveraging the cloud-native design paradigm. Furthermore, edge computing allows to connect cloud-hosted applications directly from a cloud infrastructure, to which a dependable network connectivity can be provided. This avoids the need for providing connectivity though a number of transit networks, for which the dependable connectivity cannot be controlled by a single communication provider, as discussed in section 2.4.

3.1.4.1. 3GPP edge computing support architecture

3GPP SA2 group introduced features to support edge computing defined in the technical specification (TS) 23.548 [3GPP23-23548]. This specification outlines three connectivity models supported by the 5G core to enable edge computing. It defines various functionalities for traffic steering and User Plane Function (UPF) selection for realizing these different connectivity models. TS 23.548 provides a detailed description of how the 5G Core supports the Edge Application Server (EAS) discovery/re-discovery in the case of the various connectivity models.

The TS 23.588 [3GPP23-23558], specified by the 3GPP SA6 group, describes an Application Layer Architecture for enabling edge applications, which is illustrated in Figure 3.5.

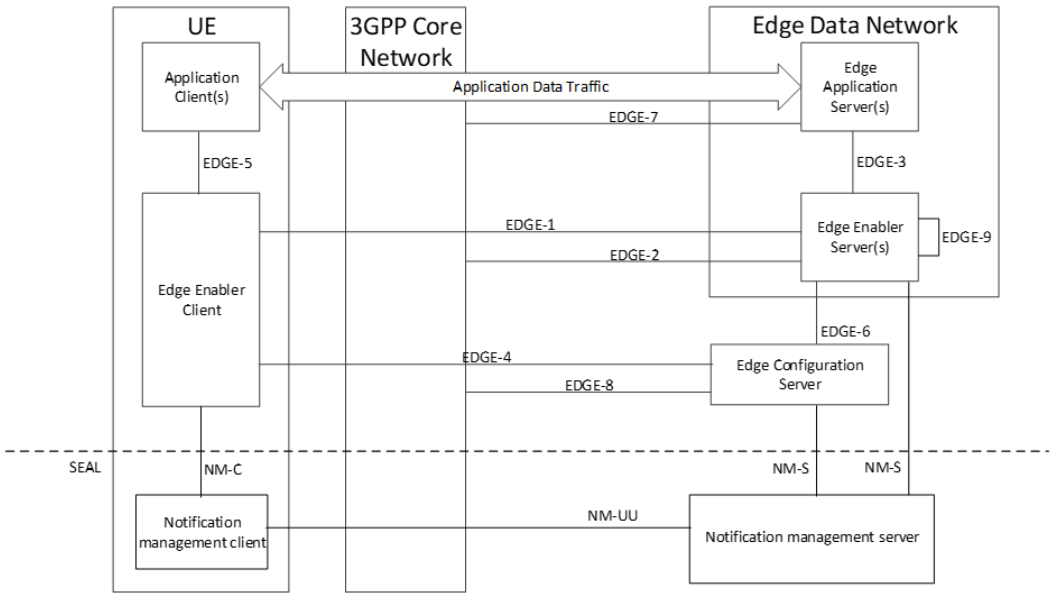

Figure 3.5. Multi-access edge computing framework (Ref ETSI GS MEC 003 V3.1.1).

This architecture makes the user equipment (UE) edge-aware, which means that all the devices include an EEC. The EEC communicates with the Edge Configuration Server (ECS), which provides the required configuration and supporting functions to setup a data session from the application client (hosted by the UE) to the EAS.

3.1.4.2. ETSI MEC Architecture

Multi-access edge computing (MEC) is an application scenario of edge computing used for mobile networks, defined by the European Telecommunications Standards Institute (ETSI). The objective of the MEC as defined in ETSI is to create a continuum between the telco and IT-cloud worlds. The ETSI architecture provides a generic architecture to facilitate the integration of IT and cloud computing capabilities with mobile networks. Figure 3.6 shows the components defined in the ETSI architecture that is structured in three layers. The lower layer consists of the network infrastructure that provides the basic connectivity i.e., local network and 3GPP mobile network, between the devices and the MEC platform. The middle layer provides the platform that is hosting the edge computing infrastructure, including the virtualization components required to run the edge applications and the management system that handles the available resources on the host where the MEC platform is deployed. The higher layer consists of a system level management that provides overall visibility of the devices and edge computing platform.

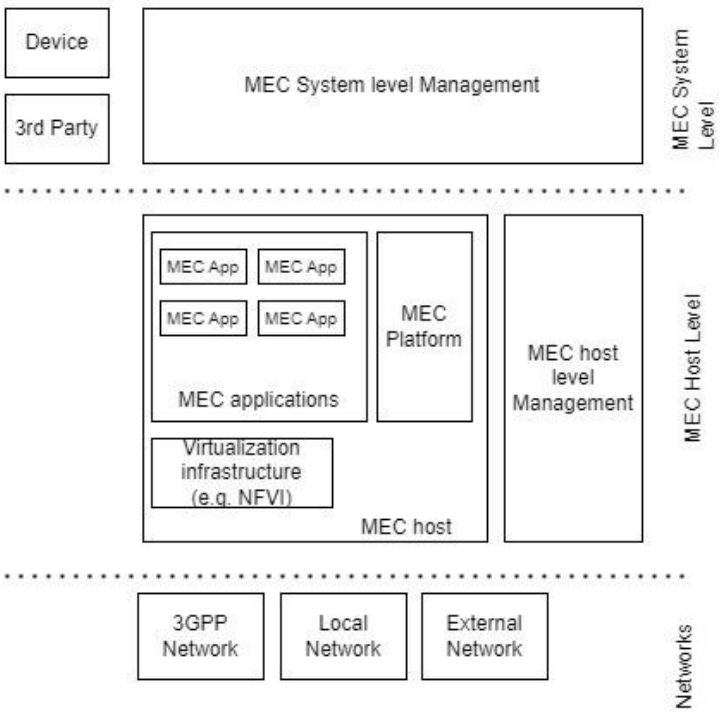

Figure 3.6. 3GPP TS23.558 defined edge computing architecture.

3.1.5. Review of AI/ML in 5G

The operations of mobile networks have been transitioning from manual processes towards sophisticated automated process flows as the networks become much larger and more complex. ML and Artificial Intelligence (AI) have been envisioned to be crucial technologies for enabling automation of network operations (e.g., resource optimization, fault prediction, security policies) in future mobile networks. In recent years, techniques based on AI/ML are being proposed to be used across various domains of the 5G system, including operations, administration, and management (OAM) (e.g., Management and Orchestration), 5G core network (CN), or radio access network (RAN) (e.g., AI/ML-enabled RAN intelligence).

The Network Data Analytics Function (NWDAF) is a key component of the 5G CN, introduced in Release 15 by the 3GPP to enhance 5G CN capabilities with respect to AI/ML [3GPP17-23288]. The primary objective of NWDAF is comprehensive data collection and analytics functionalities. To this end, NWDAF is responsible for gathering data from various sources within the 5G system and for providing analytics to the consumer network functions (NFs). NWDAF can interact with different entities in the 5G system as shown in Figure 3.7.

The first interaction domain is the 5G CN itself where the NWDAF is located. Here, various NFs can be the producers of the data towards NWDAF and also the consumers of its generated analytics. For example, the Access and Mobility Management Function (AMF) and the Session Management Function (SMF) might produce data regarding user mobility and service usage, which can then be analyzed by NWDAF to optimize handover protocols and session resource allocation across different network cells [3GPP17-23288]. The second interaction domain for NWDAF is OAM. The OAM can feed data to NWDAF which it has obtained from the measurement probes located in RAN and relevant 5G NFs. In the OAM domain, the Management Data Analytics Function (MDAF) has been specified which

is also responsible for the interactions with NWDAF [3GPP16-28533]. Lastly, the third set of interactions for NWDAF is in the service domain. Functions residing outside the scope of the 3GPP trust domain can provide data or consume analytics from NWDAF through AFs. For instance, the NWDAF might produce statistics and predictions about user service quality which is consumed in the service domain through an AF.

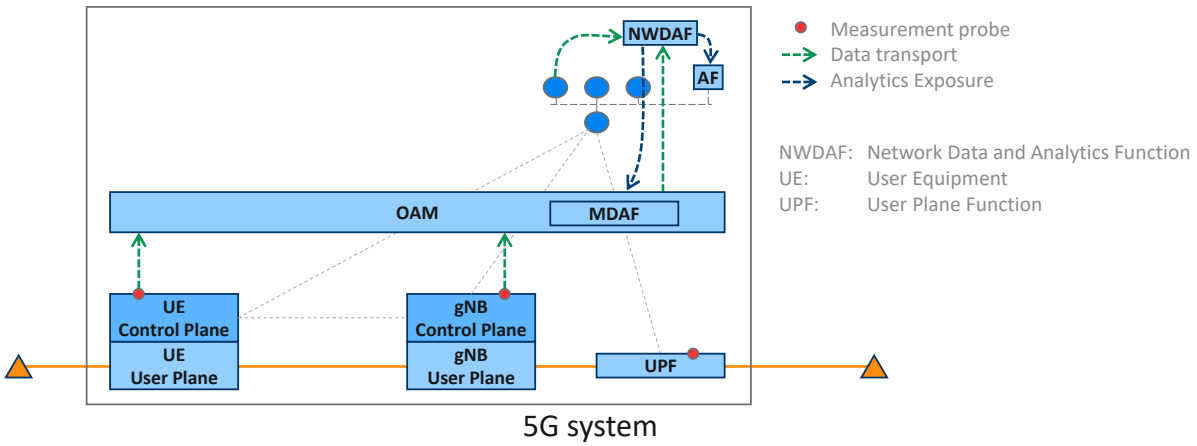

Figure 3.7: An illustration of data collection and exposure in 5G based on NWDAF.

It is important to point out that the Data Collection Coordination Function (DCCF) has been specified to avoid duplication of requests for data as well as analytics between various NFs and NWDAF. In other words, all requests for data and analytics are sent to DCCF which might further rely on a messaging framework to collect analytics and deliver it to the NFs.

To facilitate the deployment and operation of AI/ML capabilities in the 5G system, machine learning models need to be managed throughout their lifecycle. To this end, the functionality in the NWDAF is handled by two logical sub functions: (i) Analytics Logical Function (AnLF) and (ii) Model Training Logical Function (MTLF) [3GPP17-23288]. AnLF using DCCF can access the data collected and applies ML models to generate predictions, e.g., predicting network congestion based on UE mobility patterns. On the other hand, the MTLF is responsible for building and refining the ML models that AnLF utilizes. MTLF is tasked with training and updating models using the data collected.

A functional framework for AI-enabled RAN intelligence has been described in [3GPP22-37817]. This framework describes the key functional entities relevant for integrating intelligence into RAN for selected use cases. The framework serves as a good foundation to explore the architecture aspects of integrating data-driven (AI/ML) approaches for latency prediction into the 5G-Advanced (5G-Adv) / 6G architecture. Utilizing AI for enhancements of RAN performance is also explored in the AI-RAN Alliance [AIRAN24].

3.2. Architecture principles investigated for 6G

A large number of technology components are currently investigated and researched as potential candidates for a future 6G standard. With the upcoming start of 6G standardization, there is a need to assess the potential benefits and level of maturity of those different technology components, to agree within the ecosystem on the technology foundation for 6G. Furthermore, agreed technology components need to be aligned within a consistent architecture view on the 6G system.

A review of technology trends that find wider interest and support in the ecosystem, and which are relevant to the objectives of DETERMINISTIC6G has been made in [DET24-D12, chapter 2].

The main architecture principles that are considered as basis for 6G are:

Cloud-native and software-based network design

By decoupling network functionality from the execution platform, networks can be designed in a more flexible way. A cloud-native compute infrastructure based on distributed data centers serves as basis for the execution environment. The journey of softwarization and cloudification of the network has already started in 4G and 5G. In particular the radio access functionality is characterized by compute-intensive operations and is still mostly based on dedicated and highly optimized compute platforms. We expect that for 6G, cloud-native design approaches will increasingly be used also for RAN, in particular, with increasing integration of hardware accelerators into the compute domain. Cloud-native softwarized network design facilitates functions like network slicing.

Compute as a service

With an inherent distributed compute platform as basis for the network realization, the 6G network platform can also provide compute-as-a-service to the application domain [HEX2-D21] [RJS+23] [BBW+23]. This is in particular beneficial for offloading compute workloads of mobile devices, in order to provide advanced compute resources to the device applications and reduced device footprint, power consumption and complexity of the device (see section 2.4). In particular, for critical applications with a need for guaranteed performance, the combined handling of the compute and the network domains is required. Network-integrated edge computing provides a unique opportunity going beyond what is possible in distributed public cloud infrastructure of today: to provide dependable connectivity and compute services E2E, as described in section 2.4.

Intent-based management

The complexity of configuring and managing networks is increasing, at the same time as there is an increasing need to reduce operational costs and provide simplified solutions. *Network automation* is envisioned to simplify the configuration and operation of the network, following an intent-based management approach [Eri23] [HEX2-D21]. Instead of rule-based configuration requirements, goals and constraints are formally specified to be realized via an autonomous cognitive intent-based management framework.

Data-driven and AI native design

There is a general understanding that AI, and in particular ML, is going to play an increasing role in future networks, but also in the application domains and systems that use networks [HEX2-D21] [HEX-D14] [Eri23] [AIRAN24]. AI technology has matured and is considered beneficial to address problems that are highly complex, or that comprise inherent randomness and non-determinism [IJR+23]. While AI can be introduced by enhancing system components with AI, or introducing new AI-based components, a next level system design is to become AI native [HEX2-D21], which is in [IJR+23] defined as: “the concept of having intrinsic trustworthy AI capabilities, where AI is a natural part of the functionality, in terms of design, deployment, operation, and maintenance. An AI native implementation leverages a data-driven and knowledge-based ecosystem, where data/knowledge is consumed and produced to realize new AI-based functionality or augment and replace static, rule-

based mechanisms with learning and adaptive AI when needed." This leads to the following four aspects [IJR+23]:

- Intelligence everywhere: AI workloads should be executable where it is beneficial from a cost-benefit perspective. As the number of AI models grows, automation is needed, including automated model lifecycle management and hardware dependencies.
- A distributed data infrastructure is needed that enables data collection and data transport in alignment with possible constraints related to data handling. This allows for data-driven intelligence leveraging AI capabilities. More detailed discussion on a data-driven network architecture is provided in [Roe20].
- It is increasingly complex for human operators to manage the network and data infrastructure, which calls for zero-touch management with autonomous networks based on intent-based design (see above).
- AI-as-a-service (AI-aaS): as the network architecture becomes AI native, it integrates AI-related capabilities, such as AI model lifecycle management or data handling. Such capabilities could be exposed to external (network) users as platform services provided by the network infrastructure. More details on a possible AI-aaS service capability can be found in [SAR+23] and also in [HEX2-D21].

Value-based network services, network programmability via APIs and services beyond communication

It is important that the 6G network can provide value to use cases and applications and achieve this in a commercially viable way. For example, it shall be possible to define SLAs or operational agreements between the application domain and the network domain, in order to establish desired networks services. One important direction is to make the 6G network programmable via APIs. Network and service exposure functionality enables the interaction of the application / user domain with the network via standardized APIs, to configure service requests, configure and use network service capabilities [Eri23] [HEX2-D21], which turns the network into a programmable service platform [ABJ+24] [OOA+25] [BSB+25]. Services provided by the 6G platform are not only connectivity services but also comprise services beyond communication [HEX2-D21][ABJ+24], such as time synchronization, compute-as-a-service, positioning, sensing, and AI-aaS.

Network simplicity and commercial relevance

One important aspect that has been identified for 6G is to specify a standard that strives for a clear and simple design and avoids specifying multiple alternative standardized realization options [3GPP25-6GWS] [3GPP25-6GWS2]. Furthermore, from a standardization perspective it is important to identify the commercially relevant multi-vendor interfaces in the architecture that provide commercial value, see [Eri23] [CMRV+23]. With this focus, the 6G network architecture shall allow flexible service innovation for a large variety of use cases, in an interoperable and industry-aligned way that is commercially viable.

4. Architecture for E2E Dependable Communication

4.1. Overview of a system architecture

An initial architecture for dependable communication has been described in [DET24-D12] and is shown in Figure 4.1. A *deterministic network* layer is depicted above the 6G system and indicates that

for time-critical applications a specific E2E data network with support for deterministic communication is expected to be used. To this end Ethernet TSN are proposed as deterministic network technology for local area networks, and IP DetNet is proposed as deterministic network technology in IP networks which can cover wide areas. This deterministic network may end on device side at the 6G UE, but it can also extend to a local deterministic network via a 6G UE gateway to a network segment behind the UE. Applications can be hosted on end-hosts connected to the deterministic network, or they can be hosted in an edge computing infrastructure. In some cases, the applications may use a middleware framework to realize their time-critical services, which in turn uses the E2E deterministic networks available. An example is OPC UA with its extension to (dependable time-critical) field level communication (denoted as field exchange), which is being specified for a wider range of industrial use cases. The 6G system enables wireless and mobile connectivity towards the higher E2E deterministic networking layer. A mobile network, like the 5G system, has a non-deterministic delay characteristic due to the inherent stochastic properties of wireless communication [DET24-D12], and similar characteristics are expected for a future 6G system. The non-deterministic delay characteristics of the mobile network can violate the traffic management assumption of some functions of the TSN or DetNet network layer [DET23-D31], resulting in significant limitations of applying time-scheduled traffic management E2E with a mobile network in the path [DET23-D31] [EDV+25]. However, if the stochastic characteristics of 5G or 6G network delay are known, an E2E traffic management can be devised such that high capacity network performance can be achieved which provides guarantees on achievable E2E delay performance [DET24-D34] [EDV+25] [EGS+25]. In other words, by accepting stochastic variance and accommodating for it in traffic management, dependable communication services can be achieved for time-critical applications even over networks with non-deterministic delay performance. To enable this in the base TSN standard, we have proposed a standardized extension to the TSN control [IEEEQee].

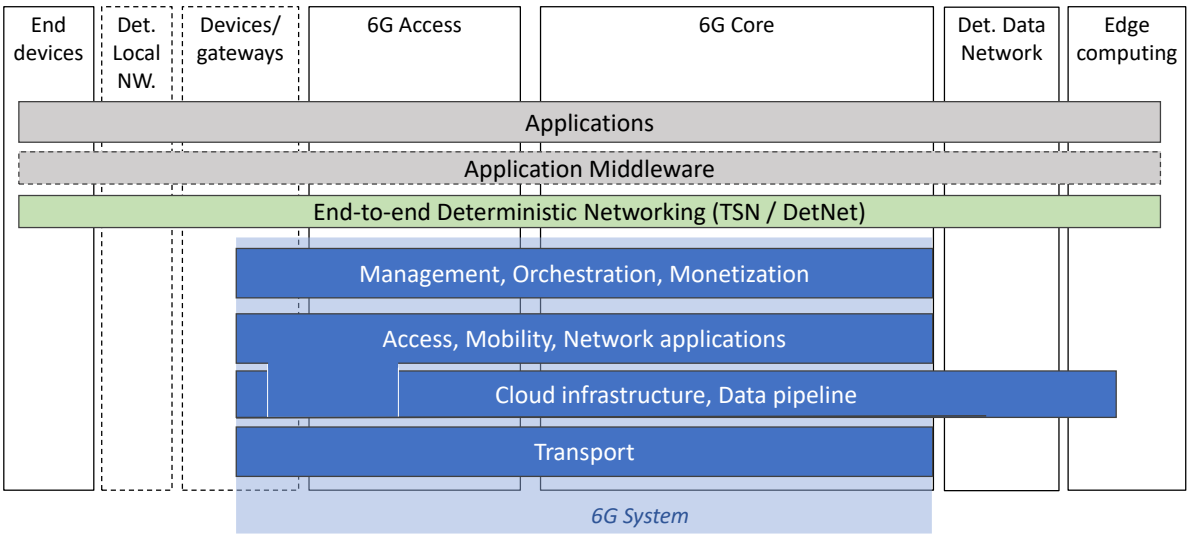

Figure 4.1: Draft DETERMINISTIC6G architecture (derived from [Eri23]).

Figure 4.2 shows how the concepts developed in DETERMINISTIC6G integrate into the E2E architecture with 6G. In the remainder of this chapter 4, we present more details of the architecture impact of the functionality for dependable networks developed in DETERMIISTIC6G in [DET23-D11], [DET24-D12], [DET25-D13], [DET23-D21], [DET23-D22], [DET25-D23], [DET25-D24], [DET23-D31], [DET23-D32],

[DET24-D33], [DET24-D34], [DET25-D35], [DET25-D36]. The numerals in Figure 4.2 indicate how these concepts influence the system architecture:

- 1: In [DET23-D11] we have described some future time-critical use cases and applications and analyzed their requirements in terms of KPIs and value creation in terms of key value indicators (KVIs). In the architecture the applications are operating in the application domain E2E, and may make use of some application middleware, such as OPC UA, as described in [DET25-D13] and section 4.5.4.
- 2: To invoke dependable communication for time-critical services, the applications need to provide their service specification, which means to specify their traffic characteristics and performance requirements, in order to request a dependable communication service from the network. In the network this is handled by the management, orchestration and monetization layer. By enhancing the information exchange and providing situational awareness between the application domain and the network domain, better service provisioning is envisaged [DET23-D11]. A dependable service design is described in [DET25-D13]. Network exposure provides an API between the network and application layer or application middleware, which allows to request dependable services from the network, and to obtain information about the provided network performance. The concept of application-communication-compute co-design has been introduced in [DET25-D13]. Critical applications, such as control applications, can be virtualized and hosted in a cloud environment and be connected via 6G to their application counterparts on the device. Under load and resource constraints, both the network and compute environment may be subject to performance variations, which may impact the application. At the same time, applications can support different levels and modes of operation [DET25-D13], which allows for some adaptivity from the application. By sharing insights from the operational state between the application, the network and the cloud environment the operational constraints of the compute, communication and application domains can be mutually aligned to the best value of the application [DET25-D13]. The management and orchestration layer of the network can also host a 6G network digital twin that can interact with a digital twin of the application domain and the CPS. The exchange of situational awareness between the network and application domains enables better planning in each of the domains by anticipating the operational characteristics of the other domain, see section 4.3.4 and [DET25-D36] [HSG+25].
- 3: The network needs to provide a dependable communication service. This means that it must be able to comply with and deliver the performance that is requested from the applications. To this end, the network needs to be able to monitor the KPIs that characterize the delivered service performance, see section 4.2 and [DET25-D23]. Furthermore, by data-driven (latency) performance prediction, the 6G network shall be able to specify which (latency) performance levels it can promise to what reliability level [DET23-D21] [DET23-D42] [DET25-D23]. One important characteristic is to be able to control also the packet delay variation as explained in [DET23-D31] [DET24-D34] [DET25-D35], for which mechanisms like packet delay corrections are proposed [DET23-D21] [DET25-D23]. These functions are part of the access and network applications in Figure 4.2 and build on the time awareness described below. The data-driven latency prediction further builds on the availability of a data pipeline for data collection and distribution to feed machine learning models, as described in [DET23-D21] [DET23-D41] [DET25-D23].

- 4: Dependable time-critical communication builds on time-awareness throughout the system. This is achieved by robust time synchronization which should also include hot standby support for time synchronization, see section 4.3.1 and [DET23-D22] [DET25-D24]. Time-awareness is provided in the transport and infrastructure layers, and may be used in the network function layer. To provide robust and secure 6G network services a paradigm of security by design shall be applied. To this end, data monitoring at the transport layer shall be possible, in combination with the data pipeline that allows for smart security assessment based on observed network behavior, as described in section 4.3.2 and [DET23-D32] [DET25-D24].
- 5: With the increasing interest to apply cloud compute capabilities, a cloudification of application / control functionality towards an edge cloud is of primary interest. This is in particular of interest for applications where functionality is offloaded to network-side (edge) compute capabilities to improve device performance. The integration of edge cloud with deterministic networking and providing dependable compute to the application domain ensure timely and effective integration of compute with time-sensitive communication in an E2E manner, see section 2.4 and [DET24-D12] [DET24-D33] [DET25-D36].
- 6: When considering latency variations of sub-components in an E2E system, gains can be provided for E2E deterministic networking by making the E2E traffic handling aware of the latency characteristics of sub-components [DET23-D31]. To this end, optimizations are proposed and evaluated, which are applied in the E2E deterministic network domain (i.e., TSN and DetNet). They provide more robust and optimized E2E deterministic network configurations that take the characteristics of the 6G network into consideration, see [DET24-D34] [DET25-D35] [DET25-D45] [EDV+25] [EGS+25] [DEH+25] [IEEEQee]. Such information is provided to the E2E deterministic networking controller (TSN or DetNet) from the 6G management layer and is based on network insights as described above in item 3.

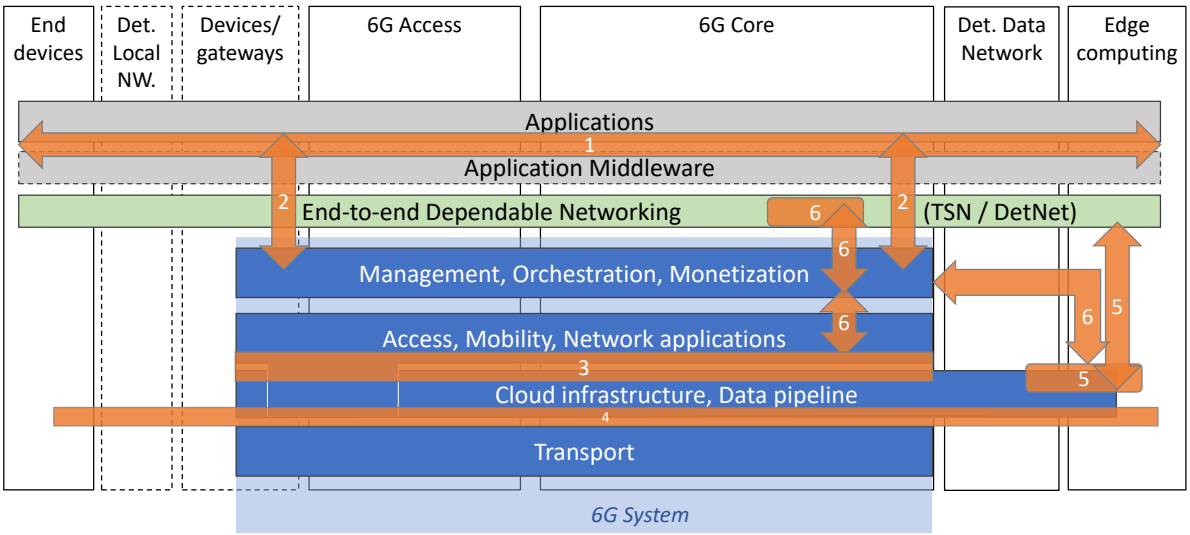

Figure 4.2. DETERMINISTIC6G enhancements in the E2E architecture.

4.2. Novel 6G capabilities for dependable connectivity

4.2.1. 6G performance monitoring

A dependable 6G network needs to be able to ascertain that it provides the required performance that has been agreed with the application. A pre-requisite for this is that the 6G network knows what performance it delivers to applications. To this end, there is a need for *performance monitoring* in-built into the 6G network. This study focuses on latency related parameters (delay, delay variations) as the primary performance metrics. This shall be complemented with functionality for *performance prediction*, which allows the 6G network to assess new service requests for acceptance [DET25-D13]. It also allows the network to re-negotiate with the application the required connectivity level in case that network conditions lead to an assessment that the network performance provided to application is at risk of not meeting the agreed performance levels. Applications which have some adaptivity and support multiple modes and / or levels of operations can in such a case still be sufficiently supported by the networks by adapting the level of operation [DET25-D13] [GSA+25]. In addition, in some cases it is required or beneficial, when the network can provide a deterministic performance level without major variations. In particular, packet delay variations may not be acceptable by some applications, or they make an E2E provisioning of dependable communication difficult, as described in section 4.5 and [DET23-D31] [DET24-D34] [EDV+25] [EGS+25].

The delays that a 6G network introduces for time-critical traffic flows include all delay components from when the packet enters the 6G networks to the moment when the packet leaves the 6G network. To quantify the 6G delay, the edge-to-edge delay of the 6G network (delay between ingress (UE or UPF) and egress (UPF or UE) of the 6G system) needs to be observed, as indicated in Figure 4.3. A break-down of delay components within the 6G network has been provided in [DET23-D21]. For a connection between an application in a mobile device and an edge server, four network segments contribute to the edge-to-edge delay (see Figure 4.3). The 6G CN processes the data frames and provides connectivity to the appropriate radio access network node; the connectivity between CN nodes and the RAN nodes is provided by a transport network. In the RAN, data transmission over the radio interface is performed between the base station and the user equipment. Radio protocols are responsible for reliable and efficient data transmission over the radio interface [DET23-D21] [PAD+25] (see Figure 4.4). In the UE device additional delay may be introduced that is not related to the radio transmission but depends rather on the device implementation architecture. Generally, the delay in the mobile network is not symmetric and can differ for uplink and downlink direction. For communication between a device and an application in an edge server, the 6G latency is introduced in both communication directions, from the device in the uplink and towards the device in downlink. For applications communicating between two devices, the communication from one device to the other comprises first an uplink delay from the device to the 6G CN, from where the data is further sent to the other device introducing a downlink delay.

For monitoring the packet delay in the network, the monitoring points need to be time-aware, which means that they can relate their observations to a common time reference. This requires time-synchronization between the observation points. A 6G RAN will work in a time-synchronized fashion so that all communications over the radio link are aligned to a radio frame structure to avoid interference. The base stations provide the reference timing of the frame structure. This can be used as time reference. For example, the number of OFDM symbols with regard to a certain frame reference provides a time scale that is shared between UEs and the base stations, as seen in Figure 4.4. In addition, the 6G network can be configured to support time synchronization towards applications and

external network nodes. In this case all traffic-handling nodes in the network, the UE, the gNB and the UPF are synchronized to the reference time of the RAN. With this setup of synchronized observation points, the packet delay of packets between two observation points can be measured by adding a time-stamp at the ingress node and determining the residence time at an egress node. Such measurements could be made edge-to-edge in the 6G network, or it could be made separately for the RAN and CN domains and then be combined for further analysis. As shown in [DET23-D21] the RAN dominates by large the delay characteristics in the mobile network.

It is important to note, that for the RAN delay there are multiple ways to determine delay measurements. One approach is to add some time stamp at a RAN ingress (i.e. transmitter) and determine the latency at the egress (i.e. receiver). This can be applied for both uplink and downlink: the delay measurement point for downlink is the UE, and the delay measurement point for uplink is the gNB. Alternatively, the measurement of delay can be integrated into the radio protocol operation. Radio protocols operate based on a radio frame structure that is time synchronized between UE and gNB. Furthermore, the radio protocols comprise support for reliable transmission (e.g. via HARQ) which includes reporting of successful data reception. Such radio protocol procedures can be enhanced with timing information so that the gNB knows the downlink delay until successful reception of a packet at the UE. More information can be found in [DET25-D23].

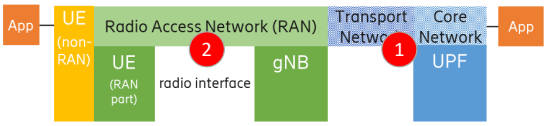

Figure 4.3: Distinct delay domains in the 5G/6G network: core and transport network (1) and radio access network (2) (ref. [DET24-D12]).

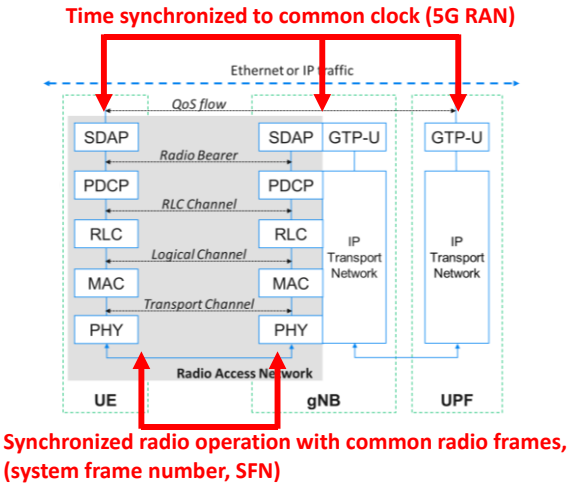

Figure 4.4: Time synchronization in 5G&6G. User-plane nodes sharing a common time reference.

4.2.2. Data-driven performance prediction

We propose a data-driven approach for delay performance prediction. To this end observations of latency that are measured in the life network (as described in section 4.2.1) are fed into a ML-

algorithm, which learns the anticipated delay probability of the network. This approach is shown in Figure 4.5. For dependable communication, it is of particular interest to understand the tail of the achievable performance, which indicates to what level the network can provide a certain guarantee in maintaining the network delay within a delay bound.

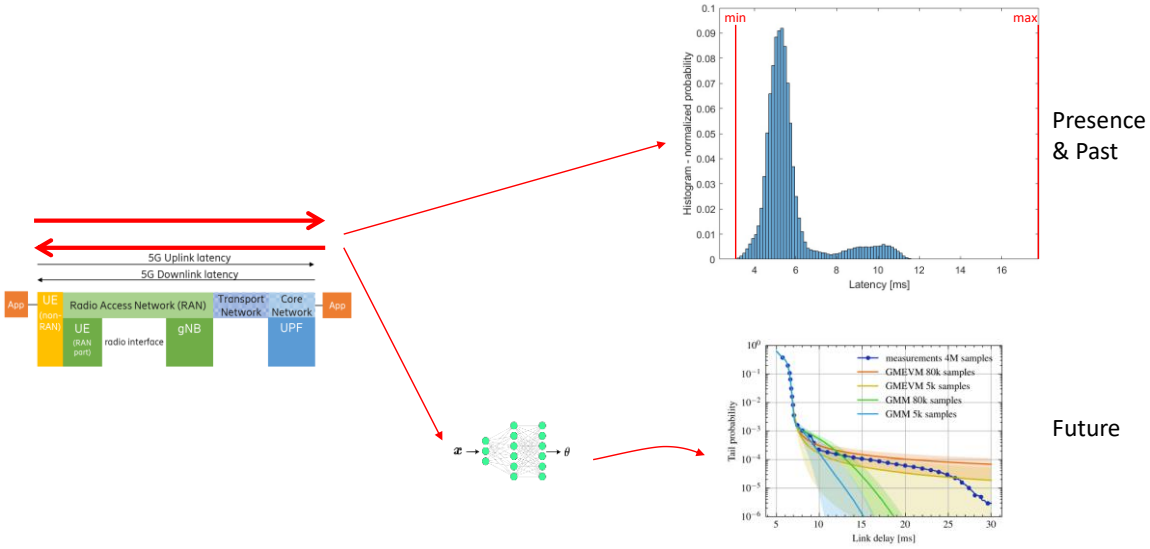

Figure 4.5: Data-driven delay analysis for dependable networks.

ML-based prediction is based on the following chain of activities:

- Performance data needs to be observed (see section 4.2.1),
- The data needs to be transferred to the ML-agent,
- The ML-agent is trained to predict the achievable performance.

The locations for performance data observations and of the ML-agent(s) determine how performance prediction can be integrated into the network architecture. If performance observations are distributed in the network, there is a distributed data collection. This enables fundamentally two approaches for realizing ML-based training, see Figure 4.6. With centralized learning (CL) a single ML agent is used for training, and all collected data needs to be transported to this ML agent. This causes the effort of data collection prior to the ML training. Alternatively, with distributed learning (DL) multiple ML agents are used for jointly training the ML system. Each DL agent locally trains the ML model based on its partial data set, and the trained model parameters are transferred to one ML agent where they are merged into a jointly trained model, which is re-distributed to all ML agents. In this case the collected data is consumed for model training locally, but an exchange of ML model parameters is needed in-between ML agents as part of the training.

These two approaches to training a performance prediction model need to be considered from the following trade-offs.

- How effective (i.e. precise) is the ML prediction?
- How much communication overhead is needed for exchange of data, either the monitored input data or the trained sub-models.

Figure 4.6 shows the options of centralized and federated learning, in a scenario where we consider that monitoring and data collection of downlink delay takes place in the UE, and for uplink in the gNB, as investigated in more detail in [DET25-D23]. In this case, either the collected data (for CL) or the ML model parameters (for DL) need to be exchanged over the radio interface and use 6G RAN capacity. It is not straightforward which option is most advantageous, in terms of ML training time versus overhead, as it depends on several parameters, like the choice of local epoch for federated learning or the available channel bandwidth, network load and channel qualities for the UEs [DET25-D23]. A clear benefit of centralized learning is, if the ML training is contextualized, which means that the network operational state is considered as context in the training of performance prediction. The information of the operation state is available in the gNB, like load and utilization of resources in a radio network area, or distribution of channel quality for the different UEs.

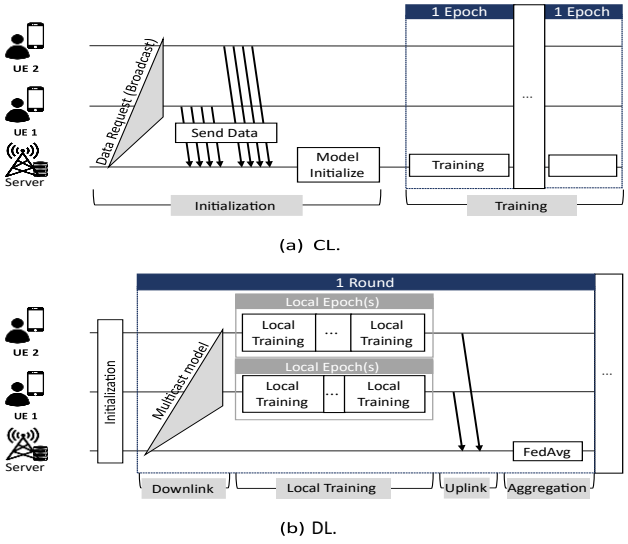

Figure 4.6: Learning procedures for the two learning schemes.

A proposed approach to apply delay performance prediction for the RAN is a combination of the following data collection and ML approaches, leading to a delay-aware RAN architecture [DET25-D23], as shown in Figure 4.7. Packet delay measurements for both uplink and downlink are collected at the gNB as described above and detailed in [DET25-D23]. This information is used by a ML-based delay predictor as described in [DET23-D21] [DET24-D42] [DET25-D23], which is located at the base station based on centralized learning. The ML training can be contextualized with the operation state of the RAN, which is available at the gNB. Examples of the operational state of the RAN are, for example, the traffic load, resource utilization, amount of traffic load with performance guarantees, radio channel characteristics for connected UEs. The obtained insights from delay performance prediction can be used to adapt the radio resource management to assure that the delay performance for the communication service is delivered according to the service requirements. While the service assurance is focused on the RAN only, this approach can be combined with the delay observability, prediction, and assurance in the CN as described.

Such a delay-assuring RAN architecture has the following benefits:

- It can work, even if (edge-to-edge) time synchronization is not used, as it operates solely on time-awareness with regards to RAN timing;

- It facilitates a centralized learning architecture (see [DET25-D23, section 4.1]),
  - which provides better learning performance compared to federated learning (assuming that the gNB has higher compute capabilities than the UEs),
  - which avoids the need for data collection transfer from the UE to the centralized learning agent, since the downlink latency can be directly observed at the gNB from the radio protocol operation. This reduces significantly the overhead for data collection for a centralized learning architecture.
  - It allows to enhance the ML-based delay prediction towards conditional delay prediction. The packet delay collected at the gNB can be associated with the contextualized operational state of the gNB. The predictor can derive delay predictions conditioned to the state of the RAN operation.

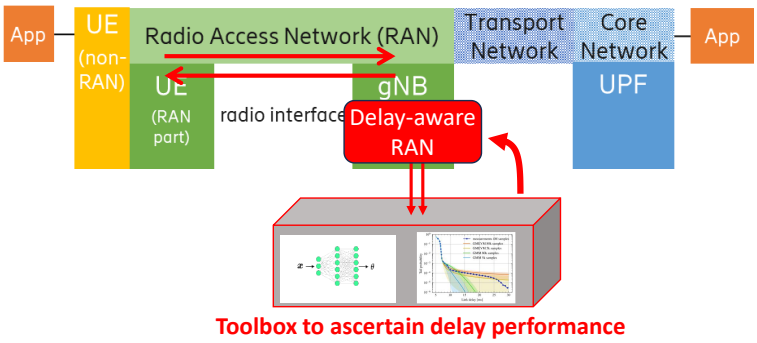

Figure 4.7: Design proposal for a delay-aware RAN architecture.

4.2.3. Packet delay correction for deterministic latency performance

As described in [DET23-D21], it is possible for 5G to provide low bounded latencies through the URLLC features, where more robust transmission modes reduce the required number of hybrid automatic repeat request retransmissions, and thereby the delay to achieve a certain reliability level. In this case, packet transmissions which perceive the worst transmission conditions are improved by boosting the reliability of their transmission. Such transmission conditions are not known beforehand, so all transmissions must be generally protected to a higher level. This implies a high resource cost. On the other hand, a bounded latency is not sufficient to provide time-critical dependable communications. The delay should also be stable, that is the packet delay variation (PDV) experienced should be very low[4]. It is difficult to guarantee such low packet delay variation only by controlling the radio transmission, which is inherently subject to stochastic variations. An alternative approach is to compensate for the incurred PDV at the edges of the 6G systems. We propose a packet delay correction (PDC) mechanism for such purpose, as illustrated in Figure 4.8.

PDC is a mechanism by which every packet is forced to remain in the 6G system approximately for the same amount of time. For example, if all the packets spend the same delay within the 6G system before it is forwarded to the next node, then the packet delay variation in 6G system is zero. This

[4]An alternative approach would be if the application would be robust to latency variations, e.g. if the application messages would contain timestamps a receiver could compensate for packet delay variations and apply a time-aware application logic. This could however not compensate for the sensitivity to packet delay variation of an intermediate network layer like TSN, see section 4.3.3.

mechanism is intended to be applied at the egress of the 6G system, e.g., at the DS-TT in downlink or at the NW-TT in uplink. The egress entity or the TT will hold the packet for some time, until it reaches the desired release time, e.g., maxDelay, and then it is forwarded to the next node in the external network, e.g., the TSN network. Note that to decide how long the packet should wait, additional information is required which is transported within every packet as metadata. For example, if a timestamp-based PDC is applied, then a timestamp metadata is carried in every packet which is added at the ingress of the 6G system. Then the egress TT can calculate the time that a packet already spent within the 6G system and from this it can derive the remaining time to reach maxDelay. This is illustrated in Figure 4.9. A broader description of PDC can be found in [DET23-D21].

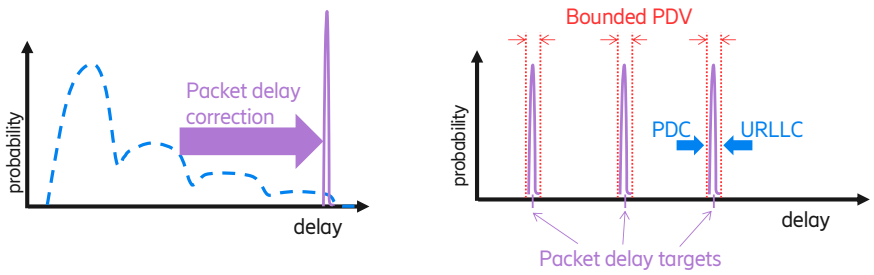

Figure 4.8: Packet delay correction (PDC) to remove packet delay variation. The combination of URLLC and PDC can be used to define lower and upper bounds of the perceived delay enabling deterministic 6G latency performance, as shown on the right.

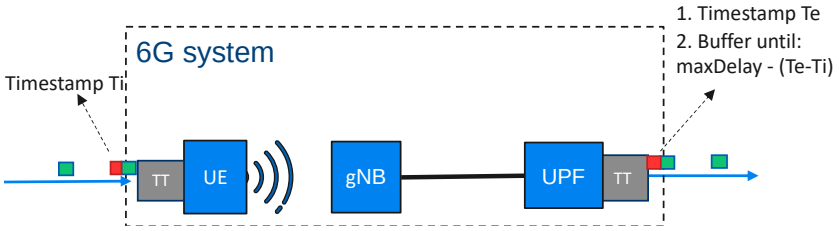

Figure 4.9. Timestamp-based PDC

PDC basically compensates (or corrects) the packet delay variation, which is basically introduced by stochastic variations in different parts of the 6G system. Since the main component of PDV intrinsically comes from the radio transmission, PDC could also be applied in the RAN segment of the 6G network (i.e. between gNB and UE in Figure 4.9) and be an integrated component of the delay-aware RAN described in section 4.2.2.

4.3. Integration of 6G with TSN and DetNet

4.3.1. Time synchronization

A time synchronization service is available in 5G since Release 16, and other improvements have been added in Releases 17 and 18 [GLR+20] [MAG+19] [PDR+21] [3GPP18-23501]. However, time synchronization reliability remains an important issue. Indeed, there is a need to support resilient time synchronization mechanisms in time critical 6G-TSN applications in order to meet high levels of availability for the time synchronization service. The current TSN time synchronization (generic Precision Time Protocol (gPTP), IEEE 802.1AS) relies on the Best timeTransmitter Clock Algorithm (BTCA) to find the next grandmaster (GM) clock, when the current GM clock has failed or degraded in performance. However, BTCA may take time to find the next GM. Additionally, BTCA is unable to

detect transient faults in a GM, hence could lead to a ping-pong effect between two potential GMs. For the emerging 6G-TSN use cases, this waiting time without a synchronization source is not desirable. With the objective to provide a continuous precise timing, a new amendment of the IEEE 802.1AS standard is ongoing (802.1ASdm) which modifies the hot standby mechanism. In the new setting, the hot standby mechanism is modified, where BTCA is not used to select the next GM, and instead a static configuration is sent down from the network management system (e.g., the CNC in the case of TSN). Note that this static configuration is possible today in 5G via the setting of IEEE 802.1AS dataset information[5] that is received from the CNC, these datasets are then used for the configuration port states within the 5G system in order to follow the GM's time distribution hierarchy. Having configured two GMs, primary and hot standby, the hot standby GM is always transmitting timing messages along with a primary GM. In order to keep the two GMs in sync, the hot standby GM synchronizes itself to the primary GM before it starts sending timing messages. Hence each end station has access to two time domains at any given time. If there is a failure or performance degradation in the primary GM, the hot standby GM takes over immediately.

From an architecture point of view [DET23-D22] analyses the different implications on the location options of the GM, the redundancy design, and the 3GPP support for such a scenario. Indeed, the location of the primary and hot standby GM will determine the coverage of the synchronization redundancy. Hence, a careful design to optimize the location of such GM clocks is required. More importantly is the aspect of 3GPP support of the 802.1ASdm amendment, and consequences based on whether the 6G GM becomes the primary GM or the hot standby GM.

3GPP support for 802.1ASdm has not been specified since this standard amendment is still under development. However, currently 5G has the capability to support it given the fact that datasets for external static configuration are available in the standard as long as the 5G GM is not set as the primary or hot standby GM. Therefore, the analysis in such cases is considered here for a future 3GPP release such as for 6G. There are different possibilities for how the 6G GM behaves in a hot standby enabled 6G-TSN network. In case the 6G GM is neither the primary nor the hot standby GM, then it is enough that the 6G system maintains the primary and secondary time domains as independent time domains. This is already possible with the available 3GPP 5G support for multiple time domains. If the 6G GM becomes the hot standby GM, this would not be acceptable from a 3GPP viewpoint since the 6G GM would need to synchronize with the external primary GM. Such an option is not feasible because the 6G GM must be guaranteed for the operation of the base stations and the general 6G network availability, and it is also not supported in the current 3GPP standards. If the 6G GM becomes the primary GM for the external system (e.g., TSN network), this would be possible as the 3GPP standards already support this case where 5G GM is the external GM. More work is required for the upcoming 6G to investigate the support for the hot standby amendment and the above description forms the ground for such study.

[5] IEEE 802.1AS includes some global variables that are part of the management datasets sent from CNC to the 5G system control plane, most relevant *defaultDS.externalPortConfigurationEnabled* and *externalPortConfigurationPortDS.desiredState*. When the variable defaultDS.externalPortConfigurationEnabled is set to "true", then static external configuration is enabled instead of BTCA. In this case, the port states are configured accordingly per time domain (i.e., per GM) using the variable externalPortConfigurationPortDS.desiredState. These variables are included in the management information containers between 5G control plane and the NW-TT and DS-TTs (see 3GPP TS 23.501, Annex K.1).

Some open questions are discussed in [DET23-D22] to enable a hot standby GM in 6G-TSN networks. In particular, the placement of these GMs can introduce dependencies that may affect synchronization performance, as highlighted in prior research [S20] [SDL+21]. When designing the time synchronization architectures to incorporate hot standby GM within the scope of standardized time synchronization mechanisms as mentioned in [DET23-D22], several architectural and operational aspects should be carefully evaluated. Detailed discussions are provided in [DET23-D22] and [DET25-D24], here we provide a summary of those discussions.

- Designating the 6G GM as one of the Grandmasters offers the advantage of leveraging the high-precision 6G clock, which typically has access to GNSS-based time. If the primary GM (i.e., the 6G GM) fails or loses its GNSS connection, the associated gNB can maintain time synchronization temporarily by entering a holdover mode.
- The 6G clock plays a central role in ensuring accurate time synchronization across key 6G components such as the gNB, UPF, and UEs, which is essential for their reliable operation. For this reason, it is generally unsuitable for the 6G GM to rely on synchronization from an external, non-6G time source, as it could compromise the internal timing integrity. Consequently, the 6G GM is not an ideal candidate to serve as a hot standby.
- In scenarios where the 6G GM acts as one of the GMs, it is recommended to disable the optional split functionality (i.e., set it to FALSE). This prevents the synchronized GM from attempting to re-synchronize the 6G GM once it or the 6G connection recovers from a failure, thereby avoiding potential timing inconsistencies within the 6G network.
- Additionally, placing the GM on the network-side of 6G helps to minimize timing errors at TSN end-devices on the network-side, since synchronization messages only need to cross the wireless interface once, reducing latency and variability.

Given the consideration above, the following time synchronization architectures have been proposed and analyzed in [DET23-D22] and [DET25-D24], see Figure 4.10 and Figure 4.11.

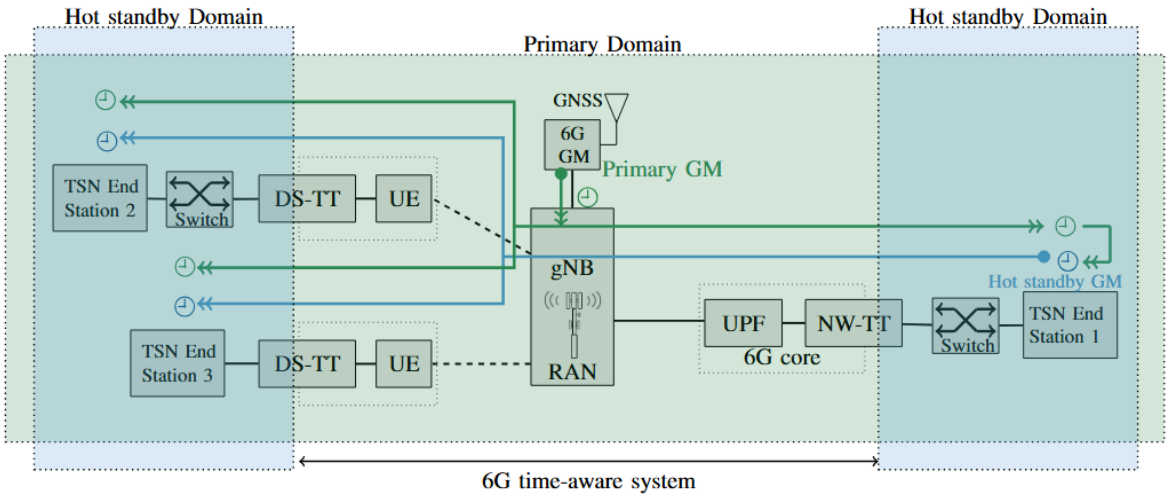

Figure 4.10: 6G as primary GM and network-side hot standby GM.

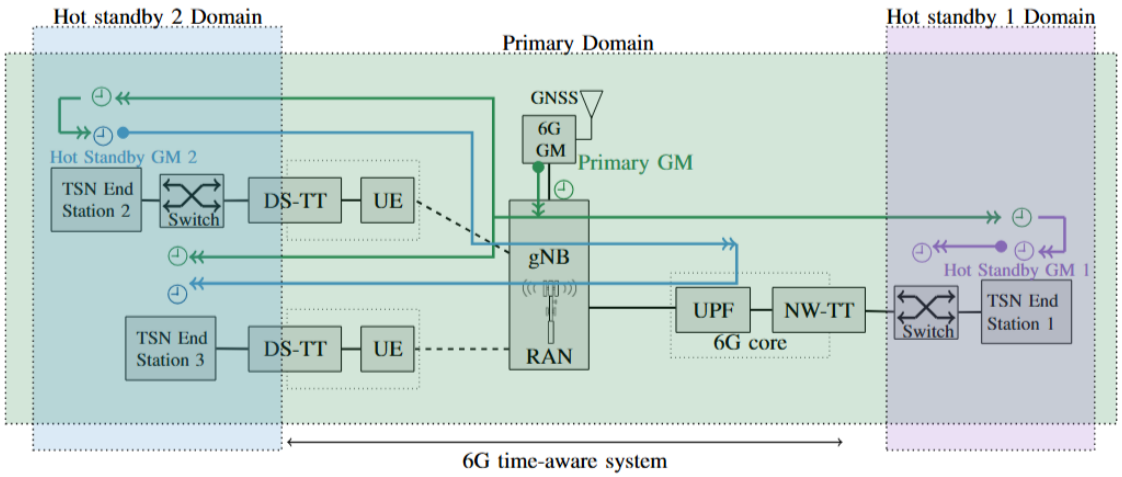

Figure 4.11: 6G GM as primary GM and one hot standby GM on device-side TSN end station and another hot standby GM on network-side TSN end station.

Scenarios that utilize a hot standby GM generally offer better performance than BTCA approaches in terms of minimizing clock drift and reducing periods when devices are out of sync during failures. However, our evaluation in [Det25-D24] shows that in certain failure situations, using a static GM setup with a hot standby can result in portions of the network being left without a functioning GM. In contrast, BTCA can automatically elect a new GM in response to failures, though this process introduces some delay.

Therefore, the choice between a static GM configuration with a hot standby and a dynamic configuration using BTCA should be guided by the specific network architecture and the availability of additional redundancy mechanisms, such as redundant communication links. For example, in networks with extensive link redundancy, a static GM setup with a hot standby may offer superior performance. On the other hand, in topologies with limited redundancy, a dynamic GM selection mechanism like BTCA could enhance fault tolerance by adapting more effectively to failures.

4.3.2. Security by design

4.3.2.1. Prior Security Architectures for 6G networks

This sub-section provides a summary of prior work related to security for 6G networks presented in [DET24-D12, section 3.2.2].

Looking at the complete picture, security-by-design integrates security into the system architecture including software assurance processes, trusted hardware and execution environments, threat analysis, continuous monitoring, countermeasures, audits, and rigorous testing. For 6G networks, security is embedded at the infrastructure level, employing E2E and defence-in-depth strategies alongside new privacy and control mechanisms. This approach is critical for dependable networks, Industrial Internet of Things (IIoT), and Industry 4.0, where dependable communication is essential. A layered architecture integrating 5G/6G, TSN, and DetNet ensures E2E dependable networking, with security spanning all layers to protect data integrity, confidentiality, and availability. Key security functions include encryption, authentication, anomaly detection, and other threat mitigations; in this work we focus particularly on time-sensitive services.

Furthermore, E2E encryption between paired protocol entities is vital for preventing data breaches but must balance security with resource availability, actual risks and communication quality, especially for latency-sensitive applications. Cryptographic methods like Advanced Encryption Standard (AES), Elliptic Curve Cryptography (ECC), and lightweight cryptographic solutions (e.g., NIST standards) are used for resource-constrained IoT devices. Threat detection employs AI/ML for real-time monitoring and adaptive responses. Security Service Level Agreements (SSLAs) define protection requirements across network domains, ensuring risk-cost-performance trade-offs.

Looking at the overall picture, to facilitate the management of complex networks involving massive number of devices, zero-touch security management (ZSM) is needed to automate risk analysis and threat response using AI-driven data collection and analytics. For this, security probes need to monitor traffic, while orchestrators need to enforce mitigation actions. Precision monitoring and traffic steering enhance resilience, while other techniques like trusted execution environments, network slicing, SDN/network functions virtualization (NFV), and redundancy can also play a role to further secure dependable networks. Frequency spectrum management mitigates jamming attacks, particularly in high frequency 5G/6G bands. Additional considerations include zero-trust networking (ZTN), MEC, adversarial ML robustness, data privacy techniques (e.g., federated learning), and secure transactions via distributed ledgers.

The DETERMINISTIC6G project has defined a security-by-design architecture for 6G but limited the scope to issues related to low-latency and dependable networking requirements, and how they impact the monitoring and security management. For this, in-band network telemetry (INT) for real-time security analytics and low-latency threat response was investigated and a prototype developed. The objective was to demonstrate how dependable communication for critical services can be assured. This work is further summarised in the next section 4.3.2.2. Special attention was given to threats targeting the Precision Time Protocol (PTP) and latency, ensuring timely detection and mitigation. This is further described in section 4.3.2.5.

4.3.2.2. Security Architecture and Enablers for 6G Dependable Networks

This section provides a summary of the work carried out in the DETERMINISTIC6G project related to the enablers for 6G Deterministic Networking presented in [DET23-D32, section 3.4].

6G networks require a robust security architecture that responds to emerging cyber threats; but for time-critical applications, rigorous performance and predictability is also required. A security-by-design approach is fundamental, embedding protective measures at every layer of the network while maintaining ultra-low latency, high reliability and predictability. The techniques used need to be adaptable to the requirements coming from applications but also to changes in the environment. This architecture integrates artificial intelligence and machine learning for analytics, real-time telemetry for monitoring, and programmable network functions to enable dynamic threat response without overly compromising network performance.

At the core of this architecture is a high-level framework structured into Security Management Domains, each responsible for distinct segments of the network, including the RAN, Edge, and Core. These domains operate independently but collaborate through an **Integration Fabric** (depicted in Figure 4.12) to enforce consistent security policies across the entire network. Within each domain,

specialized components work together to ensure comprehensive protection. **Security Data Collectors** gather critical performance metrics, traffic metadata, and threat intelligence, feeding this information into a **Security Analytics Engine** that uses AI and ML to detect anomalies such as distributed denial-of-service attacks or unexpected latency variations, and uses techniques such as similarity learning to find the root causes of the anomalies. The **Decision Engine** evaluates potential responses, balancing immediate reactive measures with longer-term strategic adaptations, while the **Security Orchestrator** dynamically deploys virtualised security functions such as intrusion detection and prevention systems to mitigate threats in real time, or implements other techniques using, for instance moving target defence (MTD) and slicing.

These functional components operate in a closed-loop manner (i.e., collection-detection-reaction as depicted in the steps 1 to 5 of Figure 4.12), enabling AI-driven software-defined security orchestration and management in accordance with expected SSLA and regulatory requirements.

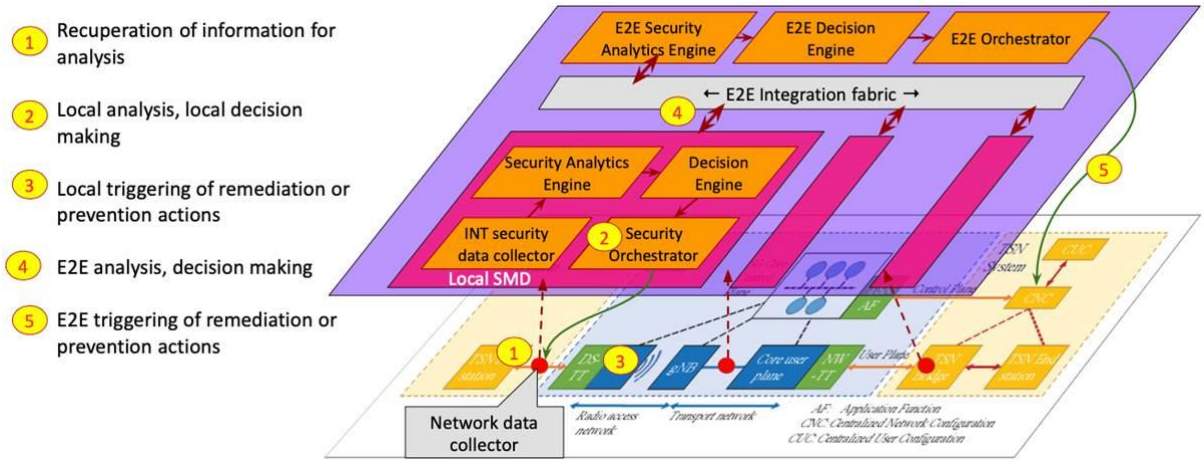

Figure 4.12: High level architecture of E2E security monitoring & management framework.

A dedicated End-to-End Security Management Domain oversees security for services that span multiple network segments, ensuring that policies are uniformly applied and threats are addressed cohesively. Key enablers for this architecture include **INT**, which provides real-time, high-precision monitoring of network traffic without introducing high additional overhead, and **programmable data planes** (using P4 language or other techniques such as NetFPGA, DPDK, etc.) that allow for customised traffic handling and in-network security enforcement. These technologies are critical for maintaining dependable performance while detecting and mitigating attacks both in the control and data planes.

AI also plays a central role in threat detection and response. The **Security Analytics Engine** employs machine learning models trained on both historical and real-time data to identify deviations from normal behaviour, predict potential attack vectors using external threat intelligence feeds, and correlate events across different network domains for comprehensive root cause analysis (RCA). This enables the system to detect sophisticated threats such as PTP attacks in time-sensitive networks or latency disruption attempts in time-critical applications.

When a threat is detected, the **Security Orchestrator** dynamically enforces countermeasures by reconfiguring SDN and NFV components to isolate affected segments, deploying virtual security

functions where needed, and employing MTD techniques to disrupt adversarial activity. This automated, adaptive approach ensures rapid response to threats while minimizing human intervention.

Two illustrative scenarios demonstrate the architecture's effectiveness. In the first scenario, an adversary floods a time-sensitive network with malicious PTP messages, disrupting clock synchronization critical for time-critical applications, such as industrial automation (this is further detailed in section 4.3.2.5). The system detects the abnormal traffic patterns using AI-driven analytics and responds by rate-limiting suspicious requests while validating legitimate sources. For long-term protection, programmable switches are configured to filter malicious packets directly in the data plane.

The second scenario involves an attacker injecting high-priority rogue traffic to destabilize a dependable network supporting robotic control systems. As in the previous scenario, real-time telemetry identifies latency violations, triggering network slicing to reroute critical traffic and dynamic QoS adjustments to restore normal operation.

Despite these advancements, several challenges remain. Balancing security measures with strict latency requirements is an ongoing concern, particularly for applications where encryption and monitoring could introduce unacceptable delays. Establishing trust and consistent security policies across multi-vendor, multi-domain networks also present complexities that require innovative solutions in federated security management. These are discussed in the following two sections (sections 4.3.2.3 and 4.3.2.4).

4.3.2.3. User and Operator Intents Translated to Security Controls

Security policies must dynamically align with both user expectations and operator requirements. Unlike traditional networks with static security rules, 6G could or should adopt an intent-driven approach called intent-based networking (IBN) or intent-based management, as already mentioned in section 3.2. Concerning network security, the objective is to translate high-level requirements into automated security controls. Users may prioritize seamless connectivity, privacy preservation, and guaranteed service levels, while operators focus on threat prevention, regulatory compliance, and infrastructure resilience. The user or operator-defined intents are mapped to granular security mechanisms through AI-driven policy engines and programmable network functions.

For users, intent-based security [OLK+24] can manifest itself, for instance, as: 1) adaptive authentication where behavioural patterns are used in addition to rigid credentials; 2) privacy requirements that translate into differential data handling, with sensitive information having stronger encryption or localized processing at the edge, while non-critical data undergoes lightweight protection to conserve resources; and 3) service-level intents, such as time-bounded communication, that trigger automated security optimizations, selective encryption to meet latency thresholds.

On the other hand, operators can leverage intents to enforce network-wide security [CCG+22]. For instance, a "zero-trust" intent deploys continuous device attestation, while a "regulatory compliance" intent generates audits and applies local data governance rules. Threat-response intents activate real-time countermeasures, e.g., detecting a distributed denial-of-service (DDoS) attack might dynamically

reroute traffic for further analysis, whereas an anomaly in industrial IoT traffic could isolate compromised devices via programmable data planes.

Translating end-user Intents to actionable security controls can, for instance, rely on [OLK+24]:

- Natural language processing to convert textual intents into machine-readable and deployable policies;
- AI-based orchestration to select and deploy optimal security controls (e.g., choosing between P4-based filtering or NetFPGA);
- Continued monitoring to ensure that the controls align with the intents.

IBN is a network architecture that leverages automation and machine learning to help better align network behaviour with business objectives. This is illustrated by the following scenario:

Title	Deterministic Security for Industrial Control Systems in 6G-Enabled Manufacturing
Introduction	Industrial Control Systems (ICS) often require security guarantees and deterministic performance to ensure real-time operation. IBN allows defining security policies that are dynamically aligned with precise timing and reliability requirements, transforming high-level intents into enforceable, time-bounded security controls.
Example intent	Protect ICS from unauthorized access while ensuring uninterrupted, low-latency control processes.
Translation	The intent is translated into a technical implementation that prioritises deterministic behaviour. Network slicing is used to create a dedicated network segment for ICS traffic, configured with reserved bandwidth and Quality of Service (QoS) parameters. The slice isolates safety-critical communications, such as robotic control signals or sensor feedback, from other network traffic. Security measures are configured so that they do not introduce unpredictable delays.
Implementation	Time-Sensitive Networking (TSN) protocols should be used to enforce deterministic scheduling, for synchronizing devices, and prioritizing time-critical data flows. Access control mechanisms are implemented with hardware-based authentication, with each endpoint device cryptographically verified before participating in the time sensitive application. Behavioural monitoring uses machine learning models trained on deterministic datasets so that anomalies can be detected and mitigated within given time limits by sampling traffic packets.
Mitigation strategy	When security incidents occur, response actions are executed within strict latency boundaries by dropping or diverting suspicious traffic using the adapted programmable data plane techniques.

4.3.2.4. Security Across Multi-domain Networks

The following description illustrates the interactions between the different components of the TSN, 6G mobile network and MEC domains to define and deploy security and performance user-defined intents, detect any anomaly (e.g., in the latency), identify the cause, and mitigate it (e.g., following a MTD strategy), which is illustrated in Figure 4.13. Note that the numbers in Figure 4.13 correspond to the numbered actions described in the following paragraph.

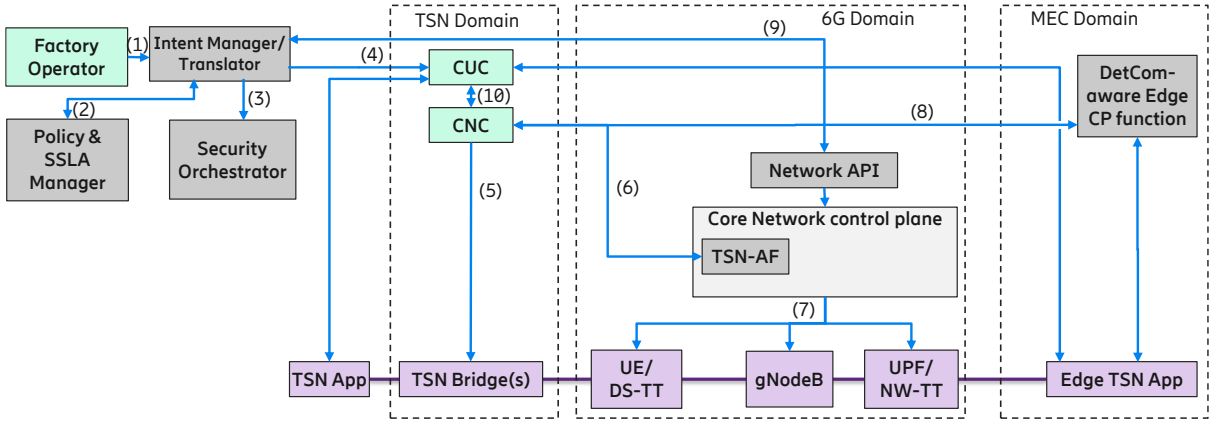

Figure 4.13: Intent-based TSN, 6G and MEC connectivity and security control.

The creation of the connectivity sequence begins with (1) the Factory Operator (a machine or a human) submitting a latency-sensitive intent (e.g., "Ensure 1ms latency for TSN application (TSN App in Figure 4.13) and the Edge TSN App" to the Intent Translator. Note that both the TSN App and Edge TSN App have by then been registered as talker and listener, respectively, to the CUC. The Intent Translator is a component introduced here that is responsible for converting high-level service requirements (e.g., "1ms latency") into technical configurations across TSN, 6G, and MEC domains. The Intent Translator operates together with a Security Orchestrator and Policy and SSLA Manager, which all have an end-to-end scope, that comprises in the example of Figure 4.13 the TSN, the 6G and the MEC domains. If OPC UA is used as application middleware, they could be related to the OPC UA connection manager, see section 4.5.4 and [DET25-D23]. The Intent manager (2) validates the intent through the Policy and SSLA Manager then (3) triggers the Security Orchestrator to activate INT and deploy security rules. After approval, (4) the Intent Translator can initiate a TSN configuration (on behalf of the talker) by directing the CUC to reserve TSN resources via the CNC using IEEE 802.1Qdj, with (5) the CNC then programming the TSN Bridge components (this includes the TSN bridges, talker, and listener) with flow rules as specified in IEEE 802.1Q. Similarly, the CNC configures the flow rules of the 6G network (as virtual TSN bridge) with (6) via the TSN-AF, which translates the CNC configuration accordingly into a 6G user plane session setup (7), according to section 3.1.2. Then with (8) the configuration of the application and the cloud deployment is initiated by invoking a TSN-aware cloud management component (related details can be found in section 4.4.1 and section 4.4.2) to ensure that the deployment complies with the TSN configuration and meets the real-time requirements (e.g., ensuring 1ms latency in SLA). Specific security related intents could be provided to the 6G mobile network via a network API (9). Upon successful deployment, a confirmation of (5), (6) and (8), is provided from the CNC to the CUC (10) via status groups, the fulfilment status could propagate back through CUC to the Intent Manager/Translator to notify the Factory Operator of completion, creating an E2E configured service chain from TSN source to MEC application with guaranteed latency and embedded security monitoring. The traffic management of end-to-end dependable communication would then follow the procedures described in section 4.5, making use of the 6G network support for TSN and DetNet as described in section 3.1.2.

The security monitoring sequence begins with the INT Collector obtaining real-time telemetry by adding timestamp data to the header of network packets to detect changes in the latency (as will be

explained in section 4.3.2.5). This is done to some packets only to reduce the overhead introduced. It is also possible to gather data from the TSN Bridge streaming IEEE 802.1Qbv scheduling statistics, gNB reporting RAN latency via 3GPP network resource manager (NRM), UPF providing N4 interface metrics, and MEC App X emitting application-level latency data through ETSI MEC 012. The Security Analytics engine then correlates these multi-domain measurements, detecting a critical latency violation (e.g., exceeding 1ms) determining its cause using RCA techniques (e.g., DoS replay attack), and informing the Decision Engine. Upon identifying a threat, the Decision Engine triggers MTD countermeasures through the Security Orchestrator, which coordinates cross-domain mitigation by interacting with domain-specific Security Orchestrators. The closed-loop verification confirms remediation success when fresh telemetry shows restored 1ms latency, prompting Security Analytics to notify the Factory Operator of the recovery, completing the autonomous detect-analyse-mitigate-verify cycle for time-sensitive 6G-TSN services.

Conclusion

To achieve E2E security-by-design for dependable 6G-TSN networks, novel interactions are desired across domains. The intent translator and security orchestrator (SO) need to provide E2E functionality and interact with different domains. For this, in the future API-based interfaces need to be specified so that the intent translator and SO can be authenticated/authorized, subscribe to domains specific performance reporting (e.g., from RAN, Core Network, MEC domains). The INT should be controlled from the application domain where the intents are defined. If a domain specific SO detects a security threat it needs to inform the E2E SO, such that it can manage the response that might imply several domains. Some of the elements involved could need the introduction of new or modified interfaces for implementing the required security policies defined by the intents and the policy managers. Most importantly, but this is for future research and development, the security architecture must enforce Zero Trust through mutual authentication, implement latency-aware encryption for TSN traffic, enable unified telemetry via INT, and enable closed-loop automation between analytics, decision, and orchestration components. Such approaches will enable dependable, secure communication while maintaining interoperability between the TSN, 6G and MEC domains.

Furthermore, to ensure security coverage across all domains, E2E analytics, decision, and orchestration engines could be integrated through a unified E2E Integration Fabric, as presented in section 4.3.2.2. This holistic approach would enable coordinated threat detection, automated response, and consistent policy enforcement throughout the entire 6G-TSN-MEC ecosystem.

4.3.2.5. Security Architecture for Assuring Time Synchronization

A comprehensive overview of time synchronization threats in [DET23-D22] highlights that delay-based attacks can only be effectively detected through continuous monitoring of timing behavior. This section presents an implementation of the proposed security-by-design approach aimed at enhancing resiliency of PTP E2E time synchronization by enabling the monitoring, detection, and localization of time-delay attacks (TDAs). Unlike several PTP vulnerabilities that can be mitigated by the use of cryptography methods to prevent packet content manipulation attacks, a TDA does not modify a PTP packet but only delays it, for example, an attacker intercepts a PTP packet and holds it for a given duration before relaying it to its destination. This delay distorts propagation delay measurements, leading to synchronization errors of all downstream clocks. Additional details regarding the nature of

TDAs and their detection can be found in [DET25-D24], while the system's implementation and evaluation are documented in [DET25-D44] and [DET25-D45], respectively.

The proposed approach was evaluated using a Mininet-based emulation environment, as illustrated in Figure 4.14, that replicates a realistic deployment scenario. In this setup, PTP-based synchronization is augmented with P4-programmable transparent clocks (TCs). It is worth noting that when PTP messages traverse a 5G/6G system, the system effectively functions as logical TC. To simulate TDAs, the 5GDetCom delay emulator was used to introduce asymmetric delays at a compromised TC. The TC omits updating the correctionField, resulting in misleading propagation delay measurements. This accurately reflects a realistic attack where only one direction of communication is affected.

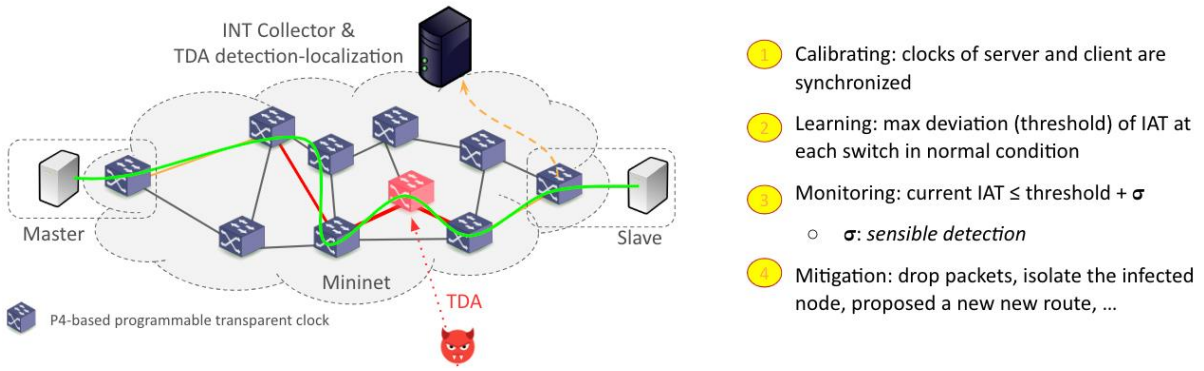

Figure 4.14: Mininet-based emulator of TDA & detection in E2E time synchronization.

It is worth noting that when PTP messages traverse a 5G system, the system effectively functions as logical TC. As can be seen in Figure 4.15, the 5G system consists of a CN with a UPF, and a RAN including gNodeB and UEs. The DS-TT, appended to the UE, and the NW-TT, appended to the UPF, are the entities in charge of handling the PTP messages in the 5G data plane. Calculating the 5G residence time and inserting it in a PTP message is part of the DS-TT's and NW-TT's responsibilities.

TDA can unintentionally occur in a TC when the correctionField does not reflect correctly the residence time of its message. For example, in the time synchronization over a 5G system, as shown in Figure 4.15, a vital prerequisite for correctly calculating the residence time in this logical TC is that the $t_{ingress}$ and t_{egress} are timestamped by a single reference clock. In other words, it requires that the clocks of the DS-TT and the NW-TT are perfectly synchronized. Otherwise, a TDA can occur in such a logical TC.

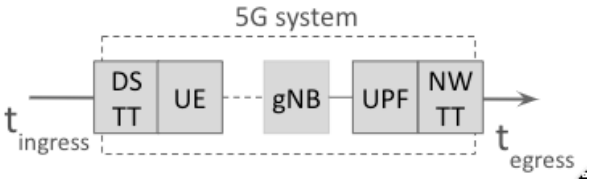

Figure 4.15: Time synchronization over a 5G system acting as a logical PTP transparent clock.

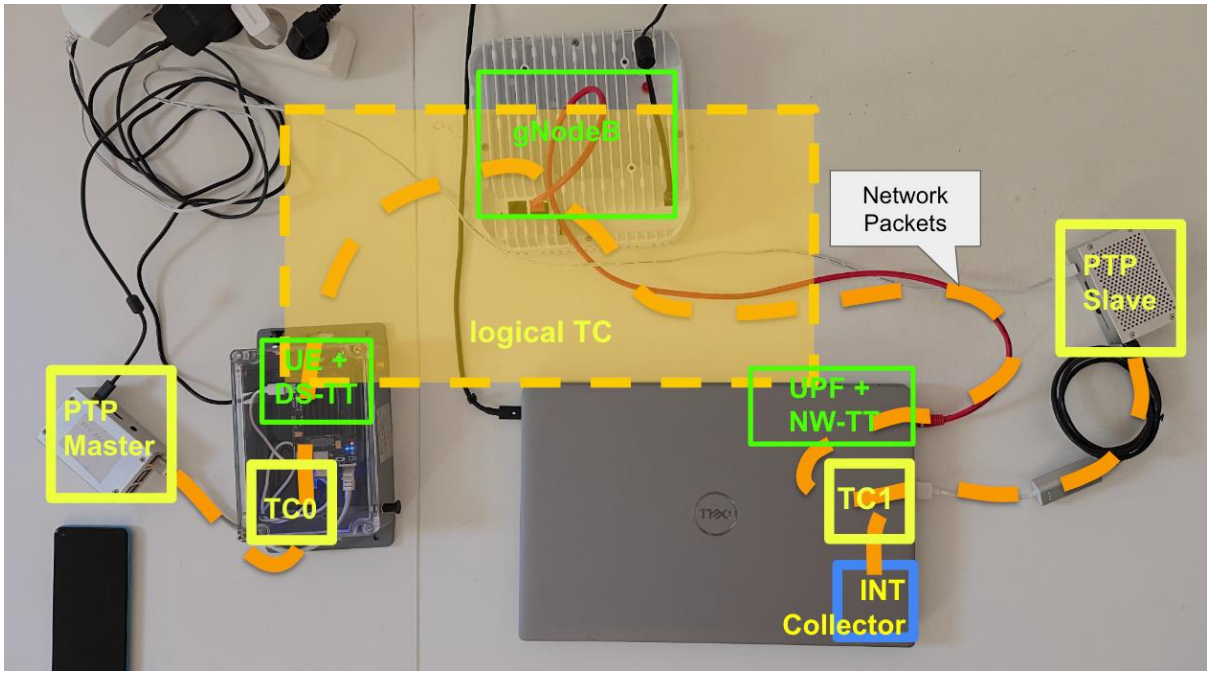

Figure 4.16: Secure PTP time synchronization over a 5G network.

Figure 4.16 illustrates our physical testbed used to simulate TDAs and evaluate detection mechanisms targeting PTP synchronization over a 5G network. The setup consists of three TCs, one of which is a logical TC emulating the 5G system. A TDA scenario is triggered when there is a time misalignment between the UE and the User Plane Function (UPF), disrupting accurate timestamp synchronization. It should be highlighted that the Mininet-based emulation approach, which utilizes software timestamping, eliminates the need for specialized hardware and allows for the flexible emulation of multiple clocks with ease.

The emulation incorporates a comprehensive pipeline for traffic collection, feature extraction, and anomaly detection. Key technologies leveraged in the framework include INT, P4-based data plane programmability, and high-precision telemetry. The detection mechanism continuously monitors clock offsets and inter-arrival time variations, enabling the identification of abnormal timing patterns that are often invisible to traditional security tools.

To support data collection and analysis, an INT collector is positioned near the client to capture traffic at the TC that is just before the client. This collector extracts telemetry data embedded within PTP extension fields and forwards it to Grafana for real-time visualization, facilitating both manual inspection and automated anomaly detection.

Correctness and Compatibility of P4-based Programmable Transparent Clocks

The experimental results demonstrate that the proposed solution effectively maintains synchronization accuracy. The P4-based TCs exhibited performance on par with conventional LinuxPTP-based TCs. During the initial synchronization phase, clock offset values were elevated, as expected, but they stabilized rapidly. Once calibration was complete, the offsets consistently remained below 150 microseconds, aligning with the accuracy typically achieved using Linux software timestamping. These results confirm that the use of programmable TCs does not compromise

synchronization quality and that they are fully suitable for deployment in time-sensitive networking environments.

INT Integration

The integration of INT into PTP messages proved highly effective for monitoring synchronization behavior across the network. By embedding telemetry data directly within existing PTP extension fields, the solution enables real-time visibility into packet timing characteristics without generating additional overhead. The collected telemetry allows for tracking of propagation delays across multiple hops, offering valuable insights into network behavior and highlighting irregularities that may impact synchronization performance.

TDA Detection and Localization

Most importantly, the validation confirmed the system's ability to detect and localize TDAs. By monitoring variations in timing between messages, it was possible to pinpoint the presence and location of delay anomalies. These insights were derived without requiring perfect clock synchronization between devices, demonstrating the robustness and practicality of the approach.

Conclusion

This evaluation validates the feasibility and effectiveness of a security-by-design framework for protecting PTP-based time synchronization in dependable networks. Through the integration of INT telemetry, P4-based programmable TCs, and continuous monitoring, the system provides a reliable means of detecting subtle delay-based attacks that are often undetectable by conventional mechanisms. The approach supports real-time, packet-level analysis and lays a strong foundation for future extensions involving automated response and mitigation, offering a promising path forward for resilient, software-defined, time-sensitive network infrastructures.

As future work we also need to consider the case where an INT data collector could be compromised. To be able to detect this and prevent it from affecting the correct detection of delay-based attacks, several observation points need to be enabled so that any misbehaving one can be detected and ignored.

4.3.3. 6G support for end-to-end traffic management with TSN

4.3.3.1. End-to-end Scheduling in TSN with 6G

E2E network paths may consist of wireline and wireless links, which have fundamentally different properties with respect to packet delay (PD) and PDV. Both, PD and PDV, are orders of magnitude greater for wireless links compared to wireline links (milli-seconds vs. micro-seconds). To cope with these characteristic delay properties, we have presented algorithms to calculate robust time-driven TSN E2E schedules as typically used for scheduled traffic in IEEE 802.1Qbv that provide provable probabilistic guarantees while ensuring efficient utilization of network resources. The algorithmic details can be found in previous deliverables and papers [DET24-D34] [EDV+25] [EGS+25].

From an architecture point of view, we can build on the standard interfaces and logically centralized architecture (fully centralized model) defined for TSN, since the 6G system presents itself as a logical TSN bridge. Therefore, the required extensions to the architecture are (intentionally) small, and can be summarized as follows:

(1) Latency monitoring and prediction: Our wireless-aware algorithms to calculate robust E2E schedules need to be aware of the probabilistic port-to-port delay of the logical TSN bridge. Moreover, to facilitate proactive ("make before break") adaptations of E2E schedules, predictions of the port-to-port delay are essential. Therefore, components for monitoring, analysis, and prediction of port-to-port delay are essential, as described in section 4.2 in this report.
(2) Extended wireless-aware TSN control plane: The control plane interface between the (logical) TSN bridge and the CNC must be able to communicate detailed information about the port-to-port delay. To this end, we have proposed control plane extensions in [DET23-D31] [DEH+25] [IEEEQee] supporting detailed delay histograms for port-delay (instead of the former min/max bounds), and utilizing event-based mechanisms of NETCONF to proactively trigger schedule adaptation.
(3) The schedule planning component in the CNC shall execute the algorithms for calculating wireless-aware E2E schedules using the input about port-to-port delay distributions and predictions. Schedule planning is a CNC internal function and does not have any architecture implications, however it needs to obtain relevant information through the architecture, like e.g. the distribution of packet delay values.

4.3.3.2. Coordinated traffic management in 6G and TSN

From an E2E perspective, traffic management is happening on different levels. The TSN CNC configures traffic management in each TSN bridge on the E2E path, including the virtual 6G bridge. In addition, the 6GS manages radio resource allocation to the E2E data streams. In particular, for periodic time critical communication as in cyclic control operations, resource allocations can be adapted to the periodicity of the traffic arrival. In E2E TSN scheduling this is achieved in the time-aware shaper by configuring the periodic schedules of each bridge (via the *gate control list*) in accordance to the traffic periodicity. Similarly, the 6G RAN can be configured for a periodic allocation of transmission opportunities. However, the periodic allocation of resources in the RAN is generally independent in timing from the application itself; only the periodicity can be matched to the application periodicity. If there is an offset between traffic arriving from the application and the instances of pre-allocated transmission opportunities, this leads to a systematic queuing delay of packets prior to radio transmission. In [DET25-D23] [HHA+24] a mechanism has been proposed, where the offset of packet arrivals and periodic resource allocations is determined in the RAN. If this information is provided back

to the application, the application transmitter (and the TSN scheduler) can adjust the transmission arrivals at the RAN with periodic RAN resource allocations by an appropriate offset. To this end, it may be beneficial to enable feedback from the RAN, via the TSN control nodes (CNC and CUC) to the application transmitter. Such a feedback flow (as indicated in Figure 4.17) may provide benefits. Similarly, such an interface may also provide the opportunity of co-optimizing the TSN and RAN resource allocation, which is identified as potential area for performance optimization [DET25-D23].

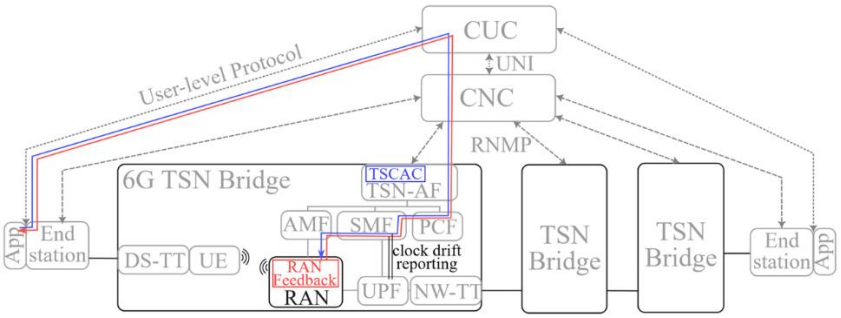

Figure 4.17: 6G-TSN integration with the flow of traffic pattern information (source [DET25-D23] [HHA+24]).

4.3.4. Digital Twin

A digital twin is a virtual representation of a physical system that is synchronized to the physical system at a certain frequency and fidelity [DET24-D33] [DET25-D36] [HSG+25]. The digital twin has knowledge about the operational state and the characteristics of the physical system. It can further apply a model of the system and predict the characteristics of the system. This can be used in a what-if analysis for different system configurations to optimize the system operation. Digital twins exist for different physical system, as indicated in Figure 4.18. A 6G Network Digital Twin may maintain a digital representation of the 6G network. Similarly a CPS digital twin can maintain a digital representation of a cyber-physical system, like a smart farming area or an industrial plant [DET23-D11] [DET24-D12]. The CPS DT can be used for the planning of application tasks in the CPS. Similarly the 6G NDT can be used to recommend network configurations. Each digital twin can create a situational understanding of its corresponding system. The mobile network may know in which area it can provide which capacity (based on network deployment), and it knows the load in the network over time and location. It can also estimate which connectivity performance and which availability levels it can provide over time and location, e.g. based on performance monitoring and prediction as described in section 4.2. The CPS knows what assets are engaged at what time and what location in which tasks. It can derive what kind of traffic and with what performance requirements needs to be supported over time and location.

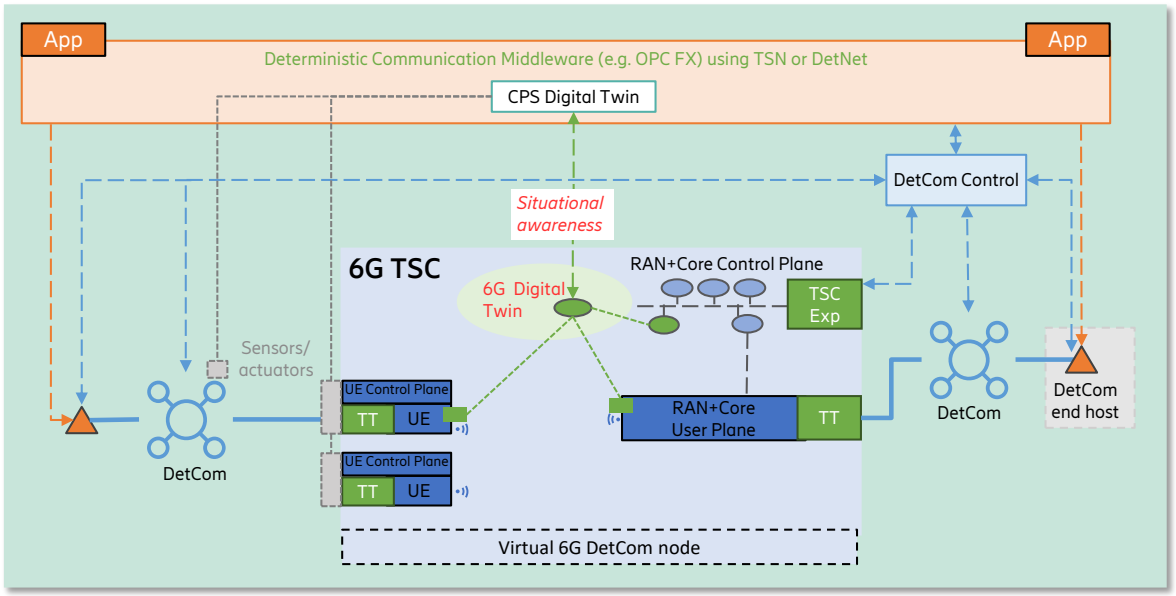

Figure 4.18: Digital Twin.

A distributed system with assets engaged in critical interactions uses and depends on the availability of an underlying network with sufficient performance, see Figure 4.19. The network itself depends in its performance and characteristics on the usage of the network, the traffic load, the location of connected assets, etc. The network and the CPS plan their future actions/configurations. The CPS may plan the sequence of activities performed by different CPS assets. The network may plan the service assurance of ongoing connectivity services and anticipate future connectivity services. In their respective planning, each system can benefit from situational awareness of the other system. As discussed in [DET24-D33] [DET25-D36] [HSG+25], a 6G network can plan its service delivery into the future, if it knows in advance when and where, what kind of new traffic load appears in the network and what performance levels need to be provided to these applications. Similarly, if an CPS task planner knows constraints or limitations of a network over time and location it can plan its activities accordingly. It can decide if a cloud-controlled mobile robot shall move from point A to point B via one trajectory or another trajectory, loading the network at different locations and with different application tasks. The exchange of situational information allows for new ways to optimize the system operation. From an architecture perspective, this situational awareness of the network or the CPS is created based on data that is collected from the network or the CPS respectively. By proper reasoning a situational understanding can be achieved. To allow cross system optimization across, a suitable exchange of situational knowledge across the system boundaries is desired. For example, some new network exposure API could be defined that allows to exchange situational knowledge between the network and an application system.

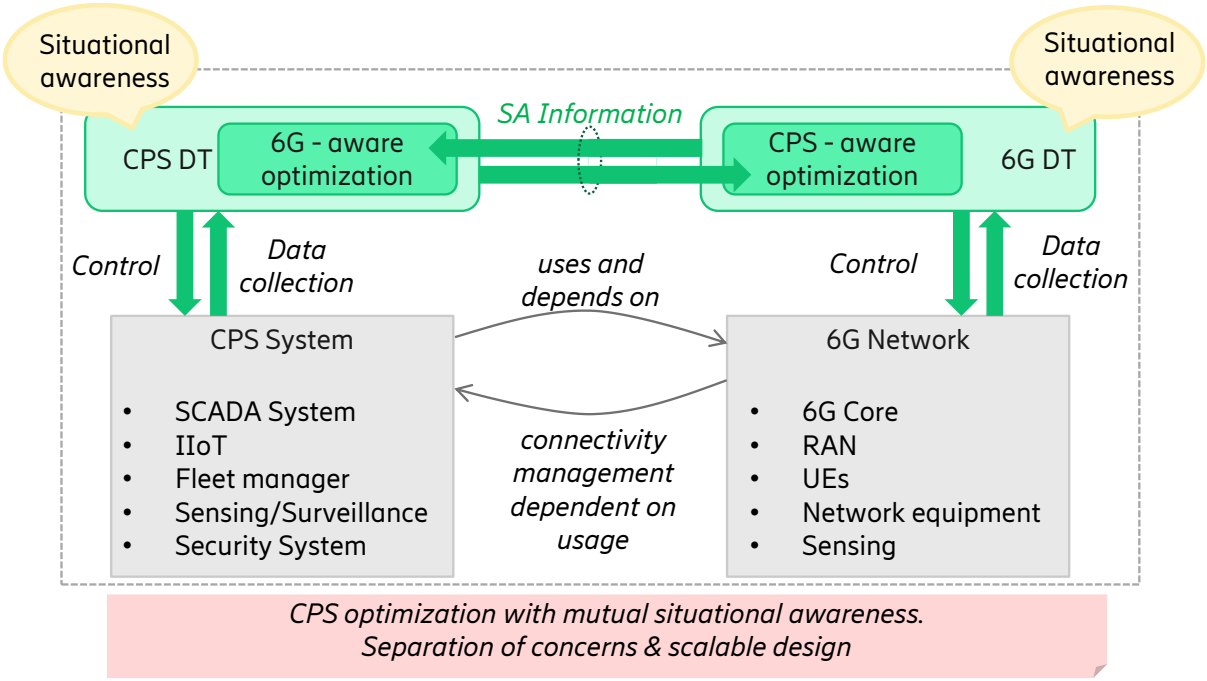

Figure 4.19: Interaction between the CPS DT and 6G DT.

4.4. Edge computing for time critical applications

In [DET24-D12] two basic deployment scenarios are identified and discussed for ensuring the integration of 3GPP-defined TSN support and edge computing. In one of the options the edge computing service is provided by the mobile network operator, and the hosted, cloudified applications (aka TSN Talkers/Listeners) are connected to a 5G/6G virtual TSN bridge that includes all the TSN network functions defined by 3GPP. In this scenario edge computing is tightly integrated with the 3GPP system, and this approach enables the full utilization of the edge computing support features specified by 3GPP SA6 (details can be found in TS23.558).

In the other option the edge computing deployment is isolated from the mobile infrastructure, and a single standalone data center is used to host the cloudified applications. This scenario enables a high degree of flexibility to tune the edge infrastructure and deployment tailored to support time-critical applications.

In addition, in [DET24-D33] and [DET25-D36] various traffic handling schemes in the host virtualized networking are proposed to ensure the seamless support of 802.1Qbv traffic scheduling for the cloudified applications.

1. 802.1Qbv-aware traffic handling in the virtualized network using a coordinated time-gating scheme for the containers' interfaces: This solution applies the TAPRIO queuing discipline on the virtual Ethernet interfaces of the containers, configured in such a way that at any given time, only one containerized application is allowed to send traffic towards the host's physical NIC.
2. The other proposed option is a centralized traffic handling scheme using the Open vSwitch (OVS) in the Kubernetes container network interface (CNI) plugin. The 802.1Qbv-aware time

gating mechanism is realized on the egress interface of the OVS, which is connected to the host's physical NIC. To ensure proper timing, a hierarchical scheduling concept is proposed, consisting of the combination of *a)* priority queuing or Cyclic-Queuing and Forwarding (CQF) for proper packet ordering, and *b)* TAPRIO qdisc for the timely packet forwarding.

3. The essence of the third solution is to use eBPF to propagate the packet timing information, so in this case no direct access to the host NIC or modification of the application is required. A *TSN proxy* is implemented as a secondary CNI plugin. Before the packet leaves the pod's network namespace, the *TSN proxy* stores the timing metadata and when the packet reaches the NIC, the *TSN proxy* restores the metadata required for proper TSN scheduling of the packet.

The details of methods 1) and 2) can be found in [DET24-D33], while the details of method 3) are provided in [DET25-D36].

4.4.1. Control plane integration support of edge computing and TSN domains

The above-mentioned methods are crucial tools for ensuring seamless user-plane integration between a legacy TSN communication domain and the compute domain (e.g. host with a cloudified TSN endpoint). However, they are not enough, as control-plane integration is also necessary to

- explore the details of the cloud host execution environment towards the TSN control plane in a standardized way, and
- handle the 802.1Qbv scheduling information provided by the TSN control plane (CNC/CUC) within the cloud ecosystem and, based on this configure the cloud deployment specific traffic handling scheme (e.g., the above-mentioned traffic handling methods require specific and distinct configurations).

To address this problem, an abstraction of the cloud host for the TSN control plane is proposed, which can be used to describe the characteristics of arbitrary cloud host deployments according to 802.1Qcc and 802.1Qdj standards. The core idea is to represent the cloud host and the deployment of cloudified application using a combination of virtual TSN endpoints and virtual TSN bridges as illustrated in Figure 4.20 for bare metal, VM-based and container-based deployment.

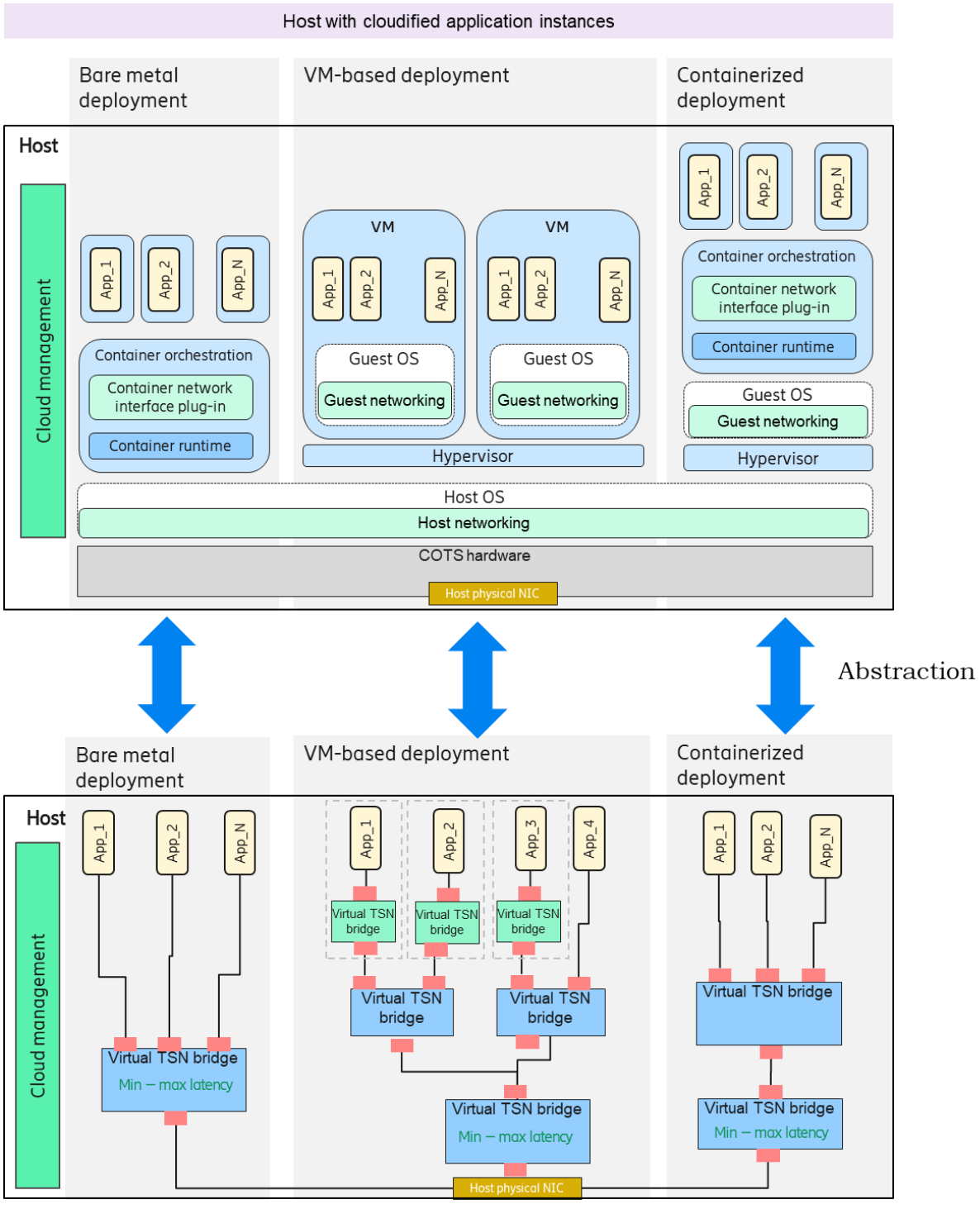

Figure 4.20: Abstraction of cloud host with various virtualization deployment options

The model consists of App instances as virtual TSN endpoints, representing the cloudified application instances in a one-to-one mapping manner, and virtual TSN bridges. The virtual TSN endpoints and their connected links to the virtual TSN bridges are used to represent the application scheduling capabilities of the (underlying) cloud deployment, such as how the CPU resources are reserved for a certain application instance for task execution - meaning when a certain application can start to forward a packet towards the NIC. The latency of the link between a virtual TSN endpoint and a virtual

TSN bridge can be interpreted as the *Transmit Offset* parameter for a given application instance according to the IEEE 802.1Qdj. If any uncertainty in the application scheduling should be considered, then the *Earliest Transmit Offset* and the *Latest Transmit Offset* could also be reported towards the CUC. Furthermore, if a more detailed description of the application scheduling is required, the cloudified application instance can be represented by a combination of a virtual TSN endpoint and a virtual TSN bridge (referred to as the green bridge in Figure 4.20) and the *min* and *max bridge delay* can represent further uncertainties in the scheduling. The virtual bridges (blue bridge entities in Figure 4.20) represent the characteristics of the virtualized networking of the cloud host. According to the various virtualization options, the networking on different virtualization levels (e.g. in the guest OS, host OS) is represented by separate virtual bridges, which form a tree structure, originating from a virtual bridge that is directly connected to the NIC.

From an architectural point of view, a *TSN-aware cloud management component* – illustrated in Figure 4.21 – is also proposed for the host.

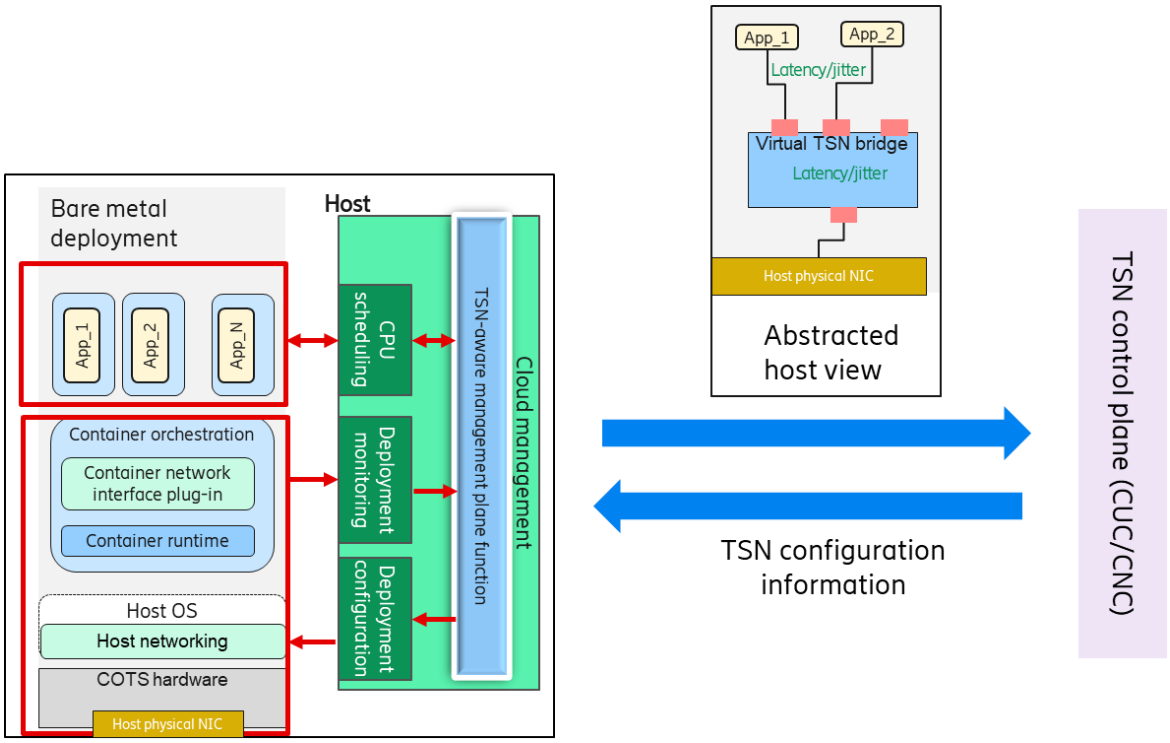

Figure 4.21: TSN-aware cloud management component

The cloud management component is responsible for exploring the cloud host and collecting the capabilities of the current deployment pertaining to a certain application, which requires a TSN communication service. Information about the CPU scheduling (e.g., type of scheduling) and the virtualized networking capabilities is collected. Based on this information, the *TSN-aware cloud management component* constructs and parameterizes the TSN-compatible, abstracted view of the host deployment. This abstracted view can be explored by the TSN control plane (CUC and CNC entities) using the standard NETCONF protocol, according to IEEE 802.1Qcc and IEEE 802.1Qdj. The *TSN-aware cloud management component* receives the 802.1Qbv scheduling plan from the TSN control plane, which is valid for the virtual TSN endpoints and virtual bridges. The application timing parameters (e.g., *Transmit offset)* are then translated into CPU scheduling configuration on the host

and together with the 802.1Qbv scheduling information, the cloud-deployment-specific traffic handling scheme is configured.

4.4.2. Architectural aspects of operator-enabled edge computing support

In the operator-enabled scenario the 6G network operator owns or operates the edge cloud domain as well, enabling a tighter integration of this domain, see Figure 4.22. The CUC communicates with the application components in the device and the cloud, while the CNC communicates with the DetCom AF (practically the 3GPP TSN AF) in the mobile network and with a corresponding component representing the cloud domain. To enable integration to the TSN control plane, the purple-colored components are needed in the cloud domain: the DetCom functions in the EAS and the DetCom-aware Control Plane functions as highlighted below and detailed in [DET25-D36].

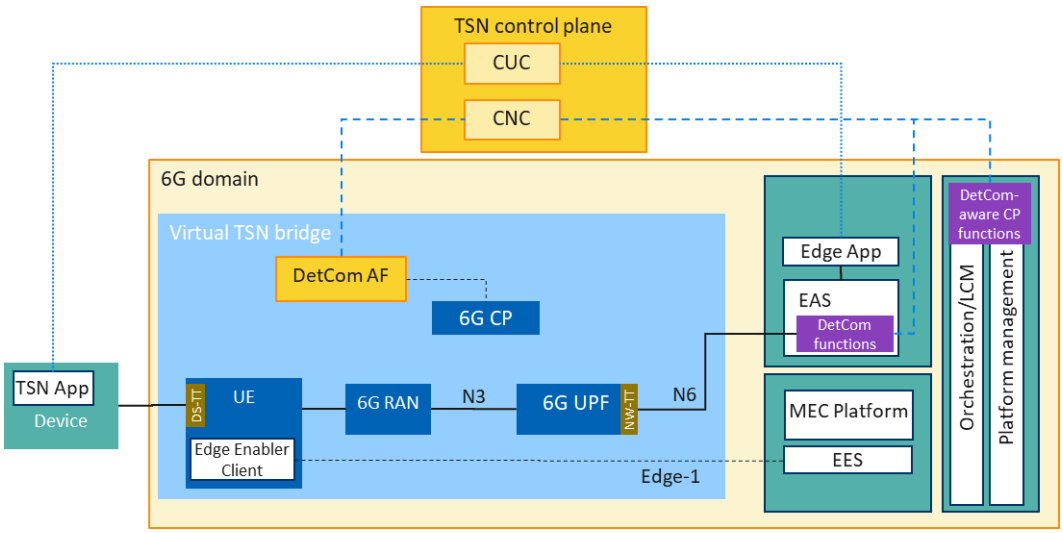

Figure 4.22: Operator-enabled Edge.

For a TSN App on the device to utilize functionality provided by another TSN App deployed in the cloud, specific steps are needed as detailed in section 2.3.4 of [DET25-D36]: application deployment, service initiation, edge capability exposure and service configuration. The edge capability exposure and service configuration highly depends on how the edge control plane is connected to the TSN control plane. We highlight two architecture options for this.

According to the first option, there is a direct interface between the Edge platform and the CUC/CNC. In this setup the Edge domain is connected to the TSN control plane like in the case of the standalone Edge (described in Sec 2.3 of [DET24-D33]), see Figure 4.23. The interface between the TSN control plane and the Edge domain is similar to the interface between the TSN control plane and the 6G domain. The domain is represented as a single or as a set of TSN Bridges and endpoints. Furthermore, the interface could be extended with (proprietary) components to better expose Edge capabilities.

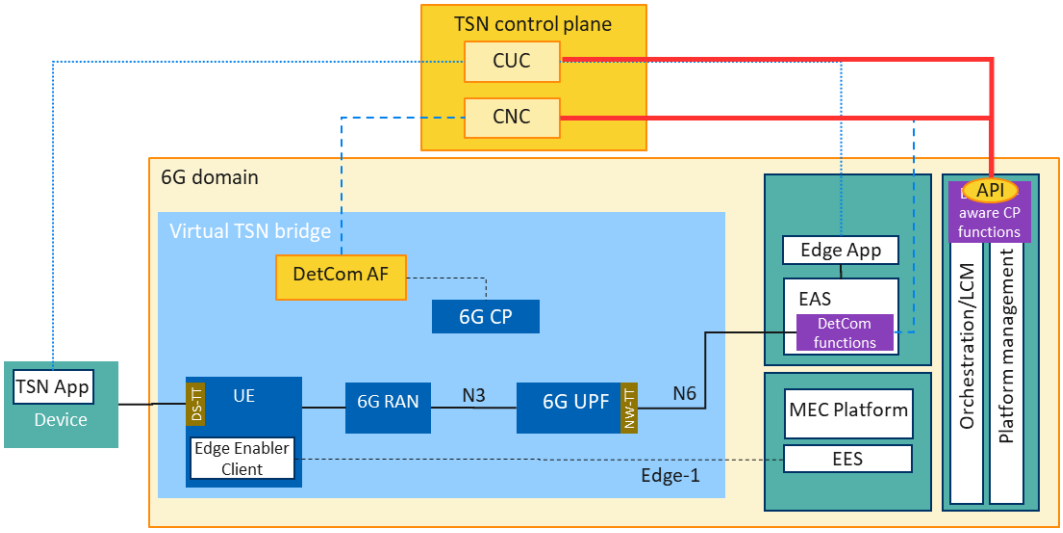

Figure 4.23: Direct Edge CP interface

The main advantage of this direct interface option is that no 3GPP standard is required, as the CNC/CUC communicates directly with the Edge domain without involving Mobile domain control plane entities.

According to the second option, the SEAL architecture is used. In the Service Enabler Architecture Layer (SEAL) [3GPP18-23434] based architecture the Edge domain control plane is connected to a SEAL NRM component. This component also acts as the TSN-AF for the 6G domain, see Figure 4.24. On one hand, the SEAL NRM is the TSN-AF, representing the 6G domain to the CNC/CUC, like in a non-Edge enabled scenario described in [DET24-D12]. In this case, the SEAL NRM configures the 6G domain via the N5 (to the PCF - Policy Control Function) or N33 (to the NEF - Network Exposure Function) interfaces. On the other hand, the SEAL NRM is responsible for the Edge capability exposure and configuration, interacting with the TSN CNC/CUC also on behalf of the Edge domain. To support this operation, the Edge-2 and Edge-7 interfaces could be used to exchange information between Edge deployment and the SEAL NRM.

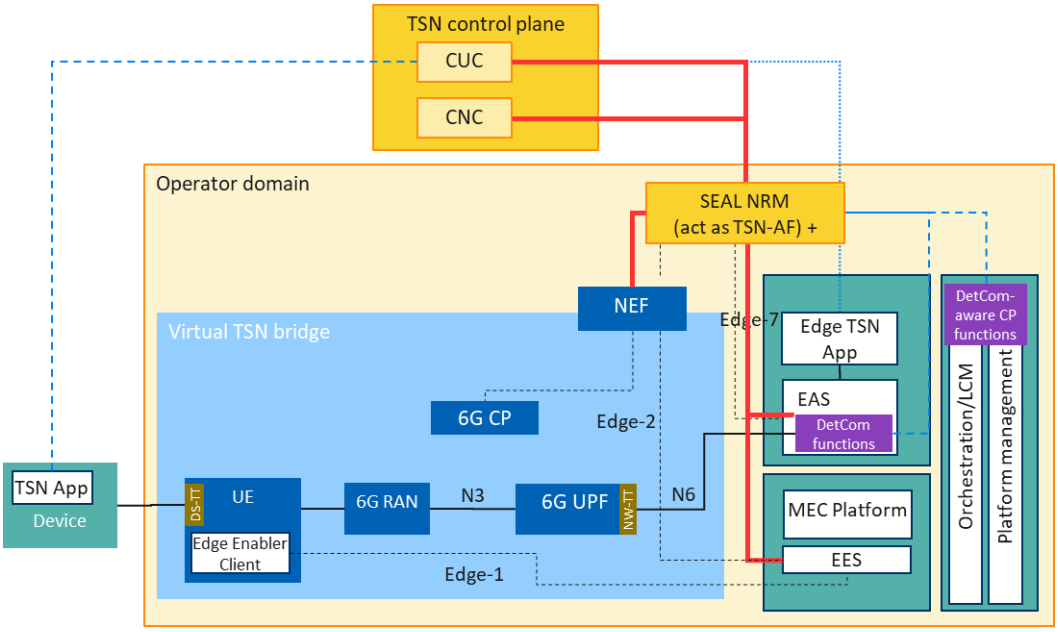

Figure 4.24: Seal-based Edge CP interface

The main advantage of this solution is that both the 6G and the Edge domain are represented by a single entity towards the TSN control plane. This enables various abstraction and optimization options. While the SEAL NRM increases in complexity, it does not affect the existing TSN-AF functionality.

4.5. End-to-end dependable communication with 6G

4.5.1. Architectural Aspects of Wireless-Aware E2E Traffic Management

End-to-end traffic management is concerned with providing dependable communication between applications over a network including wireline and wireless links. There are two major aspects of E2E traffic management considered in this report:

- **Wireless-aware** E2E traffic management: As already described in section 4.3.3.1, wireless-aware E2E traffic management enables dependable communication over E2E network paths with wireline and wireless links. The essential components of the architecture have already been described in section 4.3.3.1 and can be summarized as follows: (1) latency monitoring and prediction providing latency information to wireless-aware algorithms for planning E2E schedules; (2) Extended wireless-aware TSN control plane to communicate latency information with the CNC and proactively trigger the execution of algorithms to adapt wireless-aware E2E schedules. (3) Schedule planning component of the CNC to calculate wireless-aware E2E schedules.
- **Multi-domain** E2E traffic management: End-to-end communication might span multiple network domains. Partitioning the system into multiple domains might be beneficial for various reasons as motivated in more detail below. From an E2E traffic management perspective it is important to define an architecture that supports traffic management across domains and that allows for the integration of the wireless-aware E2E scheduling approaches mentioned before. Such architectural aspects of multi-domain E2E traffic management are described in more detail in the following section 4.5.2.

4.5.2. Multi-Domain E2E Architecture

There are various reasons for multi-domain systems:

- *Autonomous administration and management*: each network domain can be administered and managed autonomously. For instance, a vendor of a larger machine (e.g. a paper printing machine or a harvester in a smart farming use case) might provide this machine with an integrated local network. To guarantee the correct operation of the machine, the vendor might be interested in keeping control over its local network and defines the local area network as a separate network domain, with well-defined interface (border bridge/router) to other networks.
- *Technological domains:* Also separating a system into domains along technological borders might be reasonable. For instance, a single 5G/6G logical bridge including radio access network and CN can have a large geographical reach and specific internal operation and management procedures. Therefore, defining a dedicated (wireless) domain for the 5G/6G system (logical TSN bridge) and connecting it to other (wired) domains at the CN can be beneficial from an administrative and management view.
- *Scalability*: Some control and management plane mechanisms might not scale to a large number of network elements, streams, etc. A prime example are the algorithms to calculate schedules for scheduled traffic according to IEEE 802.1Qbv in TSN (gate control lists for egress queues of TSN bridges). Calculating such schedules is an NP-hard problem with no efficient (polynomial) exact solution, i.e., exponential runtime in general. Besides the algorithmic complexity, these algorithms operate on a global view onto the network (network graph of stations, bridges, links) and traffic (streams) according to the fully-centralized model, which obviously induces overhead to collect global information from distributed elements depending on the size of the network. Splitting the network into multiple domains allows for a “divide and conquer” approach, which reduces the size of individual domains, and consequently mitigates the complexity and overhead of controlling and managing a smaller individual domain. Moreover, information to be exchanged between domains or with higher-level controllers (e.g., managing streams passing through several domains) can be aggregated to reduce its traffic volume and the required state each domain has to maintain.
- *Fault tolerance and service protection*: Factoring out parts of a network into individual domains can isolate the failures within one domain and prevent propagation of failures into other domains. For instance, each domain can have its own CNC. Moreover, it can be controlled, which traffic is leaving or entering a domain at its border. Similar to the idea of Frame Replication and Elimination (FRER) [IEEE17-8021CB] providing spatial redundancy and isolation (of the different paths) within a single domain, similar service protection mechanisms can be implemented across multiple domains. If redundant E2E paths via different disjunct domains can be found from source to destination in the inter-domain topology, E2E reliability can be increased by routing messages redundantly via different domains.

We propose the following multi-domain architecture depicted in Figure 4.25 supporting all of these reasons for multi-domain systems.

First of all, the system consists of multiple **administration domains**. Each domain is autonomous in so far as it can operate without the other domains to forward streams between sources and destinations residing within the domain (intra domain streams). To this end, each domain has a CUC/CNC with a

domain-wide, logically centralized view onto the domain including all network elements like TSN bridges, end stations, and intra-domain streams. Similarly, each domain can provide additional management-related services like troubleshooting, maintenance, or monitoring that can be coordinated by the CNC of the domain.

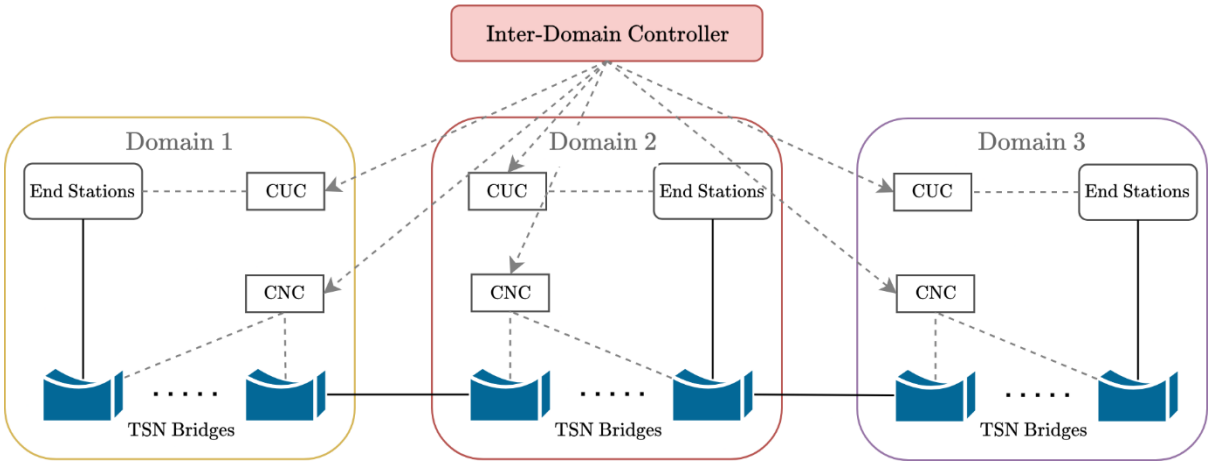

Figure 4.25: Multi-Domain End-to-End Architecture

Each domain can also use different technologies (**technological domains**). For instance, one domain can use only wireline bridges, whereas another domain might implement a logical TSN bridge with wireless 5G/6G technology. In particular, a "6G domain" can utilize the 6G architecture presented above (see Figure 4.25) and present the whole 6G network as a single logical bridge interfacing with other wireline TSN domains.

Inter-domain streams with sources and destinations in different domains require the planning of paths and schedules across domains. To this end, we use a hierarchical approach. We introduce an inter-domain controller, which interacts with the intra-domain controllers of each domain along the E2E path. **Scalability** is increased by a) aggregating multiple streams traversing a domain into aggregated streams that are handled as one batch of frames; b) delegating the planning of intra-domain paths and schedules to intra-domain controllers acting in concert with the inter-domain controller.

Service protection in multi-domain systems can be based on the same principles as within a domain, namely:

- **Active/Standby redundancy (1:1(n)):** One of the paths is selected as the "active path" (a.k.a. primary path) and used for forwarding.
- **Active/Active redundancy (1+1(n)):** All paths are set up and used to forward traffic. One of them is selected as primary path, all the others are backup paths.
- **Replication/Elimination based redundancy (per packet 1+1(n)):** Special form of active/active forwarding, where redundancy is achieved by replication/elimination on a per-packet basis.

From an architecture perspective, replication and elimination (R/E) requires the placement of R/E points in the network. Note that R/E points can be placed at the source and destination, but also *within*

the network. In particular, R/E points can be placed at the border nodes of domains as depicted in Figure 4.26.

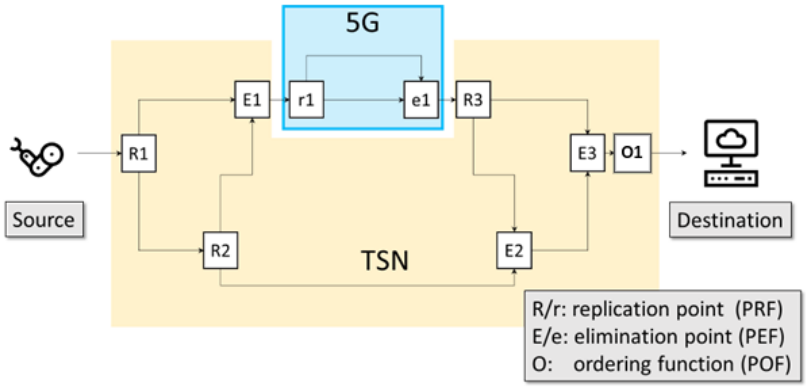

Figure 4.26: Placement of R/E points in two domains (blue and yellow).

Such domain-specific R/E functions also require domain-specific sequence numbers. TSN and DetNet specifications allow “push/pop”- type operation of the sequence numbers to add (push R-tags) sequence numbers at the ingress of the domain and remove (pop R-tags) at the egress. Figure 4.27 shows the usage of the Multi-level R/E concept in an L2 network, where two technology domains (blue and green) are distinguished. Domain border nodes execute the addition (push) and removal (pop) operation of the domain specific R-Tags. Node-A (edge node of the blue domain) adds the level-1 R-Tag and Node-B (edge node of the green domain) pushes the level-2 R-Tag. The domain specific R-Tags are removed at the egress node(s) of the domain (i.e., Node-C and Node-D removes level-2 R-Tag of the green domain, and similarly Node-E removes the blue domain specific level-1 R-Tag).

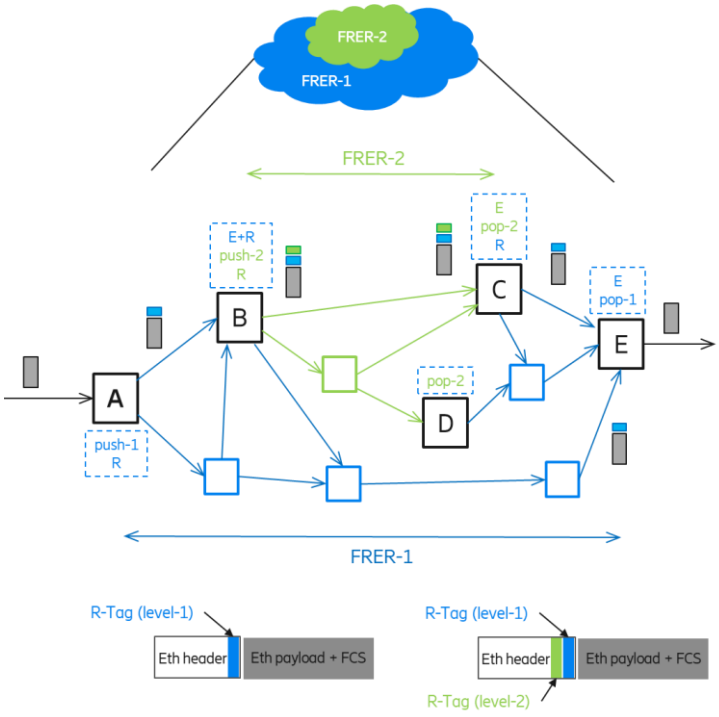

Figure 4.27: Multi-level R/E concept in a L2 network with 2 technology domains (blue and green)

R/E can also be integrated into the 5G system (5GS) components of a domain. Replication happens before the traffic (TSN Stream / DetNet Flow) enters the 5GS and the member streams/flows are transported as disjoint as possible over the 5GS (using separate UE, separate PDU session, separate gNB, separate UPF). Figure 4.28 provides a basic solution to realize a highly reliable mobile system which can provide two independent paths between industry devices. This is achieved by defining reliability groups for the networking entities. By having networking entities in more than one reliability group, redundancy is achieved, which provides protection against failures. For a deeper discussion beyond this architectural view, we refer to [DET25-D35] [3GPP18-23501, Annex F].

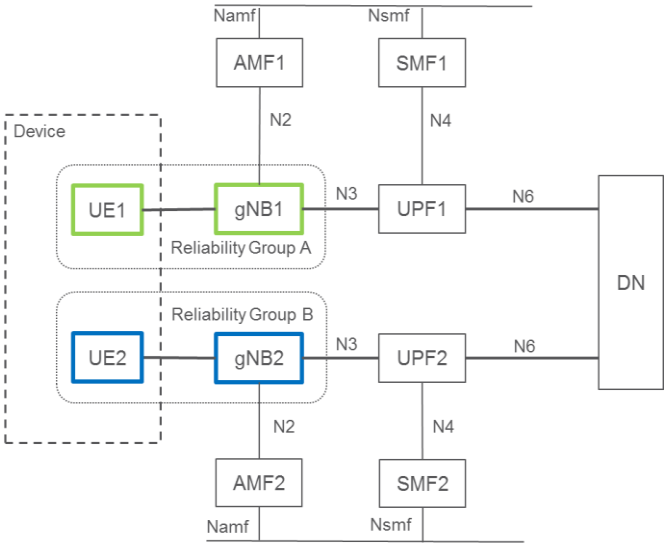

Figure 4.28: Using disjoint resource over the 5GS components (see [3GPP18-23501, Annex F]).

Finally, we would like to highlight that in addition to the control plane interaction through network controllers of the multi-domain system architecture, also the data plane architecture can be extended to support notification-based interaction between domains. In case of notification-based interactions, where notifications are on per-stream basis, scalability might be a concern in larger E2E systems. Two possible inter-domain data plane notifications are “packet arrives soon” notifications and “ready for packet delivery” notifications. The relevant components in the multi-domain architecture are the border bridges of connected domains sending and receiving these notifications to trigger the connected domain to start preparing for the transport of time critical traffic. Figure 4.29 shows an example of a “packet arrives soon” notification to a 5G/6G domain. The trigger notification from the Actor allows the preparation for the forwarding over the radio link, immediately when the data packet arrives.

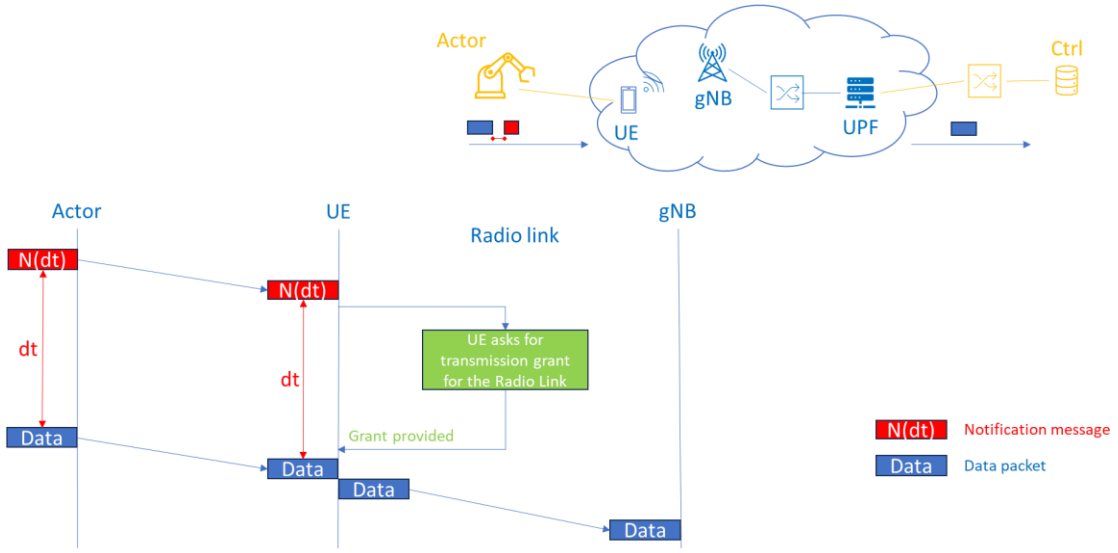

Figure 4.29: “Packet Arrives Soon” notification

4.5.3. Architectural Aspects of Reliable Control Plane and Management

We have proposed and described in [DET25-D36] the design foundation and principles of a reliable control service design, based on SDN controllers. The proposal represents an initial step towards a software network architecture for dependable control services and the management of 6G networks. It is expected to lead to a service model and tools for the dependable control and management of 6G networks, usable as a building block of the control and management plane of a 6G system, applicable mainly to the data network of the 6G systems. The scope of the proposal can be later extended to apply to 6G CN functions such as AMF or SMF, provided that the implementation of these functions relies on distributed and replicated services. One could also envision to extend the scope of this proposal to be applicable to a TSN AF, or a TSN CNC outside the 6G domain, once again, provided that the distributed and replicated implementation of these functions are envisioned. So, the keywords around the applicability of this proposal are controllability, distributed services, and service replication. The proposal also constitutes an initial step towards an architecture for the dependable control and management applications of 6G systems, and the deployment and execution of these applications on edge computing systems.

Concerning reliability, the main goal of our design approach is basically to avoid that a controller constitutes a single point of failure. This means that each network element (NE) needs to be controlled by a certain number of controllers (at-least two controllers), so that the failure of one or many controllers does not threaten the overall reliability of the control system. If a NEs in the transport network (the data plane) is controlled by many controllers, in case one of these controllers fails another controller is always available to control this NE. More generally, if a NE is controlled by a number $P_{\{P>1\}}$ of controllers, then in the worse-case of the failure of P-1 controllers, there is still one controller ready to take over the control role. There is a network between all the controllers in the system, which is used for interactions between these controllers to perform the control operation in a distributed way, either for control operations on a single NE or for control operations on multiple NEs. Figure 4.30 shows an example of such a reliable distributed control system, with four (4) physical control servers intended to control a portion of nine (9) NEs of a transport network. In this example,

the physical servers are linked with a simple ring network for the communications between these physical servers. Also, each physical server can typically support the execution of a certain number of virtual servers (or software servers) – a number varying from 0 to a given maximum number of virtual servers for each physical server – running in a software-defined network service environment based on virtual machines and/or containers. Each of these software servers will be dynamically assigned the role of controller of one or many of the NEs, and each NE will have the possibility to be controlled by a number $P_{\{P>1\}}$ of these software controllers (possibly) running on each of the physical controllers, the role of an actual controller being exclusive to one of the P software controllers and being assigned dynamically.

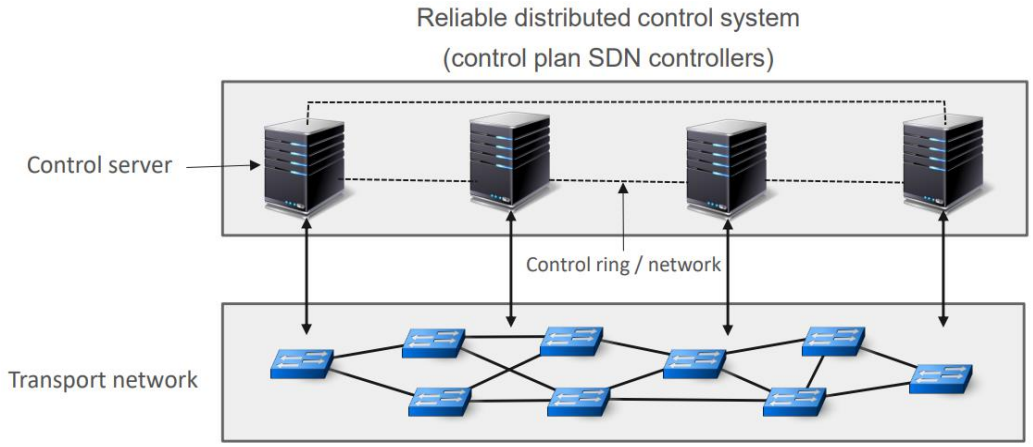

Figure 4.30: Reliable distributed control system for the transport network

A control server, whether physical or virtual, can control a certain number of NEs. The number of NEs controlled by a control server depends on time (the association between control servers and NEs can change dynamically), the load of that control server, the load of other control servers, and the policy behind the dynamic association of control servers with NEs. The policy behind this dynamic association can have as objective a well-balanced load between control servers. Each NE is controllable, potentially controlled (or covered) by at least two (2) control servers, for the purpose of permanent coverage of the controlled NEs and reliability of the control system. Each NE is also controllable by a maximum number of control servers. For a given network element NE_i, a couple (NE_i, MAX_i) is determined, with MAX_i being the maximum number of control servers that can potentially control NE_i, which depends on operational goals related to the level of reliability of the control system and the permanent coverage of NEs. At any time, each NE is assigned to (and is controlled by) exactly one control server. If the assigned control server fails for example, the assignment is transferred to another control server. The set of control servers that can control a NE are also placed on different control nodes, so that to face the situation in which a control node crashes. The association between the control servers and NEs is made to ensure complete coverage of the NEs, redundancy of control servers, and redundancy of control nodes.

The control servers may be used to provide network control and management operations, such as those that can be implemented in a cloud or cloud-native operational support systems (OSS) for data networks for example, based on edge computing resources composed of general-purpose processors. The Open Digital Architecture (ODA) [ODA], which integrates cloud-native technologies as a modern basis for future OSS and business support systems (BSS), can be used for the control and management

of 6G transport and data networks. This implementation and use of ODA-based OSS/BSS services can be seen as a potentially interesting first application domain of the control system design, with would be used to enhance the intrinsic/inherent reliability of 6G transport networks and provide new basis for network management systems, typically those based on the fault, configuration, accounting, performance, security management model and its evolutions for 6G data networks.

4.5.4. Service Design and OPC UA connection management

The industrial applications described in [DET23-D11] [DET25-D13] and chapter 5 come with increasing demands for flexibility from the system in which they are operating. Furthermore, the 6G and TSN networks used by these applications are subject to variations like usage and load of the network. The 6G RAN further needs to handle dynamically changing radio link conditions, caused by radio propagation and also movements by the device and objects in the radio environment. But even applications that are virtualized and executed in a cloud compute environment are impacted by dynamic changes of the workloads in the compute infrastructure. That means that industrial applications, as well as the communication and compute systems that they use in their operation, need to handle dynamically changing conditions and requirements. To achieve dependability in such scenarios with inherent variations, the 6G network platform, and the ongoing applications benefit from interoperation and collaboration. For this it is necessary that applications and the 6G network platform provide the right interfaces and mechanisms for managing dynamic behavior.

In the specific case of dependable applications, such as those considered in the DETERMINISTIC6G project, the application must be capable of describing the services it requires from the system platform, while the system needs a way to feedback its ability to provide a requested service to the application. Moreover, since several applications with distinct requirements may compete for the platform services and resources, a management process must be in place to ensure dependable and optimized system performance.

Managing complex and dynamic systems is a complicated task which typically involves many decisions. Furthermore, due to the increasing complexity of industrial systems, it is virtually impossible to elaborate all the decisions and to provide appropriate configurations to be deployed in a manual manner. This is due to the high amount of knowledge required to make these decisions, that involves the whole system, its services and resources, the executed applications and all their requirements; and the optimization target. Thus, it becomes apparent that manual configuration is infeasible, even for smaller system setups. Furthermore, in the dynamic systems investigated, manual management of the system becomes virtually impossible, due to the dynamic changes in the physical and logical structure of the system and application needs.

Fortunately, decision processes often follow a recurring pattern of steps that may be automated. As presented in [DET25-D13], an abstract sequence of such steps is

1. determine (possibly changed) constraints and conditions of system operation,
2. find possible solutions (e.g., target configurations),
3. rank the found solutions,
4. select the best possible solution, and
5. apply required actions to implement the targeted solution.

To automate these steps, the solutions must be described in a machine-readable format and such description should be represented utilizing a value rank. That is, an adequate service request from an application must include the functional and non-functional requirements of the requested service(s), as well as the input needed to rank the solutions proposed by the system. That is, the request should at least include a list of the minimum performance levels (e.g., data rate), capabilities (e.g., time synchronization), and properties (e.g., reliability) needed for the application to run properly. If the system cannot fulfill the application requirements, such an application cannot be executed, and its potential value is lost. Nonetheless, often applications can be operated with different and less demanding requirements, which then reduces produced value of the application as well. For example, an autonomous vehicle in a factory can operate with lower data rate (performance) for obstacle detection, which would lead to the vehicle moving slower (value) to compensate for the delay in obstacle detection.

To support the operation of applications with different performance levels, the application must provide information about the supported modes and levels of operation. As presented in [DET25-D13], a ***mode of operation*** of an application is defined as *a specific configuration and/or state in which the application operates to achieve its intended performance and reliability objectives.* The mode of operation directly impacts the value that is provided by the application, and in many cases, the application requires the execution of a sequence of modes of operation. For example, an autonomous vehicle might distinguish the modes of operation *loading*, *driving* and *battery charging*.

In contrast, ***levels of operation*** refer to *the distinct hierarchical stages or tiers at which an application functions to meet specific performance, reliability, and quality of service (QoS) requirements. Each level of operation encompasses a unique set of operational parameters and behaviors tailored to ensure the application's dependability.* Again, the value provided by the application differs at the distinguished levels. The levels of operation are intended to define multiple performance levels for a given type of service, or to focus on a specific aspect of the service provided (e.g., safety). For instance, in the *driving* mode of operation of the autonomous vehicle, the levels of operation *fast driving*, *slow driving* or *safe stop* may be distinguished.

Modes and levels of operation can widely differ depending on the application, ranging from switching between a predefined set of parameters within an algorithm, to fully changing the way a functional entity operates, including its input/output parameters. Furthermore, these levels also differ in the requirements the system must fulfill to support them.

Thus, to support different modes and levels of operation, [DET25-D13] proposes that applications provide their different supported levels of operation at once to the DETERMINISTIC6G system. The system can then determine which levels' requirements can be satisfied and then report the selected level of operation back to the application. That is, a change of the active level of operation needs to be aligned between the system and the application by appropriate feedback. This allows the system to automatically consider changing the level of operation of active applications, when operating conditions change or during dynamic changes within the system. This system capability may be used to automatically protect applications with high requirements on service from failure, by degrading the level of operation of other less critical applications. Another major benefit is the possibility to automatically upgrade and optimize the overall system performance if more resources become available or environmental conditions improve significantly. To achieve a coordinated automated switch between different levels of operation, the service request of an application shall also consider

the conditions for a seamless transition. This includes, for example, the time to handle the switch from one active level to another one.

In [DET25-D13] we describe how the OPC UAFX (Unified Architecture Field eXchange) framework can be used to model dependable subservices to support the automatic switch of modes and levels of operation. The subservices modelled are communication, compute, time-synchronization and security. [DET25-D13] also shows examples of how use cases can be modelled using these concepts. UAFX defines the *AutomationComponent* concept. An *AutomationComponent* is an entity that performs one or more automation functions and can communicate with other *AutomationComponents* via logical connections. It can represent a device, a controller, or a function within an edge- or cloud server. An *AutomationComponent* contains two main sub-models: *Assets* and *Functions*. An *Asset* typically describes a physical item, while a *FunctionalEntity* describes logical functionality. The sub-service *Assets* may also require other assets to function correctly. For example, a clock asset is required for certain types of communication. The OPC UA reference model can be used to model such dependency requirements. Figure 4.31 depicts the *AutomationComponent* model of OPC UAFX, with the different elements that build the model.

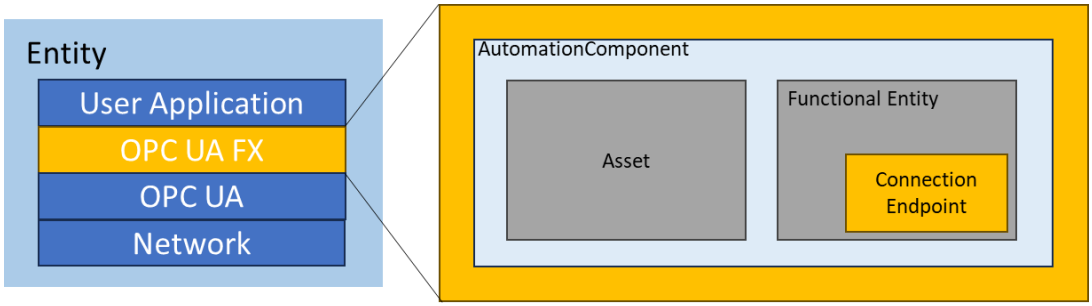

Figure 4.31: *AutomationComponent* model of OPC UA FX.

To model the interactions and data exchange between *AutomationComponents*, UAFX defines *Connections*. A *Connection* is a logical relationship between *Functions*, associated with different *AutomationComponents*. Note that while UAFX defines usage of the publish/subscribe traffic pattern to exchange data on *Connections*, we are not making that restriction. We focus more on the requirements and characteristics (like QoS, etc) of the *Connections*, rather than what protocols and traffic patterns are used to exchange the data. Note that the requirements and characteristics of a *Connection* may vary, depending on the mode and level of operations that the application is operating in. While the work in [DET25-D13] focuses on how the UAFX can be used to model dependable subservices, and the logical connections between them, future work may look more into details on how UAFX can be used to describe services, to be used as input when configuring the network and subservices needed for the service.

Finally, UAFX defines a *ConnectionManager* function, which is responsible for creating and terminating *Connections* between *FunctionalEntities*. The *ConnectionManager* can create the *Connections* based on the service description associated with the requested service.

4.6. Migration from 5G to 6G architecture

As a new 6G RAN and CN functionality will be deployed, it is important to understand how it relates to already deployed 5G networks and how a network migration can take place to integrate new

functionality. A similar challenge has happened before, most recently with the introduction of 5G after finalization of the first 5G standard in 2018. For 5G introduction several options have been standardized, allowing different migration paths from a 4G deployment to 5G. The migration comprised several network domains, the existing 4G radio access network and a new 5G radio access network, the existing 4G CN and the new 5G CN, and the paths on how the new 5G RAN and 5G CN could be connected to existing 4G network domains, resulting also in different ways in which a mobile device i.e. UE can connect to a 5G network [GSMA18] [CRVW18]. This range of migration options required support in the 5G standardization, and a total of seven different migration options (plus some additional sub-variants) were identified and largely standardized. Despite the large standardization effort to define the different migration paths, only few of them have been applied in the network migration towards 5G. By April 2025, around 800 public 5G networks have been deployed around the world by communication service providers; the vast majority of around 74% of the deployed public 5G networks are so-called 5G non-standalone networks (5G NSA) [GSA25], which means that a 5G radio access network is used to provide radio connectivity to the 4G CN. 5G NSA has been intended as an intermediate step towards a 5G standalone (5G SA) deployment, where a 5G UE is connected via the 5G RAN to the 5G Core Network. In particular, it allowed for gradual radio network coverage buildout of 5G. 5G NSA, allows 5G UEs to benefit from 5G capabilities when within 5G radio network coverage, but they continue to maintain the connectivity when leaving the 5G radio network coverage by using the 4G radio network; the connectivity service is in all cases terminated in the 4G CN. A consequence of the still prevailing 5G NSA deployments is that a large part of the standardized functionality of the 5G Core Network is not used in those 5G NSA networks. Examples of functionality not available in 5G NSA are Network Slicing, a cloud-native design, and support for industrial IoT – Ethernet LAN, TSN, time synchronization. This lack of 5G SA functionality remains, even if the amount of UEs that support 5G SA has been steadily increasing and is now with approx. 70% supported by the majority of UEs [GSA25]. It is expected that it will still take several years before 5G SA can be considered common in public 5G networks. In hindsight, the range of (partly complex) migration options has led to a fragmentation of the market and a need for multiple stepwise network investments for network upgrades, and this has contributed to a slow uptake of 5G network adoption (with full 5G SA capabilities).

For the introduction and migration towards 6G, it is proposed to limit the migration options and target directly a 6G standalone deployment providing the full 6G capabilities [RÖT23] [CMRV+23] [Hex2-D21] [3GPP25-6GWS2], as shown in Figure 4.32. As the most efficient migration of the CN, it is proposed to evolve the 5G CN to support the 6G RAN and 6G UEs. Already the 5G CN has been defined as a flexible and extensible network platform, and the service-based architecture is well suited for a cloud-based network deployment. It can be flexibly extended with functionality to support a 6G RAN and 6G UEs [RÖT23] [CMRV+23] [HEX2-D21] while building on the investments that are being made for deploying the 5G CN. In order to provide 6G access to (and aggregation of) existing spectrum carriers that are used by e.g. 5G, it is suggested to apply dynamic multi-RAT spectrum sharing (MRSS) [3GPP25-6GWS2] [Par24] [KSB+24], which allows for efficient and flexible radio capacity sharing (and eventual capacity migration towards 6G). MRSS allows to balance the allocation of spectrum resources between 5G and 6G dynamically depending on need (see Figure 4.34); it also allows to gradually shift the spectrum allocations towards 6G over a longer time as the amount of 6G devices increases and 5G devices decrease.

The timeline of 6G standardization is described in [LGP+24] [3GPP25-6GWS2]. 3GPP standardization work towards 6G started in 2024, where the first phase investigates use cases for and requirements on 6G – in alignment with the framework and objectives specified by ITU for 6G (denoted by ITU as IMT-2030) [ITU23]. Based on this first phase, a technical study phase will take place in 3GPP during approximately 2025-2027 and will be followed by a work item phase in which standard specifications will be developed until approximately the end of 2028, including a self-evaluation to be sent to ITU. This will allow commercial 6G networks to be applied around 2030.

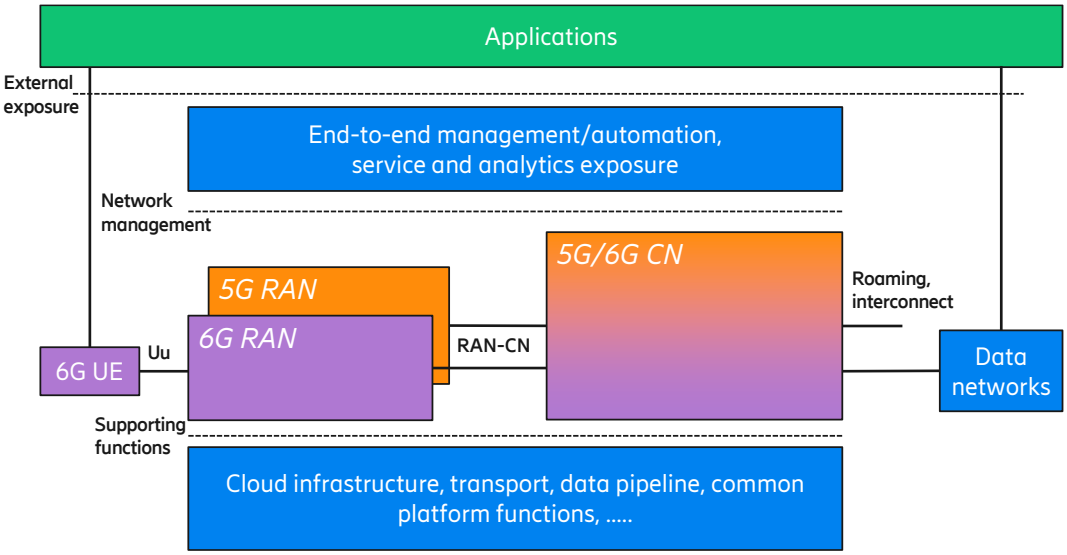

Figure 4.32: Proposal for 6G architecture from [CMRV+23]

The spectrum that is foreseen for 6G [STK+24], includes the spectrum bands allocated to earlier mobile networks like 5G. The anticipated uptake of new services will lead to an increased demand of mobile networks requiring more spectrum to be allocated. New spectrum in centimetric range 7-15 GHz is considered most promising for 6G as it can provide additional mobile network capacity while providing sufficient coverage similar to midband spectrum allocations of 5G, see Figure 4.33. For specific use cases, additional spectrum in the sub-terahertz range above 90 GHz may provide wide blocks of lightly used spectrum. However, it will be challenging to provide wider coverage in this spectrum range but it may play a role for specific use cases.

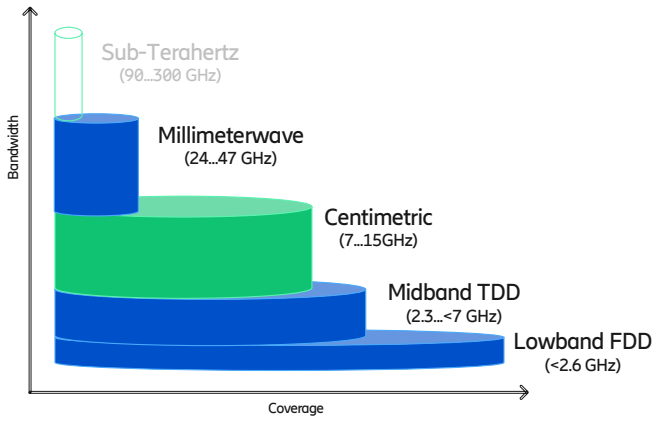

Figure 4.33: Spectrum range for 6G [STK+24]. 5G spectrum is indicated in blue and can be shared with 6G; spectrum in green represents new spectrum allocations for 6G.

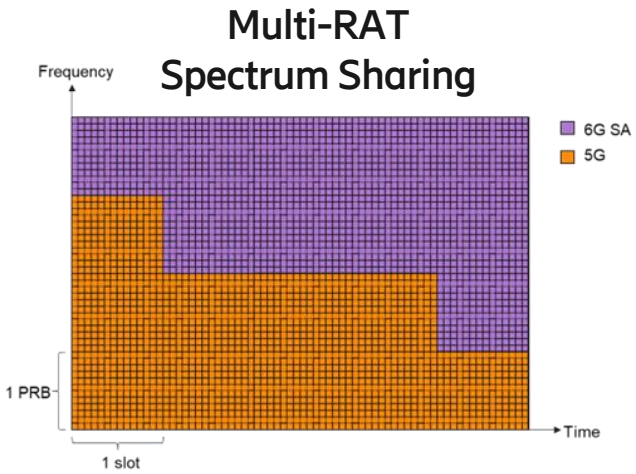

Figure 4.34: Multi-RAT spectrum sharing (MRSS), see [Par24] [KSB+24].

For wide-area networks, a migration from 5G to 6G can build on the deployed 5G networks. A new 6G RAN can be rolled out, and the 5G CN can be evolved to integrate the functionality to connect 6G UEs via the 6G RAN, see Figure 4.32. From a spectrum perspective, the 6G RAN can use new spectrum bands, share a spectrum band between 5G and 6G via MRSS, or migrate a spectrum band from 5G to 6G. 5G UEs connect via the 5G RAN to the evolved 5G/6G CN. 6G UEs connect via the 6G RAN to the evolved 5G/6G CN, which comprises also the new CN functionality required for 6G. Substantial coverage for 6G can be provided, by applying MRSS with 6G on carriers with good coverage, e.g. in lowband [STK+24] as indicated in Figure 4.33.

For local deployments in NPNs a similar migration path exists. For PNI-NPN, the public mobile network is the basis for the NPN realization, and it follows the wide-area migration above. SNPNs differ from wide-area and PNI-NPN deployments in that the available spectrum is typically restricted to spectrum that is locally available, e.g. via local licenses as described in [PHB+25]. A migration of a dedicated NPN deployment from 5G to 6G is indicated in Figures 4.36. Figure 4.35 shows an 5G NPN deployment prior to the introduction of 6G, with the 5G network capabilities as described in section 3.1. We assume that the 5G network integrates with an existing IP-DetNet and/or Ethernet-TSN network. Figure 4.36 shows the same network after the introduction of 6G. A new 6G RAN is added to the network; the CN comprises the 5G CN functionality, but is in addition evolved to include the additional CN functionality required for the 6G RAN and 6G UEs. MRSS is for the spectrum carriers being used in the NPN, which ensures that 6G UEs obtain full coverage within the NPN area with the deployment of the 6G RAN and the CN upgrade. 5G UEs and 6G UEs can simultaneously access the network, where the 5G UEs connect to the 5G RAN and 6G UEs to the 6G RAN. The 5G RAN and 6G RAN can share a common infrastructure, but from the logical perspective they provide separate (5G or 6G) functionality to the corresponding UEs. The same is true for the functionality provided by the 5G/6G CN. This means that while all (5G and 6G) UEs connect to a common mobile network infrastructure, the functionality that is available can differ. E.g. for 6G-connected devices, the UEs and RAN/CN should have better standardized performance observability that enables data-driven performance prediction (see section 4.2). Such functionality may not be available to 5G UEs as it is not supported by the 5G standard. This means that an NPN deployment can migrate from 5G to 6G. Already deployed 5G devices continue to use 5G with their known characteristics; new 6G devices can embrace the novel characteristics and improvements of 6G.

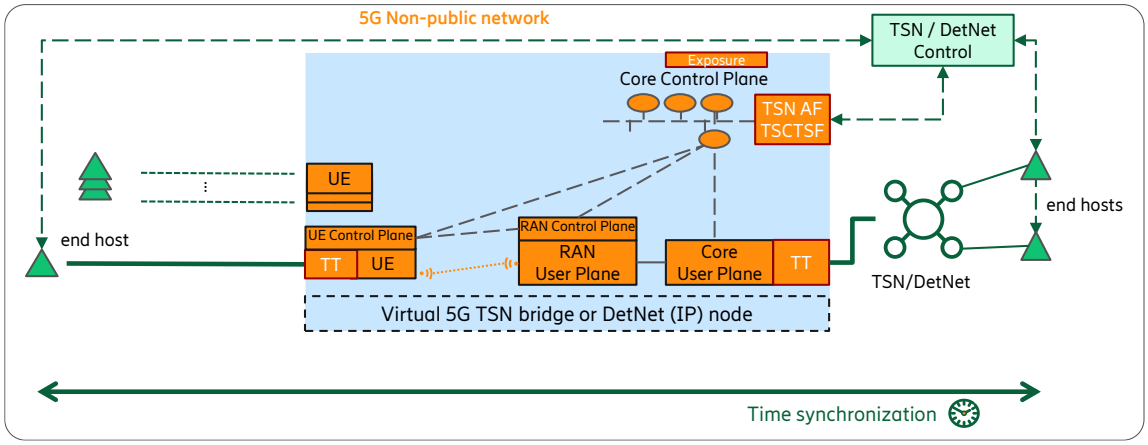

Figure 4.35: Local 5G NPN deployment, integrated with an Ethernet-TSN and/or IP DetNet network.

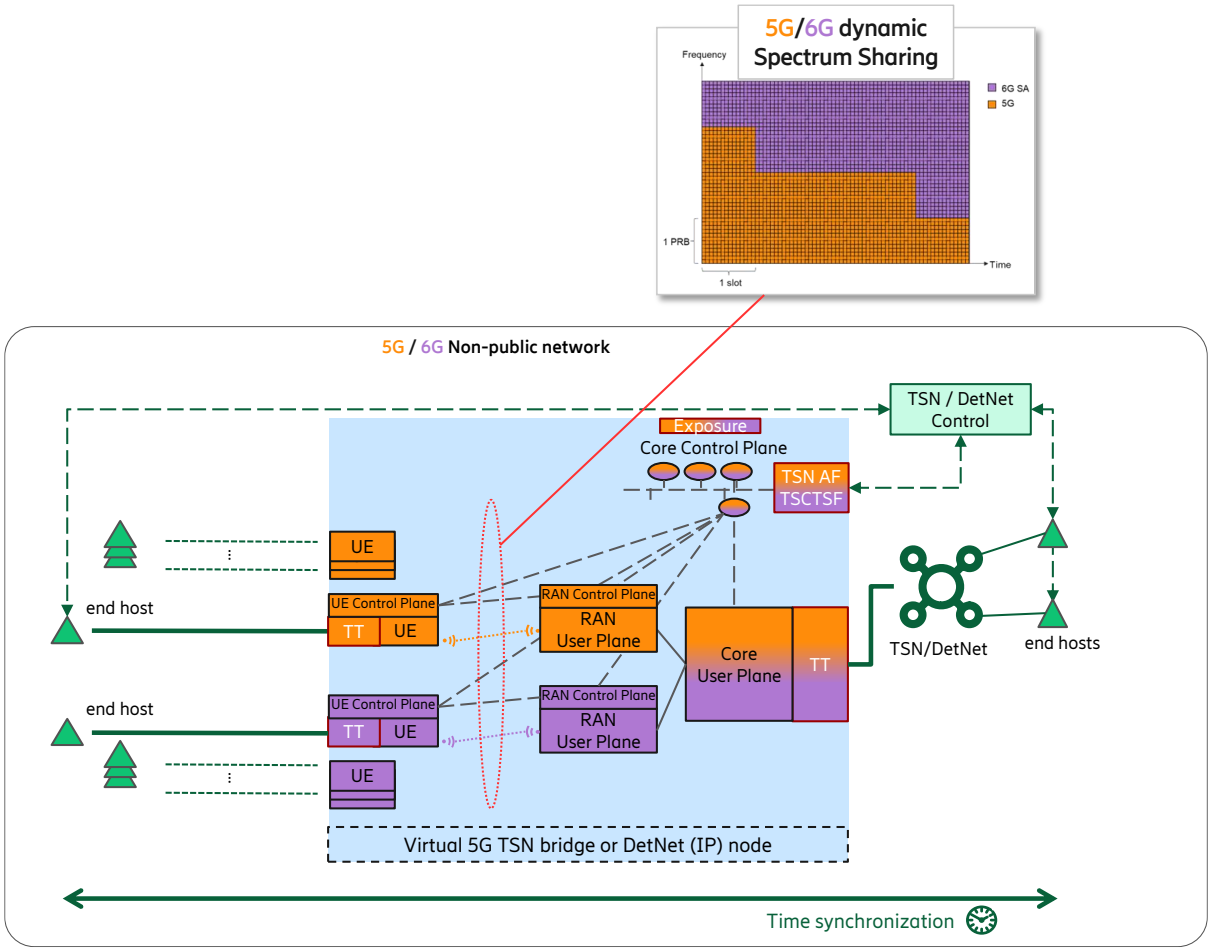

Figure 4.36: Local NPN of Figure 4.35 after the introduction of 6G. The color orange indicates 5G functionality and purple indicates 6G functionality.

5. Review of 6G dependable connectivity for different use cases

Innovative use cases have been described in [DET23-D11] [DET25-D13], which all require dependable time critical communication. In chapter 4 we have described a dependable network architecture for 6G, which integrates the functional components developed in the DETERMINISTIC6G project. In this chapter we describe how the use cases can be realized by the dependable 6G network. Similar as in [DET24-D12], we structure the use case analysis according to the physical deployment according to the following two categories:

- Shopfloor-based use cases with a dedicated network deployment
- Outdoor confined area use cases with a dedicated network service

In fact, both of these use case categories are focused on confined areas: a well-specified geographic region within which the use case takes place. The above use case categories are representative for a larger number of critical use cases requiring dependable network services going beyond what has been described in [DET23-D11] [DET24-D12] [DET25-D13].

5.1. Dependable networks in a shopfloor environment

The shopfloor-based use cases refer to a shopfloor of an industrial site like a factory; often, it is an indoor location within one or more buildings. We consider that in these use cases, a dedicated private 6G network (i.e. a SNPN) is deployed on the shopfloor and is integrated with the existing infrastructure on the shopfloor, which includes a wired deterministic local network based on Ethernet and TSN, and a local data center for shopfloor-related computing tasks. This corresponds to Figure 5.1. The scenario could also be addressed with a PNI-NPN provided by a mobile network operator, which would typically also require 1) a local dedicated build-out of radio sites on the shopfloor for providing sufficient coverage and capacity, 2) one or more local UPF gateways to provide interconnection into the local network on site, 3) a local control plane like a TSN AF that directly connects to the TSN controller on site, and 4) a local compute infrastructure. Industrial environments, like a mine, a port, a construction area, can have indoor and outdoor areas and have similarities with the shopfloor-based and the outdoor confined area use cases.

Figure 5.1 shows an example layout for the shopfloor-based scenario. On the factory shopfloor there are multiple machines, and there is a wired TSN backbone network that interconnects different stationary assets. A server room hosts a local edge cloud, where shopfloor related processing is handled. A 5G/6G private mobile network connects mobile assets (machines, robots, devices) and some stationary assets to the TSN backbone networks. On the shopfloor human workers are active in different tasks and have human-centric mobile devices, which are in our use case XR glasses and wearable exoskeletons. There are also mobile machines and robots handling different tasks on the shopfloor, such as autonomous mobile robots (AMR). A situational analysis of the environment is created by means of stationary cameras, and can be complemented by sensors on mobile devices, such as lidar / camera sensors on AMRs. The spatial compute for the environmental analysis is assumed to happen in the local compute infrastructure in a server room.

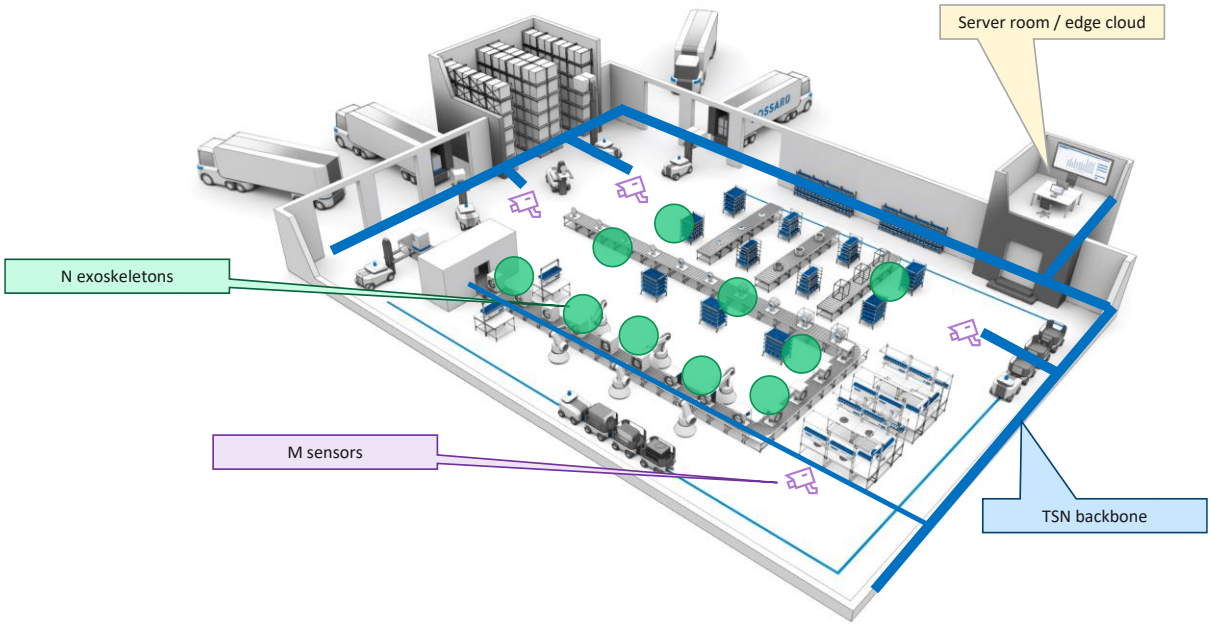

Figure 5.1: Topology of the shopfloor for the industrial use cases.

5.1.1. Shopfloor-based use case overview

Among different use cases that are relevant on a shopfloor, we focus on those that we have investigated in depth in [DET23-D11] [DET24-D12] [DET25-D13].

Two use cases relate to wearable devices for human workers, exoskeletons that support the human worker in physical tasks and XR glasses that allow a human worker to obtain in-depth contextual information from the shopfloor infrastructure.

Industrial Exoskeletons

Occupational exoskeletons (OEs) are wearable robots to reduce the physical load of workers performing demanding activities [MAD20] [DET23-D11] [DET24-D12] [DET25-D13]. Active exoskeletons (i.e. relying on powered actuators to generate the assistive action) have several advantages, such as:

- they can provide adaptive support based on the user's or environment's inputs,
- they may be fully integrated with the smart factory digital ecosystem, allowing the possibility of a real-time monitoring or tuning of the system.

On the other hand, they also have some disadvantages:

- they may be heavy and cumbersome to wear for long periods of time, due to the presence of actuators, electronic components, batteries,
- they might have high-demand requirements in terms of power supply,
- they require real-time, deterministic networking of subsystems (e.g., on board sensors of OEs, external sensors for monitoring the user's status and environment) and elaboration through complex control strategies to deliver the correct amount of assistance depending on the user's needs.

A beneficial approach is to offload the exoskeleton from computing tasks and move them into the edge cloud. This reduces power consumption, weight and size/bulkiness of the exoskeleton and connects easier to environmental sensors (like cameras) that help to identify the optimal task-oriented assistive strategy. Furthermore, the worker's movements and postures can be tracked by a virtual replica of the worker/exoskeleton and a dashboard that allows an ergonomic specialist to recommend corrections from an occupational health perspective. Two alternatives for offloading functionality from the exoskeleton are feasible: 1) offloading the high-level and middle-level control to the edge cloud and maintaining the low-level control on the exoskeleton (see Figure 5.2), or 2) offloading all control to the edge cloud (see Figure 5.3). Further details about the use case can be found in [DET23-D11] [DET24-D12].

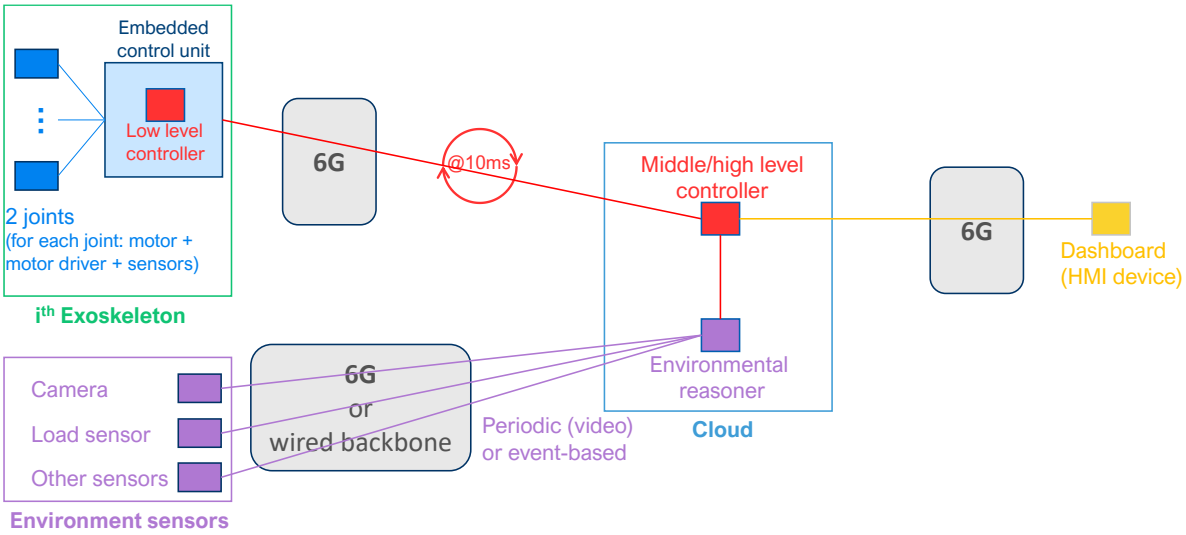

Figure 5.2: Near-term scenario for occupational exoskeleton with offloaded middle/high level control.

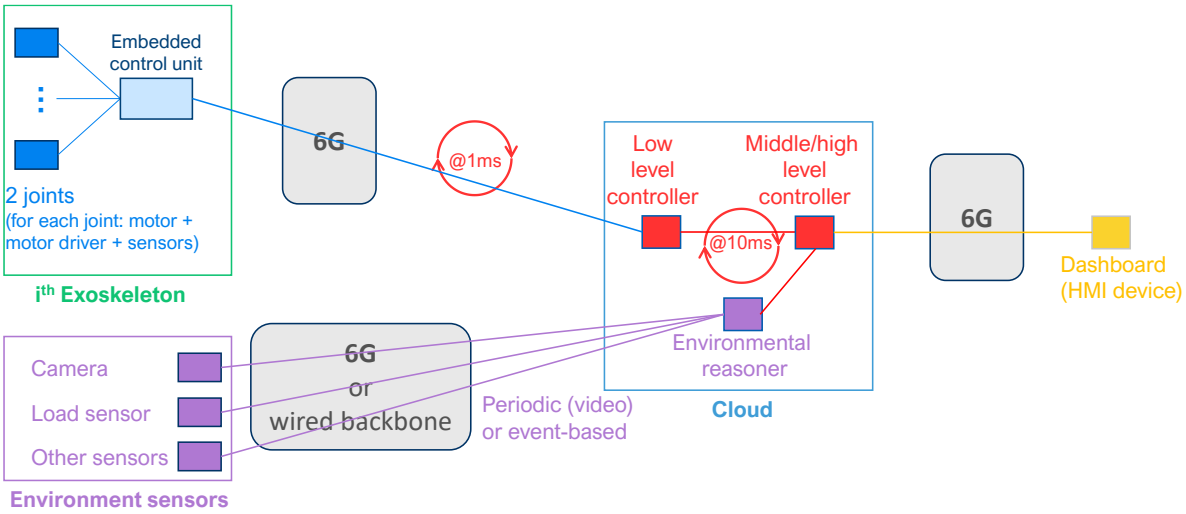

Figure 5.3: Long-term scenario for occupational exoskeleton.

Extended Reality for connected workers

In this use case a worker on the shopfloor can wear XR glasses, which provides digital information related to the work tasks, machines and environment blended into the visually observed physical surrounding. Details about the use case are provided in [DET23-D11] [DET24-D12] [DET25-D13]. The extended reality use case comprises two compute intensive tasks: spatial compute (to obtain a spatial understanding of the local environment) and rendering of the scene with the multiple integrated (digital and physical) objects. Great benefits are obtained if these functions are offloaded from the device (i.e., the XR glasses) to the edge cloud as shown in Figure 5.4. In addition, information about the digital objects related to the shopfloor that are to be immersed into the scene needs to be provided to the rendering engine; this information is provided e.g., by digital twins of the shopfloor assets. XR use cases introduce communication between the XR device and the server to which functionality has been offloaded, which is expected to be located in the server room of the factory. The workers with XR glasses are moving on the shopfloor or are located, e.g., at workstations in Figure 5.1.

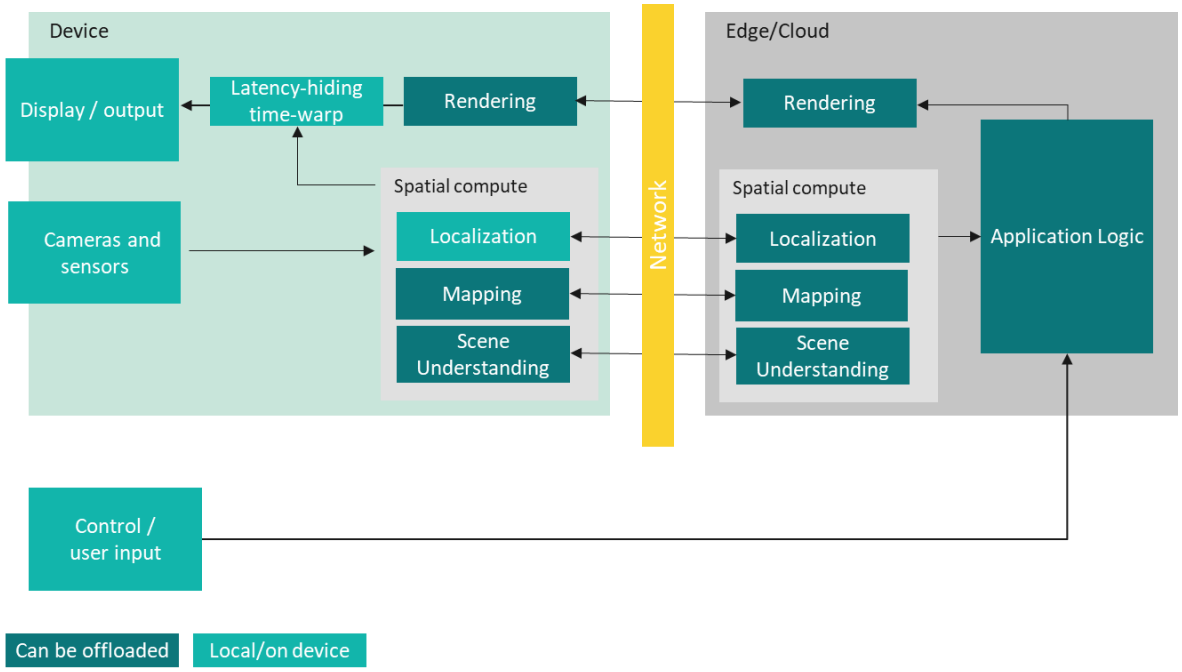

Figure 5.4: XR functionality with offloading options, see [DET23-D11].

Adaptive Manufacturing

This use case is motivated by a higher degree of production flexibility and manufacturing adaptivity. This means that the manufacturing process can easily adapt to changes in demand or production requirements, thereby improving efficiency. Furthermore, the use case can decrease the downtime or changeover time of production lines, leading to an increase in overall productivity. This can significantly enhance the factory's output and potentially lead to increased profitability.

Elements in adaptive productions include automated guided vehicles (AGVs) and Mobile Processing Modules (MPMs), which can move freely within the factory floor and either transport parts or more complex machinery (like tools or robot arms) which can be combined and used in cooperation with stationary components like a processing cell, cameras, and charging stations to reload batteries. For

readability reasons, it is left to the reader to identify appropriate locations of these components in Figure 5.1. The mobile and fixed assets are complemented with functionality located in the edge cloud in the server room which provide task planning and fleet coordination of the AGVs/MPMs, but also object detection and safety functions. More details about the adaptive manufacturing use case can be found in [DET23-D11] [DET24-D12] [DET25-D13].

The logical network architecture for the shopfloor-based use cases, is shown in Figure 5.5.

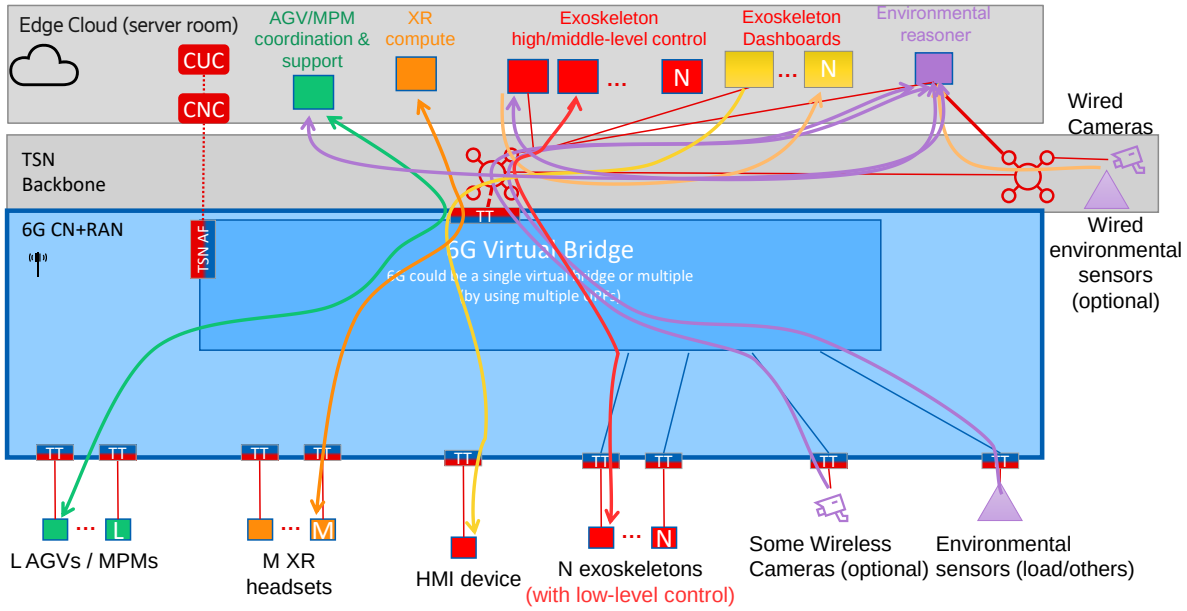

Figure 5.5: Logical 6G network architecture for the occupational exoskeleton (near-term scenario) on the shopfloor.

5.1.2. Functionality for dependable connectivity

In the shopfloor-based deployment and use cases, functionality for E2E dependable communication with 6G is used, as described in chapter 4. The overall network architecture is according to sections 4.1 and 4.6, where the 6G network integrates with the shopfloor network infrastructure and builds on an available 5G network, if available.

All assets and the network on the shopfloor are time synchronized to one or more time domains (e.g. a *wall clock* and a *working clock*) and can apply timing resiliency (see section 4.3.1). Network monitoring and analysis for security can be integrated in the network for further anomaly and threat detection and initiating related countermeasures (see section 4.3.2). End-to-end dependable communication is primarily provided via TSN in the wired network segments and including 6G wireless connectivity (see section 4.3.3). Some applications are virtualized and executed in an edge compute infrastructure, which is tightly integrated into the communication infrastructure and can provide dependable compute (see sections 2.4 and 4.4). The planning of tasks and related configurations are largely coordinated via an OPC UA middleware (see section 4.5.4 and [DET25-D13]); configurations are partly automated or controlled by plant process managers via dashboards. Applications can support different modes and levels of operation; in case of severe resource constraints in the network or compute infrastructure, a best matching level of operation (i.e. application-communication-compute co-design) can be agreed via exposure mechanisms between the application and the network infrastructure (see [DET25-D13] [GSA+25]). A digital twin of the industrial processes and the network

may be used, by exchanging situational information between the network and the application domain; this allows for improved task planning and network configuration (see section 4.3.4). An essential functionality by the 6G network and the 6G RAN is to monitor and predict the achievable performance in terms of packet delays (see sections 4.2.1 and 4.2.2). This allows to agree on suitable levels of operation for the application, provide necessary information for E2E TSN traffic engineering and optimization (see section 4.3.3), and provide useful insights towards a network digital twin (see section 4.3.4). In addition, the 6G network may provide packet delay correction (see section 4.2.3), which may be important for applications that are sensitive to packet delay variations; but it is also effective to significantly improve E2E traffic engineering in TSN (see section 4.3.3).

5.2. Dependable networks in an outdoor environment

For the outdoor use case category, we assume a use case realization that builds on a public 6G mobile network. Figure 5.6 shows an example layout of the outdoor scenario. Machines and other assets are operated in a confined outdoor area (i.e. the agricultural field in the smart farming use case); the smart farming equipment on the field is also connected to the smart farming application backend. We assume that this backend is a cloud-hosted smart farming system. The farmer can access and control the smart farming system from her farm via an application front-end with an operation panel that connects to the cloud-hosted smart farming backend. To connect its smart farming assets, like harvesters, to the smart farming application backend, the farmer uses a dedicated dependable network service of a public mobile network operator (i.e. a PNI-NPN). This dedicated dependable network service can be provided as a virtual dedicated network service restricted to a specified closed group of end devices. We further assume that the public mobile network operator provides a network slice for time-critical connectivity; in this slice connectivity can be configured for devices or applications to provide specified performance, see e.g. [OOA+25] [BSB+25]. In our example, the farmer would specify and request from the mobile network operator the performance levels that the virtual dedicated network would need to provide to the applications and devices of the virtual dedicated networks service. It would also define the area (e.g. the green the smart farming area in Figure 5.6), in which the virtual dedicated network devices would obtain the guaranteed network performance. This virtual dedicated network service with dependable performance would be agreed and documented in an SLA between the mobile network operator and the end user, i.e. the farmer. The SLA would also comprise the availability of the agreed dependable network service, which defines the level of guarantee at which the agreed network performance is provided to the end user. To the farmer this dedicated network service provides a plannable connectivity for all her connected assets. A secure interconnection of the assets can be assume, e.g. based on IPsec. This use case covers a larger geographical area and we assume that the E2E private farming network, that uses the 6G virtual dedicated network service, is realized as an IP network. DetNet can be applied to this IP network to provide dependable IP connectivity for time-critical applications (see section 2.3).

Many outdoor use cases beyond smart farming would follow a similar approach, and we provide some examples to illustrate that the dependable networking principles of the smart farming use cases have a much broader relevance. For example, a team of reporters could book a dedicated dependable network service for a certain area in a certain time span, for media production and reporting from a special event[6]. An enterprise might operate an automated transportation service in a certain region with a fleet of automated or tele-operated vehicles, for which it would require a dedicated dependable

[6] https://github.com/camaraproject/DedicatedNetworks/blob/main/documentation/SupportingDocuments/UsageScenarios.md

network service. Other examples are connected construction sites, delivery robots and drones, and many more. In some cases, some variations of the described situations are possible. For example, in the smart farming use case, the smart farming application backend could also be hosted in a local on-premise compute infrastructure in the farm instead of a cloud-hosted solution. Also, in the targeted coverage area of the field, the network coverage and capacity might be insufficient and for a (longer term) dedicated network service contract the mobile network operator would build out the radio access network within the confined area with, e.g., an additional antenna site or spectrum carrier.

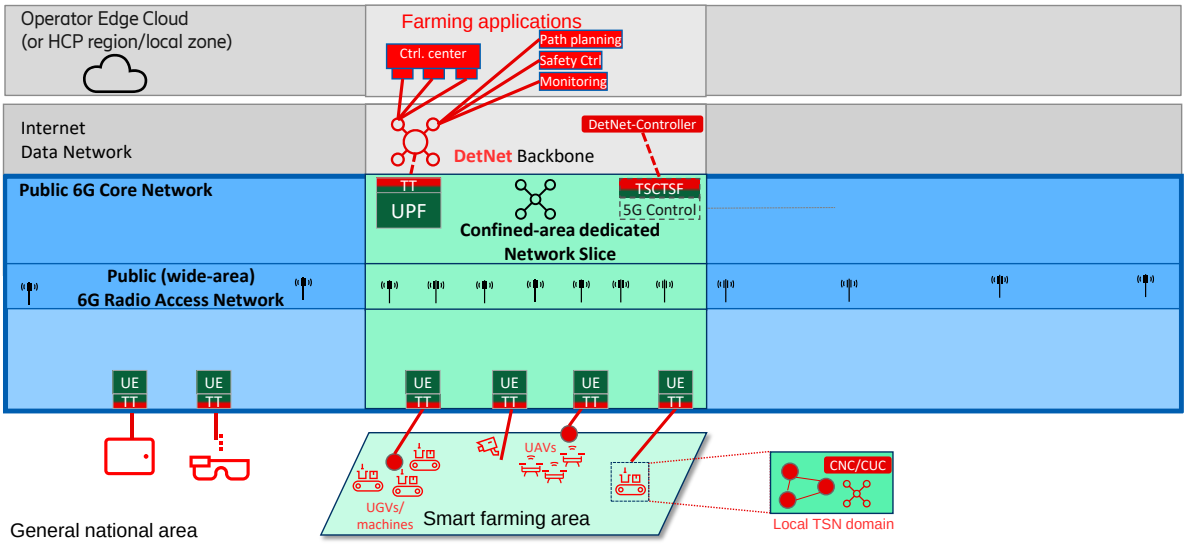

Figure 5.6: Logical 6G network architecture for the smart farming use case.

5.2.1. Outdoor use case overview

Smart Farming addresses a societally important question of cost efficient global food production and supply, which require increased levels of automation to efficiently use, e.g., available land and water. Distinguishing features of this use case [DET23-D11] entail: scalable field monitoring and exploitation of the collected data in managing farming operations, a timely identification of crops affected by pests and bad weather as well as planning ways of crop treatment, ground and/or aerial vehicles which inter-share information and cooperate to execute farming tasks, etc.

Smart Farming targets mobile automation and outdoor communication over large areas. A remote-control center delivers work plans to unmanned ground vehicles (UGVs) and unmanned aerial vehicles (UAVs):

- An **Unmanned Ground Vehicle** is an autonomous or remote-controlled farming vehicle, which is used for different tasks such as ploughing, sowing, harvesting, etc. Examples of an UGV encompass planters, (combine) harvesters, trolleys, and rollers. Being a central field device for the overall farming process, UGVs put forth different communication requirements. For motion planning of a single UGV, a periodic exchange of control data every 2-20 ms is established by its associated motion controller. To (locally) coordinate movements among different UGVs, a complementary, periodic transmission of vehicle status information is carried out every 2-20 ms. In addition, when the UGVs need to coordinate their actions, for instance, a harvester emptying yield to different trolleys, two types of information flow are

used among the vehicles: an aperiodic transmission of, e.g., connect/disconnect commands and a periodic exchange of application status information, every 50-100 ms.

- An **Unmanned Aerial Vehicle** is an autonomous or remote-controlled farming vehicle used for crop inspection, transportation of light objects, detection of obstacles for UGVs, etc. This can, for instance, be a drone or light sport aircraft. Motion planning for each UAV relies on a periodic exchange of control data with its controller, every 2-20 ms. Real-time communication among the UAVs and the UGVs is employed to coordinate their inter-movement while collaborating on a farming task. That communication is a periodic transmission of vehicle status information every 2-20 ms.

The navigation paths of UGVs and UAVs are planned by motion controllers close to the farming vehicles. The global motion planning is based on tasks which are provided by a specific farming application, which, in turn, uses reporting on both task status and vehicle status from the UGVs and/or the UAVs to decide on the execution of the next task. For the decision making, farming applications may also take advantage of sensor data collected from the vehicles by one or more monitoring applications. All farming tasks are supported by safety applications, which are responsible for, e.g., avoiding collisions among the farming vehicles and with field personnel and animals. To this end images and videos are analyzed for possible obstacles. We assume that computationally intensive processing of data is offloaded from the UGVs and UAVs to an edge cloud [DET23-D11]. This reduces power consumption and increases battery life of the unmanned vehicles. Furthermore, edge computing can host AI/ML algorithms, which allow to carry out an advanced fusion of data from different sensors (e.g., temperature, humidity, and air pressure) and, thus, adapt decisions for both high-level work planning and motion control of the farming vehicles. The overall work planning and monitoring of farming operations is handled from a remote control center, which can in specific cases also remotely control motion of the farming vehicles. At the core of the whole system infrastructure is 6G, which is expected to provide a dependable wireless communication for a diverse set of applications and connects the farming equipment to the farming applications. Ultimate benefits of using the technological “pillars” for Smart Farming include a more efficient use of critical resources, such as land and water, improvement in crop yield from existing fields, as well as a reduction of production waste. A more detailed description of the smart farming use case can be found in [DET23-D11] [DET24-D12] [DET25-D13].

5.2.2. Functionality for dependable connectivity

In this outdoor deployment and use case, functionality for E2E dependable communication with 6G is used, as described in chapter 4. The overall network architecture is according to sections 4.1 and 4.6.

The communication interactions in the use case are shown in Figure 5.7. A large part of communication happens between local devices in the field and their applications hosted in the edge cloud. It is important to note, that authorized devices of the virtual dedicated network can also be located outside of the confined farming area. However, if we assume that the SLA for guaranteed dependable network performance is restricted to the confined farming areas, connectivity for devices at other places would not be covered by network performance guarantees. It still allows, e.g., the farmer to remotely monitor farming operations on the field via a dashboard in the remote-control center, where the progress of the operations is tracked.

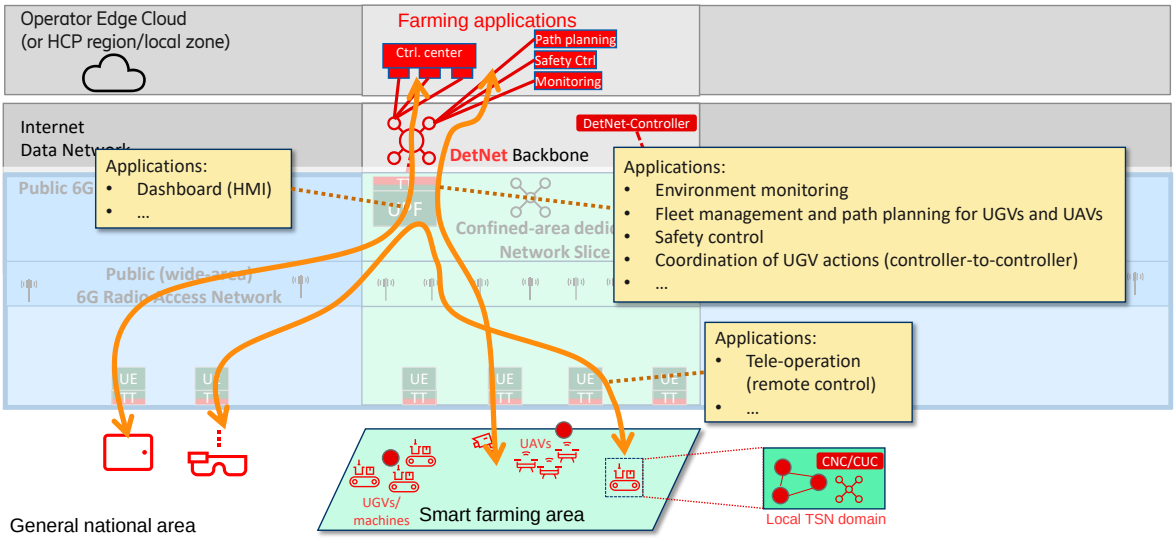

Figure 5.7: Communication relations for the smart farming use case.

Larger machines, like UGVs, have a set of functions integrated, like controllers, cameras, drives, sensors, and actuators that are inter-connected for the autonomous operation of the vehicle. To this end, we assume that a local TSN network within the vehicle is used. However, the UGVs will also communicate with the applications located in the edge cloud. In this case, a function on the UGV communicates via DetNet with an application in the edge cloud, whereas the local communication within the UGV is based on TSN. The DetNet capability can be applied, where DetNet uses TSN as a subnet on a part of the E2E path [5GS21-D53] [5GAC24b], as shown in Figure 5.8. E2E communication builds on multi-domain traffic management, as described in section 4.5.2.

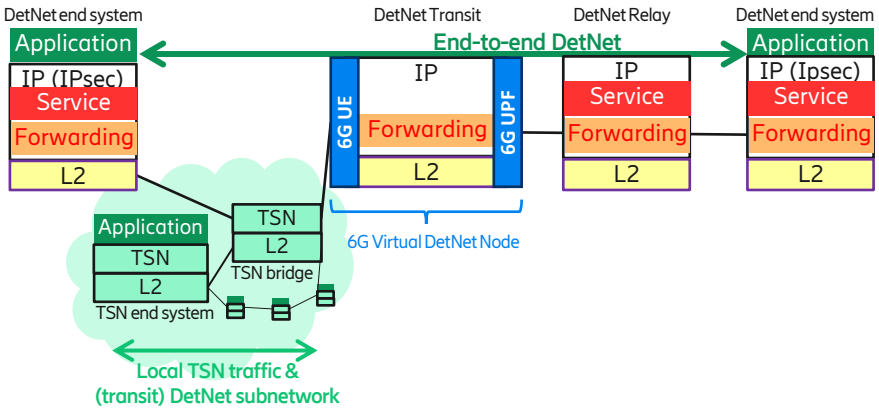

Figure 5.8: DetNet communication from a machine with a local TSN network towards the edge cloud.

In some scenarios multiple vehicles are acting as a synchronized swarm, e.g., when a UAV is used for the purpose of environment surveillance ahead of a UGV, or when a harvester is loading a trolley during the harvesting process. In such a case, communication between, for instance, controllers of the different vehicles pass through multiple TSN configuration domains, as shown in Figure 5.9. In this case also the controller-to-controller coordination within the swarm happens largely within the swarm domain (in contrast to Figure 5.8 where the controllers are in the edge cloud).

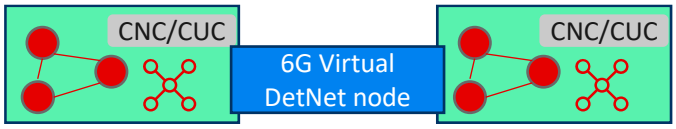

Figure 5.9: Communication across multiple TSN configuration domains, e.g., for interconnecting two vehicles within a swarm, by using the DetNet to inter-connect two TSN domains, see [5GAC24b].

All assets in the use case are time synchronized to one or more time domains and can apply timing resiliency (see section 4.3.1). Network monitoring and analysis for security can be integrated in the network for further anomaly and threat detection and initiating related countermeasures (see section 4.3.2). End-to-end dependable communication is realized via a combination of TSN and DetNet. TSN may be used within local domains, e.g. within a machine using a cabled infrastructure, but also across two TSN segments that are bridged via a DetNet domain, as described in [5GS21-D53] [5GAC24b]. Yet largely connectivity is provided based on IP DetNet. Many applications are virtualized and executed in an edge compute infrastructure, which is tightly connected to the communication infrastructure and can provide dependable compute (see sections 2.4 and 4.4). Communication relationships and their configurations can be provided by the smart farming application suite; however, the usage of an application middleware like OPC UA may be beneficial (see section 4.5.4 and [DET25-D13]). Applications can support different modes and levels of operation. Generally the network is configured to provide guaranteed performance to the application. However, due to the inherently stochastic nature of the underlying systems (e.g, wireless connectivity), in some case of severe resource constraints in the network or compute infrastructure, a best matching level of operation can be agreed via exposure mechanisms between the application and the network infrastructure. This form of application-communication-compute co-design enables some resilience to worst case situations and assures that a suitable operation of the network and application is found that is beneficial to the overall task performance (see [DET25-D13] [GSA+25]). A digital twin of the smart farming processes may be integrated into the smart farming operation center. Allowing the digital twins of the application and the 6G network to exchange situational information between the network and the application domain, enables improved task planning and network configuration (see section 4.3.4). An essential functionality by the 6G network and the 6G RAN is to monitor and predict the achievable performance in terms of packet delays (see sections 4.2.1 and 4.2.2). This allows to agree on suitable levels of operation for the application, provide necessary information for E2E traffic engineering and optimization (see section 4.5.2), and provide useful insights towards a network digital twin (see section 4.3.4). In addition, the 6G network may provide packet delay correction (see section 4.2.3), which may be important for applications that are sensitive to packet delay variations; but it may also improve E2E traffic engineering (see section 4.3.3). The establishment of the dedicated dependable network service is initiated from the end user, in our case the farmer, or the application developer of the farming application system, and it builds on the API-based programmability of the network via network exposure [SKM+21] [OOA+25] [ABJ+24], as depicted in Figure 5.10. As network functionality and performance has been increasing over time, the commercial usage of these capabilities has primarily happened on a best effort basis. In order to make network capabilities easily accessible to applications and the application developers, network programmability via APIs seems a promising direction to enable market adoption and commercialization. Service intents provided via APIs are automatically handled and translated towards an appropriate network configuration. Network exposure through APIs is evolving and a larger ecosystem alignment is appearing in the GSMA's (GSM

Association's) telco global API alliance CAMARA[7]. Beyond configuring network services towards the needs of the application, APIs include functionality for authorization of service requests and a commercialization framework that enables connecting service requests to SLAs [SKM+21] [OOA+25] [ABJ+24]. While APIs enable application developers to integrate the configuration of connectivity into the application design, application developers target a usage of their applications beyond the scope of individual networks and mobile network operators. API aggregation platforms are appearing to provide a global scale to application developers embracing a large number of operator networks, see e.g. GSMA OPG[8]. Examples of API-based network configuration relevant for the smart farming use case, is the *CAMARA dedicated network*[9]. It allows an application developer (as API consumer) to request a virtual private network service with reserved network resources according to some dedicated network profiles and specify, the service area and time validity of the dedicated network and which devices are entitled as members for the dedicated network. The related *CAMARA quality on demand API*[10] allows to request a specific QoS profile for an application traffic flow.

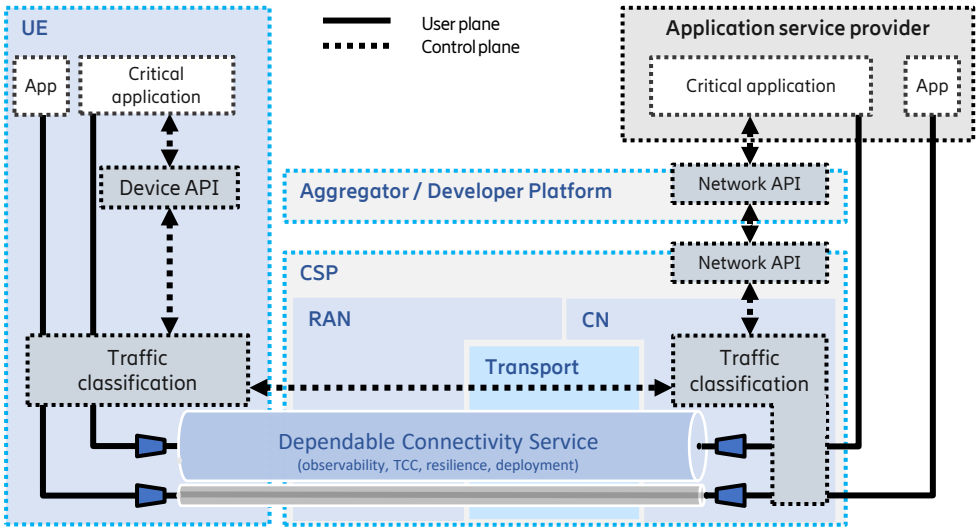

Figure 5.10: Programmable dependable network connectivity for critical applications (source [BSB+25]), see [OOA+25] [BSB+25].

6. Conclusions and Future Work

Digitalization is continuing to drive use cases to increasing levels of adaptivity, embracing of enablers like cloud computing and data-driven design with ML and leading towards a cyber-physical design. This journey is already progressing and will continue well into the time frame of 6G. There will be a wide range of time-critical services with the need for high availability. In many cases, such services will have to be supported E2E by deterministic networking technologies, such as TSN/Ethernet or DetNet/IP, which need to work seamlessly also for (sub-)systems connected wirelessly with 6G. With an interest to adopt cloud computing also for time-critical applications, there is a further need to develop solutions for dependable time-critical computing, which integrate tightly with time-critical

[7] https://camaraproject.org/
[8] https://www.gsma.com/solutions-and-impact/technologies/networks/operator-platform-hp/
[9] https://github.com/camaraproject/DedicatedNetworks/tree/main
[10] https://github.com/camaraproject/QualityOnDemand

E2E communication. DETERMINISTIC6G is studying several use cases with time-critical applications to explore and define corresponding 6G capabilities.

In this report we present an architecture design that integrates functionality for dependable time-critical services as developed in DETEMRINISTIC6G. The proposed architecture integrates robust time synchronization, packet-delay control reducing large packet delay variations, builds on data-driven latency prediction and integrated time-aware edge computing, and considers security-by-design principles for dependable time-critical services. It proposed novel interactions between applications to invoke dependable communication services. Furthermore, novel E2E traffic management for TSN and DetNet is described that operates in conjunction with 6G wireless communication and virtualized application design, and improves E2E dependable networking. We demonstrate how the architecture framework is applied in order to realize the DETERMINISTIC6G use cases, which include local deployments on an industrial shopfloor, but also deployments over wider areas.

References

[3GPP16-28533] 3GPP TS 28.533, “Management and Orchestration of Networks and Network Slicing; Management and Orchestration Architecture (Release 16),” v16.0.0, Jun. 2019.

[3GPP17-23288] 3GPP TS 23.288, “Architecture enhancements for 5G system (5GS) to support network data analytics services,” v17.9.0, June 2023.

[3GPP18-23434] 3GPP TS 23.434,” Service Enabler Architecture Layer for Verticals (SEAL); Functional architecture and information flows”, technical specification, Dec 2018

[3GPP18-23501] 3GPP TS 23.501, “System Architecture for the 5G system,” v18.4.0, Dec. 2023, https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3144

[3GPP19-22261] 3GPP TS 22.261, “Service requirements for the 5G system,” v19.5.0, March 2024.

[3GPP22-37817] 3GPP TR 37.817, “Study on enhancement for data collection for NR and ENDC”, v17.0.0, Apr. 2022.

[3GPP23-23548] 3GPP TS 23.548, ”5G System Enhancements for Edge Computing; Stage 2”, technical specification, April 2023, https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3856

[3GPP23-23558] 3GPP TS 23.558, ”Architecture for enabling Edge Applications”, technical specification, March 2023, https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3723

[3GPP23-29522] 3GPP TS 29.522, ”5G System; Network Exposure Function Northbound APIs; Stage 3”, technical specification, December 2023, https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3437

[3GPP25-6GWS] 3GPP, ”Meeting Report for 3GPP 6G Workshop,” March 2025, 3GPP https://www.3gpp.org/ftp/workshop/2025-03-10_3GPP_6G_WS/Report

[3GPP25-6GWS2] 3GPP, ”Chair’s summary of the 3GPP workshop on 6G,” 3GPP document 6GWS-250238, March 2025, https://www.3gpp.org/ftp/workshop/2025-03-10_3GPP_6G_WS/Docs/6GWS-250238.zip

[5GAC21a] 5G-ACIA, "Exposure of 5G Capabilities for Connected Industries and Automation Applications," white paper, Feb. 2021, https://5g-acia.org/whitepapers/exposure-of-5g-capabilities-for-connected-industries-and-automation-applications-2/

[5GAC21b] 5G-ACIA, “5G non-public networks for industrial scnearios,” white paper, September 2021, https://5g-acia.org/download/18734/?tmstv=1706108824

[5GAC21c] 5G-ACIA, “Integration of 5G with Time-Sensitive Networking for Industrial Communications”, white paper, February 2021, https://5g-acia.org/whitepapers/exposure-of-5g-capabilities-for-connected-industries-and-automation-applications-2/

[5GAC21d] 5G-ACIA, “5G QoS for Industrial Automation”, white paper, November 2021, https://5g-acia.org/whitepapers/5g-quality-of-service-for-industrial-automation-2/

[5GAC23] 5G-ACIA, “Industrial 5G Edge Computing – Use Cases, Architecture and Deployment”, white paper, February 2023, https://5g-acia.org/whitepapers/industrial-5g-edge-computing-use-cases-architecture-and-deployment/

[5GAC24a] 5G-ACIA, “NPNs for Industrial Scenarios,” white paper, March 2024, https://5g-acia.org/download/18718/?version=a4

[5GAC24b] 5G-ACIA, “DetNet-Based Deterministic IP Communication Over a 5G Network for Industrial Applications,” white paper, September 2024, https://5g-acia.org/whitepapers/detnet-based-deterministic-ip-communication-over-a-5g-network-for-industrial-applications/

[5GS20-D14] 5G-SMART Deliverable 1.4, ”Report describing the framework for 5G system and network management functions”, November 2020, https://5gsmart.eu/deliverables/

[5GS20-D51] 5G-SMART Deliverable 5.1, ”First report on new technological features to be supported by 5g standardization and their Implementation impact”, May 2020, https://5gsmart.eu/deliverables/

[5GS20-D52] 5G-SMART Deliverable 5.2, ”First Report on 5g network architecture options and assessments”, November 2020, https://5gsmart.eu/deliverables/

[5GS21-D15] 5G-SMART Deliverable 1.5, ”Evaluation of radio network deployment options”, November 2021, https://5gsmart.eu/deliverables/

[5GS21-D53] 5G-SMART Deliverable 5.3, ”Second report on new technological features to be supported by 5g standardization,” November 2021, https://5gsmart.eu/deliverables/

[5GS21-D55] 5G-SMART Deliverable 5.5, ” Framework for 5g system and network management functions,” November 2021, https://5gsmart.eu/deliverables/

[5GS22-D44] 5G-SMART Deliverable 4.4, ”Report on validation of 6G use cases in the factory”, May 2022, https://5gsmart.eu/deliverables/

[ABJ+24] J. Arkko, M. Björn, W. John, J. Sjöberg, M. Wildeman, G. Wikström, P. Öhlén, “Beyond bit-pipes – new opportunities on the 6G platform”, Ericsson Technology Review, July 2024, available at https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/6g-platform

[AIRAN24] AI-RAN Alliance, “AI-RAN Alliance Vision and Mission White Paper,” 2024, accessed May 2025, https://ai-ran.org/publications/

[AKP+21] F. Alriksson, D.H. Kang, C. Phillips, J.L. Pradas, A. Zaidi, ”XR and 5G: Extended reality at scale with time-critical communication,” in *Ericsson Technology*

Review, vol. 2021, no. 8, pp. 2-13, August 2021, doi: 10.23919/ETR.2021.9904681, https://ieeexplore.ieee.org/document/9904681

[ARS+25] R. Andreoli; R. Mini; P. Skarin; H. Gustafsson; J. Harmatos; L. Abeni, "A Multi-Domain Survey on Time-Criticality in Cloud Computing", IEEE Transactions on Services Computing, vol 18, Issue: 2, March-April 2025, **DOI:** 10.1109/TSC.2025.3539197

[BBW+23] C.G. Blázquez, A. Balador, A. Williams, A, Hata, "Offloading for the future: current use cases and scenarios," Ericsson blog, August 24, 2023, https://www.ericsson.com/en/blog/2023/8/computational-offloading-for-future-innovations

[BKM+98] T. P. Barzilai, D. D. Kandlur, A. Mehra and D. Saha, "Design and implementation of an RSVP-based quality of service architecture for an integrated services Internet," in *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 3, pp. 397-413, April 1998, doi: 10.1109/49.669047

[BSB+19] D. Bruckner, M.-P. Stănică, R. Blair, S. Schriegel, S. Kehrer, M. Seewald, T. Sauter, „An Introduction to OPC UA TSN for Industrial Communication Systems," Proceedings of the IEEE, 2019, doi: 10.1109/JPROC.2018.2888703

[BSB+25] H. Bergström, J. Sachs, L. Boström, R. Wang, S. Sorrentino, J. Vikberg, "Dependable networks: from best-effort to guaranteed performance", Ericsson Technology Review, May 2025, available at https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/dependable-networks

[CAM24a] CAMARA, "Dedicated Networks," draft API, accessed May 25, 2024, https://github.com/camaraproject/DedicatedNetworks

[CAM24b] CAMARA, "Quality on Demand," draft API, accessed May 25, 2025, https://github.com/camaraproject/QualityOnDemand

[CAS+22] J.B. Caro, J. Ansari, J. Sachs, P. de Bruin, S. Sivri, L. Grosjean, N. König, R. H. Schmitt, "Empirical Study on 5G NR Cochannel Coexistence" Electronics 11, no. 11: 1676. May 2022, https://doi.org/10.3390/electronics11111676

[CAS+23] J. B. Caro, J. Ansari, A.R. Sayyed, P. de Bruin, J. Sachs, N. König, R.H. Schmitt, "Empirical study on 5G NR Adjacent Channel Coexistence," 2023 IEEE Wireless Communications and Networking Conference (WCNC), Glasgow, United Kingdom, 2023, pp. 1-6, doi: 10.1109/WCNC55385.2023.10119074.

[CCG+22] Clemm, A., Ciavaglia, L., Granville, L. Z., & Tantsura, J. (2022). RFC 9315: intent-based networking-concepts and definitions.

[CFL+21] B. Cellarius, R. Fritzsche, T. Lohmar, F.-C. Kuo, "Design of an FRMCS 5G E2E System for Future Rail Operation," Technical report, October 2021, https://digitale-schiene-deutschland.de/Downloads/StudyReport_Ericsson_DB_FRMCS_5G_Design.pdf

[CMRV+23] T. Cagenius, G. Mildh, G. Rune, J. Vikberg, M. Wahlqvist, P. Willars, "6G network architecture – a proposal for early alignment", in *Ericsson Technology Review*, vol. 2023, no. 11, pp. 2-7, October 2023, doi: 10.23919/ETR.2023.10313589, https://ieeexplore.ieee.org/document/10313589

[CRVW18] T. Cagenius, A. Ryde, J. Vikberg, P. Willars, “Simplifying the 5G ecosystem by reducing architecture options”, in *Ericsson Technology Review*, November 2018, https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/simplifying-the-5g-ecosystem-by-reducing-architecture-options

[DEH+25] F. Dürr, S. Egger, L. Haug, J. Sachs, J. Farkas, "Control Plane Extensions for Wireless-Aware Traffic Engineering," IEEE 802 plenary session , March 10, 2025, https://www.ieee802.org/1/files/public/docs2025/new-farkas-control-plane-extensions-for-wireless-aware-TE-0325-v03.pdf

[DET23-D11] DETERMINISTIC6G, Deliverable 1.1, “DETERMINISTIC6G use cases and architecture principles,” Jun. 2023, https://deterministic6g.eu/index.php/library-m/deliverables

[DET23-D21] DETERMINISTIC6G, Deliverable 2.1, “First report on 6G centric enablers”, Dec. 2023, https://deterministic6g.eu/index.php/library-m/deliverables

[DET23-D22] DETERMINISTIC6G, Deliverable 2.2, “First Report on the time synchronization for E2E time awareness,” Dec. 2023, https://deterministic6g.eu/index.php/library-m/deliverables

[DET23-D31] DETERMINISTIC6G, Deliverable 3.1, “Report on 6G convergence enablers towards deterministic communication standards,” Dec. 2023, https://deterministic6g.eu/index.php/library-m/deliverables

[DET23-D32] DETERMINISTIC6G, Deliverable 3.2, “Report on the Security solutions,” Dec. 2023, https://deterministic6g.eu/index.php/library-m/deliverables

[DET23-D41] DETERMINISTIC6G, Deliverable 4.1, “DETERMINISTIC6G DetCom simulator framework release 1,” Dec. 2023, https://deterministic6g.eu/index.php/library-m/deliverables

[DET23-D42] DETERMINISTIC6G, Deliverable 4.2, “Latency measurement framework,” March 2024, https://deterministic6g.eu/index.php/library-m/deliverables

[DET24-D12] DETERMINISTIC6G, Deliverable 1.2, “First report on DETERMINISTIC6G architecture,” April 2024, https://deterministic6g.eu/index.php/library-m/deliverables

[DET24-D33] DETERMINISTIC6G, Deliverable 3.3, “Report on Deterministic edge computing and situational awareness via digital twinning security solution”, Jun. 2024, https://deterministic6g.eu/index.php/library-m/deliverables

[DET24-D34] DETERMINISTIC6G, Deliverable 3.4, “Optimized deterministic end-to-end schedules for dynamic systems,” June 2024, https://deterministic6g.eu/index.php/library-m/deliverables

[DET25-D13] DETERMINISTIC6G, Deliverable 1.3, “Report on dependable service design”, Jan. 2025, https://deterministic6g.eu/index.php/library-m/deliverables

[DET25-D23] DETERMINISTIC6G, Deliverable 2.3, “Second report on 6G centric enabler,” Apr. 2025, https://deterministic6g.eu/index.php/library-m/deliverables

[DET25-D24] DETERMINISTIC6G, Deliverable 2.4, “Second report on the time synchronization for E2E time awareness,” Apr. 2025, https://deterministic6g.eu/index.php/library-m/deliverables

[DET25-D35] DETERMINISTIC6G, Deliverable 3.5, “Multi-domain end-to-end schedules”, Apr. 2025, https://deterministic6g.eu/index.php/library-m/deliverables

[DET25-D36] DETERMINISTIC6G, Deliverable 3.6, “Second report on deterministic edge computing and situational awareness via digital twinning”, Apr. 2025, https://deterministic6g.eu/index.php/library-m/deliverables

[DET25-D43] DETERMINISTIC6G, Deliverable 4.3, “Latency measurement data and characterization of RAN latency from experimental trials,” April 2025, https://deterministic6g.eu/index.php/library-m/deliverables

[DET25-D44] DETERMINISTIC6G, Deliverable 4.4, “DETERMINISTIC6G DetCom simulator framework 2,” April 2025, https://deterministic6g.eu/index.php/library-m/deliverables

[DET25-D45] DETERMINISTIC6G, Deliverable 4.5, “Validation for DETERMINISTIC6G concept”, Apr. 2025, https://deterministic6g.eu/index.php/library-m/deliverables

[ECAT] EtherCAT Technology Group, EtherCAT Specification, https://www.ethercat.org/en/downloads/downloads_A02E436C7A97479F9261FDFA8A6D71E5.htm

[EDV+25] S. Egger, F. Dürr, B. Varga, M. De Andrade, G.P. Sharma, J. Sachs, J. Harmatos, J. Gross, “Wireless-aware TSN Engineering: Implications for 5G and Upcoming 6G Networks” in *IEEE Network*, vol. 39, no. 3, pp. 99-107, May 2025, doi: 10.1109/MNET.2025.3556002. https://ieeexplore.ieee.org/document/10945893

[EGS+25] S. Egger, J. Gross, J. Sachs, G. P. Sharma, C. Becker, and F. Dürr, “End-to-End Reliability in Wireless IEEE 802.1Qbv Time-Sensitive Networks,” to appear in proceedings of the IEEE/ACM International Symposium on Quality of Service (IWQoS) 2025, Gold Coast, Australia, July 2-4, 2025.

[Eri23] Ericsson, “Future Network Architecture”, Ericsson White Paper, April 2023, available at https://wcm.ericsson.net/en/future-technologies/architecture

[FMK23] J. Friman, E. Mueller and B. van Kaathoven, "Monetizing API exposure for enterprises with evolved BSS," in Ericsson Technology Review, vol. 2023, no. 1, pp. 2-10, January 2023, doi: 10.23919/ETR.2023.10035875. https://ieeexplore.ieee.org/document/10035875

[GLR+20] I. Godor, M. Luvisotto, S. Ruffini, K. Wang, D. Patel, J. Sachs, O. Dobrijevic, D. P. Venmani, O. Le Moult, J. Costa-Requena, A. Poutanen, C. Marshall, J. Farkas, "A Look Inside 5G Standards to Support Time Synchronization for Smart Manufacturing," in *IEEE Communications Standards Magazine*, vol. 4, no. 3, pp.

14-21, September 2020, doi: 10.1109/MCOMSTD.001.2000010. https://ieeexplore.ieee.org/document/9204594

[GLS+22] L. Grosjean, K. Landernäs, B. Sayrac, O. Dobrijevic, N. König, D. Harutyunyan, D. Patel, J.F. Monserrat, J. Sachs, "5G-enabled smart manufacturing – a booklet of 5G-SMART," technical report, September 2022, https://arxiv.org/abs/2209.10300

[GSA+25] L. Grosjean, J. Sachs, J. Ansari, N. Reider, A. Hernandez Herranz, C. Holmberg, „A Framework for Communication–Compute–Control Co-Design in Cyber–Physical Systems," *Electronics* vol. 14, no. 5, February 2025. https://doi.org/10.3390/electronics14050864

[GSA25] Global mobile Suppliers Association (GSA), "5G Standalone, Status upate April 2025," April 2024, https://gsacom.com/paper/5g-standalone-april-2025/

[GSD+22] G. Seres, D. Schulz, O. Dobrijevic, A. Karaağaç, H. Przybysz, A. Nazari, P. Chen, M.L. Mikecz, Á.D. Szabó, "Creating programmable 5G systems for the Industrial IoT," Ericsson Technology Review, vol. 2022, no. 10, pp. 2-12, October 2022, doi: 10.23919/ETR.2022.9934828. https://ieeexplore.ieee.org/document/9934828

[GSMA18] GSMA, "Road to 5G: Introduction and Migration", report, April 2018, https://www.gsma.com/futurenetworks/wp-content/uploads/2018/04/Road-to-5G-Introduction-and-Migration_FINAL.pdf

[HEX2-D21] HEXA-X-II, "Initial Architectural enablers," deliverable D3.2, October 2023, , https://hexa-x-ii.eu/results/

[HEX-D14] HEXA-X, "Hexa-X architecture for B5G/6G networks – final release," deliverable D1.4, July 2023, https://hexa-x.eu/deliverables/

[HHA+24] D. Hamidovic, A. Hadziaganovic, M. Ahmed, R. Muzaffar, M. De Andrade, J. Sachs, H.-P. Bernhard, "6G Schedule and Application Traffic Alignment for Efficient Radio Resource Utilization", IEEE 100th Vehicular Technology Conference (VTC2024-Fall), 2024. DOI: 10.1109/VTC2024-Fall63153.2024.10757713

[HSG+25] A. Hadziaganovic, J. Sachs, J. Gross, D: Hamidovic, M. Ahmed, R. Muzaffar, A. Springer, H.-P. Bernhard, "Digital Twins of Industrial and 6G Systems: Enablers Towards Situational Awareness," to appear in 30th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Porto, September 9-12 2025.

[IEEE17-8021CB] IEEE Std 802.1CB-2017 "IEEE Standard for Local and metropolitan area networks--Frame Replication and Elimination for Reliability," Oct. 2017, https://standards.ieee.org/ieee/802.1CB/5703/

[IEEEQcc] IEEE Std 802.1Qcc-2018 "IEEE Standard for Local and Metropolitan Area Networks--Bridges and Bridged Networks -- Amendment 31: Stream Reservation Protocol (SRP) Enhancements and Performance Improvements," (Amendment to IEEE Std 802.1Q-2018 as amended by IEEE Std 802.1Qcp-2018) , vol., no., pp.1-208, 31 Oct. 2018, doi: 10.1109/IEEESTD.2018.8514112. https://ieeexplore.ieee.org/document/8514112

[IEEEQdj] IEEE Std 802.1Qdj-2024 "IEEE Standard for Local and Metropolitan Area Networks--Bridges and Bridged Networks -- Amendment 38: Configuration Enhancements for Time-Sensitive Networking,", 31 May 2024. https://standards.ieee.org/ieee/802.1Qdj/7669/

[IEEEQee] "IEEE Draft Standard for Local and Metropolitan Area Networks--Bridges and Bridged
Networks. – Amendment: Traffic Engineering for Bridged Networks with Significant Delay Variance," project authorization request (PAR) for the IEEE P802.1Qee amendment to IEEE Standard 802.1Q-2022, May 2025, https://www.ieee802.org/1/files/public/docs2025/ee-draft-PAR-0525-v01.pdf

[IEEE-TSN] IEEE 802.1 Time-Sensitive Networking (TSN) Task Group, https://1.ieee802.org/tsn/ (accessed June 2025)

[IETF-DETNET] IETF Deterministic Networking Working Group, https://datatracker.ietf.org/wg/detnet/about/ (accessed June 2025)

[IJR+23] M. Iovene, L. Jonsson, D. Roeland, M. D'Angelo, G. Hall, M. Erol-Kantarci, "Defning AI native: A key enabler for advanced intelligent telecom networks," Ericsson white paper, February 2023, https://www.ericsson.com/en/reports-and-papers/white-papers/ai-native

[ITU23] ITU-R, "Framework and overall objectives of the future development of IMT for 2030 and beyond," recommendation ITU-R M.2160-0, November 2023, https://www.itu.int/en/ITU-R/study-groups/rsg5/rwp5d/imt-2030/Pages/default.aspx

[JPK+25] J. Jin, Z. Pang, J. Kua, Q. Zhu, K. H. Johansson, N. Marchenko, D. Cavalcanti, "Cloud-Fog Automation: The New Paradigm towards Autonomous Industrial Cyber-Physical Systems," in IEEE Journal on Selected Areas in Communications, May 2025, doi: 10.1109/JSAC.2025.3574587. https://ieeexplore.ieee.org/document/11016756

[KDS+23] A. Karaagac, O. Dobrijevic, D. Schulz, G. Seres, A. Nazari, H. Przybysz, J. Sachs, "Managing 5G Non-Public Networks from Industrial Automation Systems", 2023 IEEE 19th International Conference on Factory Communication Systems (WFCS), Pavia, Italy, 2023, pp. 1-8, doi: 10.1109/WFCS57264.2023.10144248. https://ieeexplore.ieee.org/document/10144248

[KPB+24] H. Lyu, Z. Pang, A. Bengtsson, S. Nilsson, A. J. Isaksson, G. Yang, "Latency-Aware Control for Wireless Cloud-Fog Automation: Framework and Case Study," in *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 5400-5410, July 2024, doi: 10.1109/TASE.2024.3420770. https://ieeexplore.ieee.org/document/10586839

[KSB+24] A. Karapantelakis, B. Sahlin, B. Balakrishnan, D. Roeland, G. Rune, G. Wikström, H. Wiemann, J. Arkko, K. Vandikas, M. Coldrey, P. Lioliou, P. Persson, P. Schliwa-Bertling, P.-E. Eriksson, P. Öhlén, R. Baldemair, S. Parkvall, W. John, "Co-creating a cyber-physical world," Ericsson white paper, July 2024, https://www.ericsson.com/en/reports-and-papers/white-papers/co-creating-a-cyber-physical-world

[KSR+24] L. Khaloopour, Y. Su, F. Raskob, T. Meuser, R. Bless, L. Janzen, K. Abedi, M. Andjelkovic, H. Chaari, P. Chakraborty, M. Kreutzer, M. Hollick, T. Strufe, N. Franchi, V. Jamali, "Resilience-by-Design in 6G Networks: Literature Review and Novel Enabling Concepts," in *IEEE Access*, vol. 12, pp. 155666-155695, 2024, doi: 10.1109/ACCESS.2024.3480275.

[LGP+24] D.C. Larsson, A. Grövlen, S. Parkvall, O. Liberg, "6G standardization – an overview of timeline and high-level technology principles," Ericsson blog, March 22, 2024, https://www.ericsson.com/en/blog/2024/3/6g-standardization-timeline-and-technology-principles

[LSW+19] O. Liberg, M. Sundberg, E. Wang, J. Bergman, J. Sachs, G. Wikström, *Cellular IoT - from massive deployments to critical 5G applications*, Nov. 2019, Academic Press, ISBN: 9780081029022

[MAD20] Monica L, Anastasi S and Draicchio F, "Occupational exoskeletons: Wearable robotic devices and preventing work-related musculoskeletal disorders in the workplace of the future", pp. 1–12., September 2020, https://osha.europa.eu/en/publications/occupational-exoskeletons-wearable-robotic-devices-and-preventing-work-related

[MAG+19] A. Mahmood, M. I. Ashraf, M. Gidlund, J. Torsner and J. Sachs, "Time Synchronization in 5G Wireless Edge: Requirements and Solutions for Critical-MTC," in IEEE Communications Magazine, vol. 57, no. 12, pp. 45-51, December 2019, doi: 10.1109/MCOM.001.1900379. https://ieeexplore.ieee.org/document/8930825

[MRW14] R. Motamedi, R. Rejaie and W. Willinger, "A Survey of Techniques for Internet Topology Discovery," in *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 1044-1065, Secondquarter 2015, doi: 10.1109/COMST.2014.2376520.

[Nur21] A. Y. Nur, "Analysis of Autonomous System Level Internet Topology Graphs and Multigraphs," 2021 International Symposium on Networks, Computers and Communications (ISNCC), Dubai, United Arab Emirates, 2021, pp. 1-7, doi: 10.1109/ISNCC52172.2021.9615677

[ODA] TM Forum, "The Open Digital Architecture (ODA)", https://www.tmforum.org/oda/about/

[OLK+24] Orlandi, B.; Lataste, S.; Kerboeuf, S. et al., "Intent-Based Network Management with User-Friendly Interfaces and Natural Language Processing," 2024 27th Conference on Innovation in Clouds, Internet and Networks (ICIN), Paris, France, 2024, pp. 163-170, doi: 10.1109/ICIN60470.2024.10494458.

[OOA+25] H. Olofsson, K. Olsson, F. Alriksson, A. Seth, B. Zhong, J. Backman, "Differentiated connectivity: Unleashing the full potential of 5G," Ericsson white paper, April 2025, https://www.ericsson.com/en/reports-and-papers/white-papers/differentiated-connectivity-unleashing-the-full-potential-of-5g

[Par24] S. Parkvall, "6G RAN – key building blocks for new 6G radio access networks," Ericsson blog, May 15 2024, https://www.ericsson.com/en/blog/2024/5/future-6g-radio-access-network-design-choices

[PDR+21] D. Patel, J. Diachina, S. Ruffini, M. De Andrade, J. Sachs and D. P. Venmani, "Time error analysis of 5G time synchronization solutions for time aware industrial networks", 2021 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control, and Communication (ISPCS), NA, FL, USA, 2021, pp. 1-6, doi: 10.1109/ISPCS49990.2021.9615318. https://ieeexplore.ieee.org/document/9615318

[PHB+25] F. Pedersen, R. Högman, M. Buchmayer, A. Zaidi, „5G spectrum for local industrial networks," Ericsson white paper, March 2025, https://www.ericsson.com/en/reports-and-papers/white-papers/5g-spectrum-for-local-industrial-networks

[PLNK] EPSG Draft Standard 301, Ethernet POWERLINK Communication Profile Specification Version 1.5.1, 2023,https://www.br-automation.com/downloads_br_productcatalogue/assets/EPSG_301_V-1-5-1_DS-c710608e.pdf

[RFC 6241] IETF, Network Configuration Protocol (NETCONF), Request for Comments 6241, June 2011, https://datatracker.ietf.org/doc/html/rfc6241

[RFC 8040] IETF, RESTCONF Protocol, Request for Comments 8040, January 2017, https://datatracker.ietf.org/doc/html/rfc8040

[RFC8655] N. Finn, P. Thubert, B. Varga, J. Farkas, "Deterministic Networking Architecture," IETF RFC 8655, October 2019, https://datatracker.ietf.org/doc/rfc8655/

[RJS+23] R. Robert, W. John, J. Sjöberg, J. Halén, "Opportunities with dynamic device offloading as a 6G service," Ericsson blog, September 07, 2023. https://www.ericsson.com/en/blog/2023/9/dynamic-device-offloading-as-a-6g-service

[Roe20] D. Roeland, "An introduction to data-driven network architecture", October 2020, https://www.ericsson.com/en/blog/2020/10/data-driven-network-architecture

[RÖT23] G. Rune, P. Öhlén, Z. Turányi, G. Mildh "Six talking points for architecting the next wireless generation", Ericsson blog, September 13 2023, https://wcm.ericsson.net/en/blog/2023/9/six-talking-points-future-network-architecture

[S20] M Schüngel , Analysis of time synchronization for converged wired and wireless networks," in 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), vol. 1, pp. 198–205, 2020.

[SAA+19] J. Sachs, L. A. Andersson, J. Araújo, C. Curescu, J. Lundsjö, G. Rune, E. Steinbach, G. Wikström, "Adaptive 5G Low-Latency Communication for Tactile InternEt Services," in Proceedings of the IEEE, vol. 107, no. 2, pp. 325-349, Feb. 2019, doi: 10.1109/JPROC.2018.2864587

[SAR+23] M. Saimler, M. D'Angelo, D. Roeland, A. Ahmed, A. Kattepur, "AI as a service: How AI applications can benefit from the network," Ericsson blog, December 14, 2023. https://www.ericsson.com/en/blog/2023/12/ai-as-a-service

[SDL+21] M. Schüngel, S. Dietrich, L. Leurs, D. Ginthör, S.-P. Chen, and M. Kuhn,

"Advanced grandmaster selection method for converged wired and wireless networks," in 22nd IEEE International Conference on Industrial Technology (ICIT), vol. 1, pp. 1007–1014, 2021.

[SK23] D. Schulz, A. Karaagac, "Cutting the cables," ABB review, January 2023, https://new.abb.com/news/detail/99141/cutting-the-cables

[SKM+21] M. Svensson, B. Kovács, E. Mueller, M. Maggiari and R. Szabó, "Service exposure and automated life-cycle management: The Key Enablers for 5G Services," in Ericsson Technology Review, vol. 2021, no. 13, pp. 2-12, December 2021, doi: 10.23919/ETR.2021.9904697 https://ieeexplore.ieee.org/document/9904697

[SPS+23] G. P. Sharma, D. Patel, J. Sachs, M. De Andrade, J. Farkas, J. Harmatos, B. Varga, H. -P., Bernhard, R. Muzaffar, M. Ahmed, F. Duerr, D. Bruckner, E.M. De Oca, D. Houatra, H. Zhang and J. Gross, "Toward Deterministic Communications in 6G Networks: State of the Art, Open Challenges and the Way Forward," in IEEE Access, vol. 11, pp. 106898-106923, 2023, doi: 10.1109/ACCESS.2023.3316605.

[STK+24] E. Semaan, E. Tejedor, R. K. Kochhar, S. Magnusson, S. Parkvall, "6G spectrum - enabling the future mobile life beyond 2030," Ericsson white paper, May 2024, https://www.ericsson.com/en/reports-and-papers/white-papers/6g-spectrum-enabling-the-future-mobile-life-beyond-2030

[SWD+18] J. Sachs, G. Wikstrom, T. Dudda, R. Baldemair and K. Kittichokechai, "5G Radio Network Design for Ultra-Reliable Low-Latency Communication," in *IEEE Network*, vol. 32, no. 2, pp. 24-31, March-April 2018, doi: 10.1109/MNET.2018.1700232.

[Toz16] M. E. Tozal, "The Internet: A system of interconnected autonomous systems," *2016 Annual IEEE Systems Conference (SysCon)*, Orlando, FL, USA, 2016, pp. 1-8, doi: 10.1109/SYSCON.2016.7490628.

[VHC+21] J. Vikberg, G. Hall, T. Cagenius, R. Wang, J. Schultz, "Robustness Evolution: Building robust critical networks with the 5G System," in *Ericsson Technology Review*, vol. 2021, no. 11, pp. 2-12, November 2021, doi: 10.23919/ETR.2021.9904657, https://ieeexplore.ieee.org/document/9904657

[Wik1-25] Wikipedia, "Autonomous system (Internet)," https://en.wikipedia.org/wiki/Autonomous_system_(Internet) (last visited May 14, 2025).

[Wik2-25] Wikipedia, "Integrated services," https://en.wikipedia.org/wiki/Integrated_services (last visited May 14, 2025).

List of abbreviations

5G NSA	5G Non-Standalone
5G SA	5G Standalone
3GPP	3rd Generation Partnership Project
5G-ACIA	5G Alliance for Connected Industries and Automation
5G-Adv	5G-Advanced
5GS	5G System
AES	Advanced Encryption Standard
AF	Application Function
AGV	Automated Guided Vehicle
AI	Artificial Intelligence
AI-aaS	Artificial Intelligence as a Service
AMF	Access & Mobility Management Function
AMR	Autonomous Mobile Robots
AnLF	Analytics Logical Function
API	Application Programming Interface
BTCA	Best timeTransmitter Clock Algorithm
BSS	Business Support Systems
CAMARA	*the Telco Global API Alliance*
CL	Centralized Learning
CN	Core Network
CNC	Centralized Network Configuration
CNI	Container Network Interface
CPS	Cyber-Physical System
CPU	Central Processing Unit
CQF	Cyclic Queuing and Forwarding
CQI	Channel Quality Indicator
CTI	Cyber Threat Intelligence
CUC	Centralized User Configuration
DCCF	Data Collection Coordination Function
DetNet	Deterministic Networking
DiffServ	Differentiated Services
DL	Distributed Learning
DS-TT	Device-Side Time Sensitive Networking Translator
E2E	End-to-End
EAS	Edge Application Server
ECC	Elliptic Curve Cryptography

ECS	Edge Configuration Server
EEC	Edge Enabler Client
ETSI	European Telecommunications Standards Institute
FPGA	Field Programmable Gate Array
FRER	Frame Replication and Elimination for Reliability
GM	Grand Master
GNSS	Global Navigation Satellite Systems
gNB	Next Generation Node B
GPP	General Purpose Processor
gPTP	Generic Precision Time Protocol
GPU	Graphics Processing Unit
HARQ	Hybrid Automatic Repeat Request
IBN	Intent-Based Networking
ICT	Information and Communication Technology
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IMT-2030	International Mobile Telecommunications towards 2030 and beyond
INT	In-band Network Telemetry
IntServ	Integrated Services
IIoT	Industrial Internet of Things
IP	Internet Protocol
ITU	International Telecommunication Union
KPI	Key Performance Indicator
KVI	Key Value Indicator
LAN	Local Area Network
LIDAR	Light Detection and Ranging
MDAF	Management Data Analytics Function
MEC	Multi-Access Edge Computing
ML	Machine Learning
MPM	Mobile Processing Module
MRSS	multi-RAT spectrum sharing
MTD	Moving Target Defence
MTLF	Model Training Logical Function
NE	Network Element
NEF	Network Exposure Function
NETCONF	Network Configuration Protocol
NF	Network Function
NFV	Network Function Virtualization
NIC	Network Interface Card

NIST	National Institute of Standards and Technology
NPN	Non-Public Networks
NRM	Network Resource Manager
NWDAF	Network Data Analytics Function
NW-TT	Network-side Time Sensitive Networking Translator
OAM	Operations, Administration, and Maintenance
OE	Occupational Exoskeleton
ODA	Open Digital Architecture
OPC UA	Open Platform Communications Unified Architecture
OS	Operating System
OSI	Open Systems Interconnection
OSS	Operational Support Systems
OVS	Open vSwitch
PD	Packet Delay
PDC	Packet Delay Correction
PDV	Packet Delay Variation
PLC	Programmable Logic Controller
PNI-NPN	Public Network Integrated NPN
PTP	Precision Time Protocol
QoS	Quality of Service
RAN	Radio Access Network
RCA	Root Cause Analysis
RESTCONF	Representational State Transfer Configuration Protocol
SDN	Software-Defined Networking
SEAL	Service Enabler Architecture Layer for Verticals
SLA	Service Level Agreement
SMF	Session Management Function
SNPN	Standalone NPN
SO	Security Orchestrator
SSLA	Security Service Level Agreement
TAPRIO	TSN Time-aware Priority Shaper
TC	Transparent Clock
TDA	Time-Delay Attack
TS	Technical Specification
TSC	Time-Sensitive Communication
TSCTSF	Time-Sensitive Communication and Time Synchronization Function
TSN	Time-Sensitive Networking
TSN AF	Time-Sensitive Networking Application Function

TT	TSN Translator
UAFX	Unified Architecture Field eXchange
UAV	Unmanned Aerial Vehicle
UE	User Equipment
UGV	Unmanned Ground Vehicle
UPF	User Plane Function
URLLC	Ultra Reliable and Low Latency Communications
VM	Virtual Machine
VPN	Virtual Private Network
WP	Work Package
XR	eXtended Reality
YANG	Yet Another Next Generation data modelling language
ZSM	Zero-touch network and Service Management
ZTN	Zero-Trust Networking