D4.2

First report on PREDICT-6G system validation

SIM

Funded by
the European Union

Revision v0.1

Work package	WP4
Task	T4.1, T4.2, T4.3
Due date	30-09-2024
Submission date	30-09-2024
Deliverable lead	[SIM]
Version	v1.0
Authors	Sebastian Robitzsch, Abinaya Babu, Muhammad Awais Jadoon, Filipe Conceicao (IDE) Péter Szilágyi, Tamás Kárász, Szabolcs Nováczki, Zoltán Vincze, Csaba Vulkán (NOK) Marc Molla, Rocio Dominguez, Miguel Angel Lopez (ERC) Constantine Ayimba, Alejandro Calvillo (UC3M) Otilia Bularca, Iacob Crucianu (SIM) Juan Brenes (NXW) José Luis Cárcel (ATOS) Fernando Agraz, Salvatore Spadaro (UPC)
Reviewers	CLAUDIO ZUNINO (CNR) ANTONIO DE LA OLIVA DELGADO (UC3M)

Abstract

This deliverable will provide the first version of the PREDICT-6G system prototype integrating the initial outcomes ofWP2 and WP3 and deployed in the two Open labs, with the related documentation.

Keywords

Lab environments, NOKIA Lab, 5TONIC Lab, PoC, integration, test, verification, validation, roadmap, AICP, MDP, TSN, 3GPP

Document revision history

Version	Date	Description of change	Contributor(s)
v0.1	18-06-2024	Table of contents	Otilia Bularca (SIM) Iacob Crucianu (SIM)
V02	22-07-2024	Table of contents re-structured	Otilia Bularca (SIM) Peter Szilagyi (NOK) Juan Brenes (NXW)
V03	06-09-2024	First round of contributions consolidated version	All
V03	10-09-2024	Localisation and Sensing PoC integration description and Data Unit Group Validation experiments	Sebastian Robitzsch, Abinaya Babu, Filipe Conceicao, Muhammad Awais Jadoon (IDE)
V04	20-09-2024	Version ready for internal review and internal review suggestions and recommendations	Otilia Bularca (SIM) CLAUDIO ZUNINO (CNR) ANTONIO DE LA OLIVA DELGADO (UC3M)
V05	29-09-2024	Version including comments and suggestions from reviewers	Otilia Bularca (SIM)
V10	30-09-2024	Version ready for submission	ANTONIO DE LA OLIVA DELGADO (UC3M

Disclaimer

The information, documentation and figures available in this deliverable are provided by the PREDICT-6G project's consortium under EC grant agreement **101095890** and do not necessarily reflect the views of the European Commission. The European Commission is not liable for any use that may be made of the information contained herein.

Copyright notice

Document information

Nature of the deliverable	[R *]	
Dissemination level		
PU	Public, fully open. e.g., website	✓

R: document, report (excluding periodic and final reports)..

Table of contents

Funded by
the European Union

List of figures

Funded by
the European Union

Funded by
the European Union

List of tables

Funded by
the European Union

Acronyms and definitions

AI	Artificial Intelligence
AICP	AI-driven Multi-stakeholder Inter-domain Control-Plane
AMF	Access and Mobility Management Function
API	Application Programming Interface
CUPS	Control and User Plane Separation
DoA	Description of Action
DPDK	Data Plane Development Kit
DT	Digital Twin
E2E	End-to-End
eMBB	enhanced Mobile BroadBand
gNB	Next Generation Node B
IP	Internet Protocol
KPI	Key Performance Indicator
QoS	Quality of Service
MDP	Multi-domain Data-Plane
MBH	Mobile Backhaul
ML	Machine Learning
NIC	Network Interface Card
PDU	Power Distribution Unit
PM	performance measurement
PMD	Poll Mode Drivers
PoC	Proof of Concept
RAN	Radio Access Network
SA	Stand Alone
SMF	Session Management Function
TAP	Test Access Point
TD	Technology domain
UPF	User plane function
VM	Virtual Machine
WP	Work Package

Table of partners

Short Name	Partner
UC3M	Universidad Carlos III de Madrid
NOK	Nokia Solutions and Networks KFT
ERC	Ericsson Espana SA
INT	Intel Deutschland GMBH
TID	Telefonica Investigacion Digital SL
ATOS	ATOS IT Solutions and Services Iberia SL
GES	Gestamp Servicios SA
NXW	Nextworks
COG	Cognitive Innovations Private Company
SIM	Software Imagination & Vision SRL
AUSTRALO	AUSTRALO Alpha Lab MTU
POLITO	Politecnico di Torino
UPC	Universitat Politecnica de Catalunya
CNR	Consiglio Nazionale delle Ricerche
UNIPD	Universita degli Studi di Padova
IDE	td

1 Executive summary

This report provides an update of the integration activities focusing on the PREDICT-6G components and functionalities available or testing.

The testing and validation methodology was defined in previously released deliverables and followed here using templates to provide a similar approach in describing the progress on each PREDICT-6G open labs and their subsequent proof of concepts. Based on the roadmap defined in D4.1 [1] the progress at the end of Cycle 3 is detailed in this report.

This integration methodology leverages the interfaces, and workflows defined in WP1 and is based on the definition two different level of tests: (i) infrastructure integration test in which the aim is to validate the correct implementation of a specific of the generic interface by the MDP functionality; (ii) E2E use case validation tests and activities focus on the KPIs and functional validation of the AICP platform.

The first version of the PREDICT-6G integrated prototype and the initial outcomes of WP2 and WP3 tools deployed in the two Open Labs are included in this report.

PREDICT-6G relays in two open laboratories to perform the integration and testing of the innovations developed: i) the Madrid Open Lab (5TONIC) and ii) the Budapest Open Lab. Both laboratories provide an open experimental platform based on commercial and non-commercial hardware, although each of the labs has a special focus.

For each open lab an environment description was provided alongside with infrastructure components, while also adding tests at different levels and results report for available components and functionalities.

An updated version with the components & functionalities in progress at the moment of submitting this report will be included in D4.3 due at the end of the project (M30)

2 Testbed setup, deployment and validation

PREDICT-6G relays in two open laboratories to perform the integration and testing of the innovations developed: i) the Madrid Open Lab (5TONIC) and ii) the Budapest Open Lab. Both laboratories provide an open experimental platform based on commercial and non-commercial hardware, although each of the labs has a special focus.

The testing capabilities of the project are extended through the capabilities provided by the Budapest Open Lab. This lab is a private 5G standalone (SA) network deployed to develop, test, and validate industrial use cases supported by network and service automation technology. The network can be used as a sandbox for disruptive research due to no SLA or commercial operation obligations. The network can be integrated with any platform (e.g., traffic generators, AI/ML platforms), either hosted by its own local edge cloud, or connected to any cloud (e.g., other 5G Non-Public Network (NPN) or public cloud).

A great advantage of the Open Lab in Budapest is the possibility to inject any kind of data, control and management-plane traffic, network and service management actions, configuration directly into the network itself, as it provides open access to every Network Function (NF) and interface. These attributes make the Open Lab ideal to research and validate network management automation innovations.

5TONIC is composed of two main sites, the IMDEA Networks site provides access to a commercial R16 RAN and core by Ericsson. This infrastructure supports the most common 5G services: enhanced Mobile Broadband (eMBB), massive Machine Type Communication (mMTC) and Ultra-reliable Low Latency Communications (URLLC). Those services are supported by dedicated network slices to meet the requirements of each service. The lab provides access to below 6GHz and mmWave 5G variants.

In addition, the UC3M site in 5TONIC provides a complete open-source alternative. It provides access to a full Open-Air deployment based on USRPs N310 with up to 100MHz of bandwidth. In addition, this site has deployed a commercial TSN network, based on RELYUM hardware switches, in-house developed IEEE 802.1 Qur implementation, and IEEE 802.11 wireless TSN based on Intel developments.

In the following subsections each lab environment is described in terms of network infrastructure and available equipment alongside with the description of the proposed use cases (proof of concepts)

For each proof of concept (PoC) the approaches on AICP and MDP implementation are elaborated, and a list of integration and verification tests is presented.

Funded by
the European Union

2.1 NOKIA Open Lab

The implementation and integration of selected PREDICT-6G system components in the Nokia Open Lab environment aim at enabling the automated setup and management of deterministic services across technology domains with no native deterministic capabilities. The section will first describe the lab environment and then it will elaborate on the AICP and MDP implementation aspects. The implemented AICP enables the dynamic and automated management of deterministic services whereas the implemented MDP Agent delivers means to provide deterministic services over natively non-deterministic segments.

2.1.1 Description of the lab environment

The lab environment features two network domains (5G SA network and programmable fixed IP network), server infrastructure and end user devices (Figure 2-1).

5G SA network

The indoor private 5G Stand Alone (SA) network is composed of 5G base stations (gNBs) that act as the Radio Access Network (RAN) and a 5G Core Network (5G CN) running on a cloud infrastructure (Figure 2-1). The 5G Core Network implements all the important functions required for the 5G network operation (e.g., UPF, SMF, AMF). It also provides API for exposure of topology, capability and measurement information. Besides exposing network information, there is also API different managing settings in the network (e.g., setting the QoS of the PDU sessions of the user equipment). The 5G SA network provides non-deterministic services.

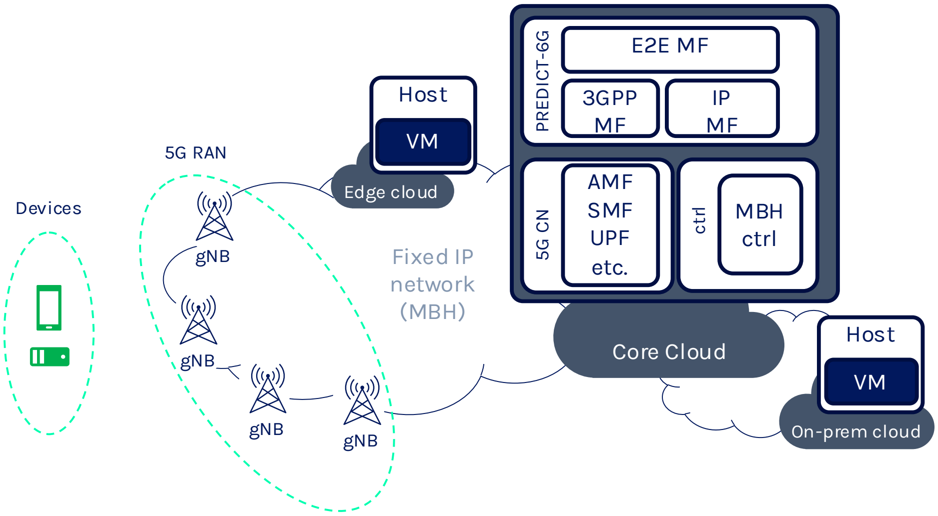


Figure 2-1. Laboratory setup in Nokia Open Lab

Fixed IP network

The RAN and the 5G CN is connected by a fixed IP network acting as the Mobile Backhaul (MBH) network of the 5G SA network (Figure 2-1). The IP network provides non-deterministic services over a multi-link network topology with multiple disjunct paths between the 5G base stations and the 5G CN allowing for dynamic path selection. The topology, capability and resource exposure APIs and measurement/monitoring API for the IP network is provided by the MBH controller software module.

Server infrastructure

The laboratory setup includes multiple servers composing a Core cloud, On-premises cloud and Egde clouds as shown in Figure 2-1. The core cloud is responsible for running the 5G Core network, the controller of the programmable fixed IP network connecting the 5G RAN and 5G Core network. It also hosts the PREDICT-6G AICP components (both domain and end-to-end managed services). The On-premises cloud and Edge clouds can host any software components, e.g., applications.

The servers are general purpose computing servers with 80 vCPUs and 384 GB RAM. They provide a virtualized environment for running VMs or cloud native SW components in Kubernetes.

Devices

There are two main device types available in the laboratory setup: (1) 5G capable devices (e.g., smartphones running Android operating system) and (2) non-5G capable ones (e.g., minicomputers). The minicomputers run Linux operating system and can host MDP Agents making it possible to extend the deterministic service till the end user devices. The non-5G capable devices can be connected to the 5G SA network via the tethering over the 5G capable devices.

Technology domains

The Nokia Open Lab setup has two main technology domains (1) 3GPP 5G and (2) fixed IP network. Both domains provide non-deterministic services natively, however the MDP Agent (see section 2.1.4) implemented in the Nokia Open Lab provides features that make it possible to ensure deterministic services over these non-deterministic domains. In the Nokia Open Labs setup this is achieved by placing the MDP Agent on the N3 interfaces of the 5G SA network. Similarly, the MDP Agent based approach is extendable to any other technology domain which are not present in the Nokia Open Lab setup by placing the MDP Agent in the right locations.

Funded by
the European Union

2.1.2 Description of the PoC

The subsection describes on high level the proof-of-concept that will be showcased using the PREDICT-6G technology in the Nokia Open Lab.

- **Summary:** automated management of deterministic services for critical communications over domains with no native deterministic capabilities.
- **Goal:** showcase that using the PREDICT-6G MDP and AICP components implemented in Nokia Open Lab, deterministic service can be provided over network domains without native deterministic capabilities.
- **Tests:** integration of selected MDP and AICP components in Nokia Open Lab, automated setup, assurance, modification and release of deterministic service over 5G SA and fixed IP network domains with
- **KPIs:** jitter, latency, packet loss
- **Components:** 5G SA network (RAN, Core), fixed IP network, server infrastructure (core, edge, on-prem cloud), 5G capable and non-5G capable end devices
- **Technology domains:** 3GPP, IP without deterministic capabilities

2.1.3 Implementation of the AICP

This subsection will describe the various AICP components implemented and integrated in the Nokia Open Lab for creating a system that is able to setup and maintain deterministic services across multiple network segments with no native deterministic capabilities. Before introducing the domain level and end-to-end level AICP components of the Nokia Open Lab system, two general AICP related concepts are introduced: (1) intent-based service management and (2) digital twinning. These concepts have great impact on the actual implementation of the PREDICT-6G system in the lab, thus their description helps to better understand the setup.

General concepts

Intent based approach. In this approach the user of the PREDICT-6G system is interacting with the user interface of the intent-based service ingestion and assurance (IBS) module (Figure 2-2. The module provides means for the user to interact with the system on a higher level without the need for deep technical knowledge. The user can select the possible service end point locations, service types and receives feedback on the status of the requested service. The IBS translates the intent of the user to actual configuration and requests the service with the determined configuration via the E2E service ingestion MS. It also follows-up the status of the requested service and in case it detects that PREDICT-6G AICP cannot assure the requested service it may requests service modification.

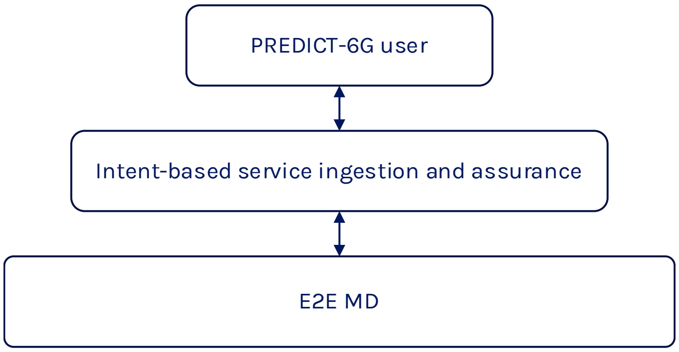


Figure 2-2: Intent-based service management

Digital Twinning. Digital Twins (DT) are integral part of the PREDICT-6G system implemented in Nokia Open Lab. A dynamic, modular approach is used for creating and building the DTs, i.e. instead of using one global DT for the whole system, multiple purpose-built DTs with different scope (e.g., per domain, E2E DTs) and level of granularity are populated based on the actual requirements and state of the system.

There are two main data sources for DTs, (1) static or descriptive (including topology, capabilities, configuration) data and (2) dynamic or functional data (measurements) describing the actual state of the network. Using the data, predictive DTs are built in the system which can be dynamically extended with new data sources. The predictive DTs are used by the E2E and domain service ingestion MSs for “what-if” kind of analysis purposes. The analysis may investigate scenarios like is it possible accommodate a given service between two end points or would the service still work as expected if two new users arrive to the system, etc.

AICP components in Nokia Open Lab

The PREDICT-6G system in Nokia Open Lab implements the subset of the full reference AICP architecture (D1.2, section 5 [2]) managed services which required for the automated and dynamic management of deterministic services. Figure 2-3 shows the E2E and domain specific AICP components implemented and integrated in Nokia Open Lab. It shall be noted that though the MSs are considered as logically separate services, their implementation can be different, i.e., in some cases one software component implements a single MS whereas in other cases several MSs can be combined into one software component, e.g., the MDP Agent implements multiple MSs like measurement collection, resource exposure and configuration, etc.

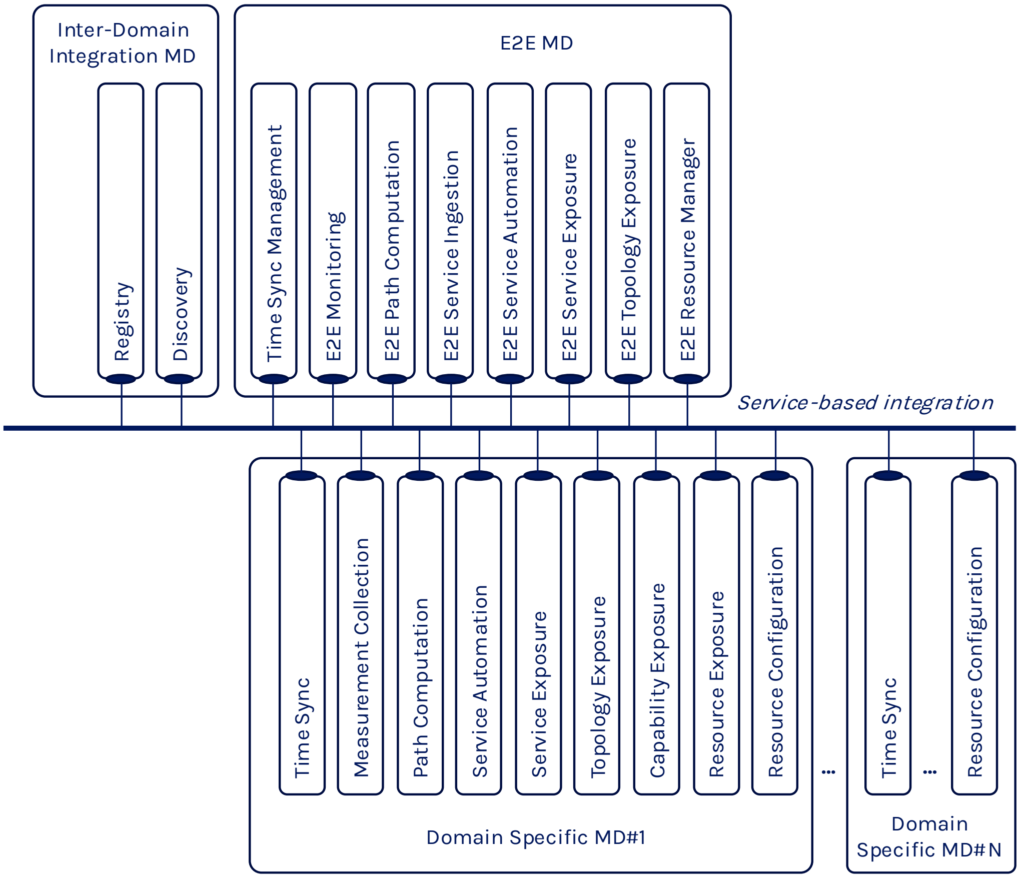


Figure 2-3: AICP components in Nokia Open Lab

Domain level MS. These managed services are responsible for domain level operation and implement the operation defined in D1.2 [2] and D3.1 [3]. The list of the implemented domain level MSs are:

- Time Sync
- Measurement Collection
- Path Computation
- Service Automation
- Service Exposure
- Topology Exposure

- Capability Exposure
- Resource Exposure
- Resource Configuration

E2E management services

These managed services are responsible for E2E level operation and implement the operation defined in D1.2 [2] and D3.1 [3]. The list of the implemented E2E MSs are:

- Registry, Discovery
- E2E Time Sync Management
- E2E Monitoring
- E2E Path Computation
- E2E Service Ingestion
- E2E Service Automation
- E2E Service Exposure
- E2E Resource Manager

System procedures

The system procedures of a PREDCT-6G system are described in D1.2, section 9.1 [2]. In Nokia Open Lab two of those procedures were implemented, MS registration & deregistration, discovery and cross-domain time sync.

MS registration & deregistration, discovery. This procedure ensures that the MS modules can be dynamically added/removed in the system. Upon the start of a new MS, it uses the MS registration procedure as described in D1.2, section 9.1 [2]. to register itself in the system. Via the discovery mechanism, the MSs can discover the required services and connect to them automatically.

Cross-domain time sync. Cross-domain time sync system procedure ensures that the time synchronization is harmonized in the domains of the PREDICT-6G system, thus allowing for proper time-aware operation in the E2E system.

Service management procedures

D1.2, section 10 [2]. describes the service management procedures of a PREDICT-6G system. The Nokia Open Lab implements three of those procedures.

Deterministic e2e service request. When the user requests a deterministic service from the PREDICT-6G system, the E2E Service ingestion MS triggers the procedure to automatically create the e2e deterministic service.

Deterministic e2e service release. When the service is not needed anymore (either due to user request or it is a scheduled service and its timing expired) the E2E Service Ingestion triggers the procedure to remove the E2E deterministic service from the system.

Deterministic e2e service assurance. Upon the detection of the violation of deterministic service requirements (e.g., QoS characteristics), or when the DT model predicts the violation of the requirements, the E2E service assurance procedure is automatically triggered in the system to take the necessary corrective actions.

2.1.4 Implementation (integration) of the MDP

The MDP implementation in the Nokia Open Lab focused on providing means for improved determinism over non-deterministic domains.

Determinism in non-deterministic domains

The technology domains in the Nokia Open Lab (3GPP and IP) provide only non-deterministic data plane features. To introduce deterministic capabilities over these domains the MDP Agent software module was designed and implemented (Figure 2-4). This approach combines the non-deterministic data plane features of the domain with the deterministic data plane features provided by the MDP Agent for improved determinism in natively non-deterministic networks. This requires that the management of the non-deterministic and deterministic data plane features are harmonized, i.e., the non-deterministic data plane features are configured so that their operation matches the best possible way the requirements of the deterministic service. E.g., the 5G SA network is configured so that the end device's PDU session has the highest possible QoS settings providing a good basis for the MDP Agent's operation to further improve the service quality to match the deterministic requirements.

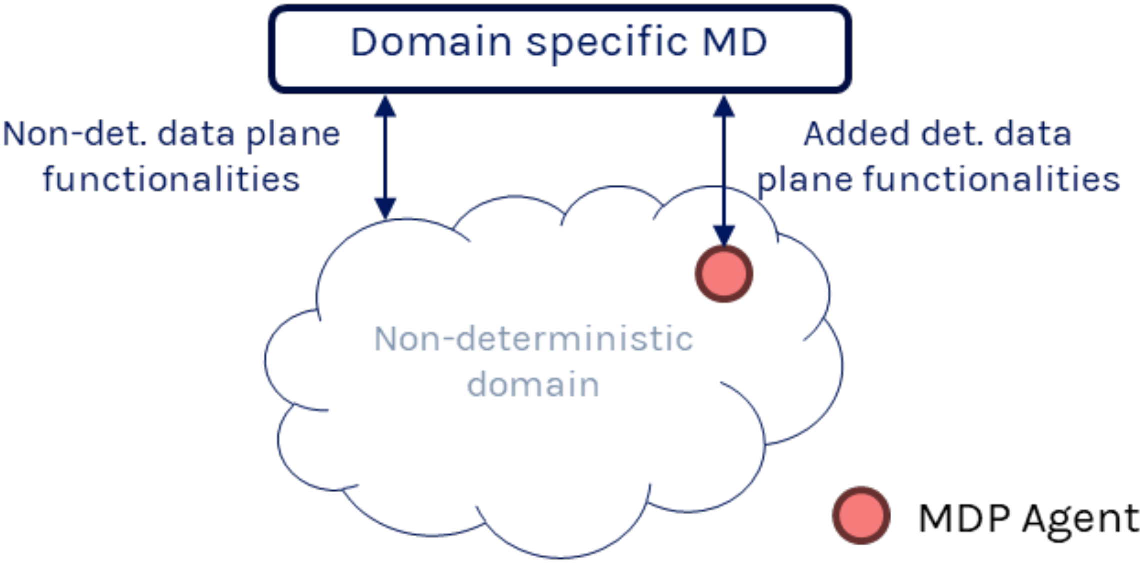


Figure 2-4: Combining non-deterministic and deterministic data plane capabilities

MDP Agent

The functionalities required to achieve providing deterministic services over non-deterministic domains (e.g., time synch, scheduling, etc.) were implemented as software modules and are compiled into one entity called MDP Agent. Figure 2-5 shows the architecture of the MDP Agent.

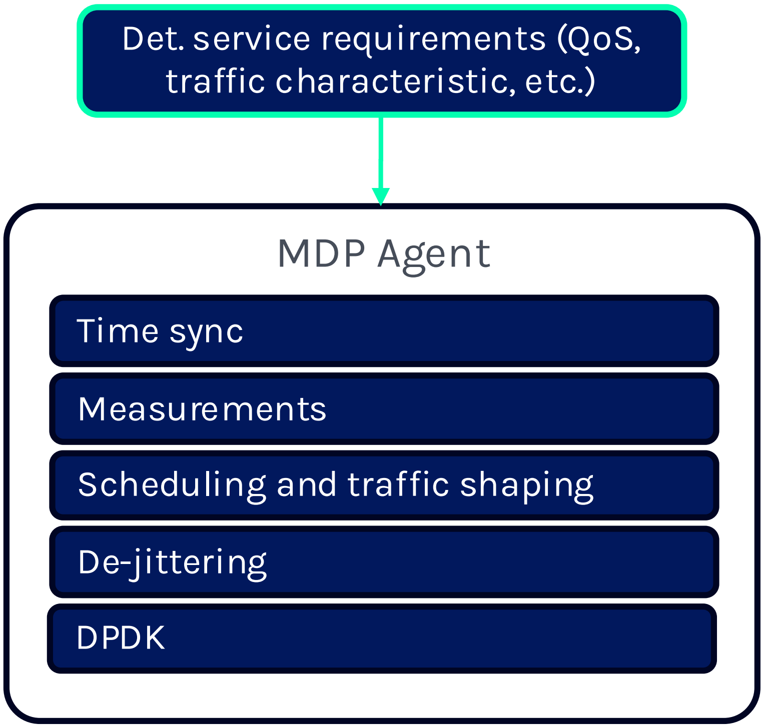


Figure 2-5: Architecture of the MDP Agent

The MDP Agent is an in-line application that processes user plane traffic with line speed and applies a configurable set of functionalities on the processed user plane packets to achieve determinism. The functionalities implemented in the MDP Agent are described on high level in the following.

Time sync

Deterministic operation requires that the devices participating in real time communication are time aware and use a harmonized time synch setup. The time synch functionality provides this by implementing a gPTP module which can act as slave clock and is able to synch to the grand master clock of the PREDICT-6G system. The MDP Agent configures its packet forwarding to support the synchronization plane by providing priority/reservation for gPTP packets.

DPDK

MDP Agent is deployed in the user plane traffic flow, thus it shall process the packets with line speed. To achieve this, MDP Agent uses Data Plane Development Kit (DPDK), which provides a user-space programming environment for high-performance packet processing by bypassing the operating system's kernel networking stack and allowing direct access to NICs from user space via Poll Mode Drivers (PMD).

Measurement

The MDP Agent performs various measurement actions using the information available from DetNet encapsulation: identification/classification, timestamping and calculating delays and jitters, sequence numbering (for packet loss detection), throughput measurement. The Agent reports the measurements towards the Measurement Collection MS of the domain specific PREDICT-6G MD.

Scheduling and traffic shaping

This functionality uses the scheduling hierarchy of the DPDK and creates/deletes buffers within a scheduler hierarchy dynamically based on the deterministic services. It classifies the packets to buffers and enforces the scheduling paradigm (time and traffic characteristics awareness, allow burst sizes, etc.).

De-jittering

This process leverages timestamps/calculated delays/jitters derived by the measurement actions to adaptively adjust the hold time based on actual jitter and QoS target to cause minimum delay.

MDP Agent deployment options

Deployment scenarios range from single MDP Agent in a domain to multiple MDP Agents in a domain. Measurement actions and packet replication/elimination requires collaboration among MDP Agents, thus those features can be used in case of multiple MDP Agents per domain.

Funded by
the European Union

Figure 2-6 shows an example how MDP Agents can be deployed in a 5G SA network. They can be added on the N3 interfaces of the 5G network enabling all the features of the MDP Agent. Moreover, the MDP Agent can be added on or next to the end device which extends the determinism toward the end devices.

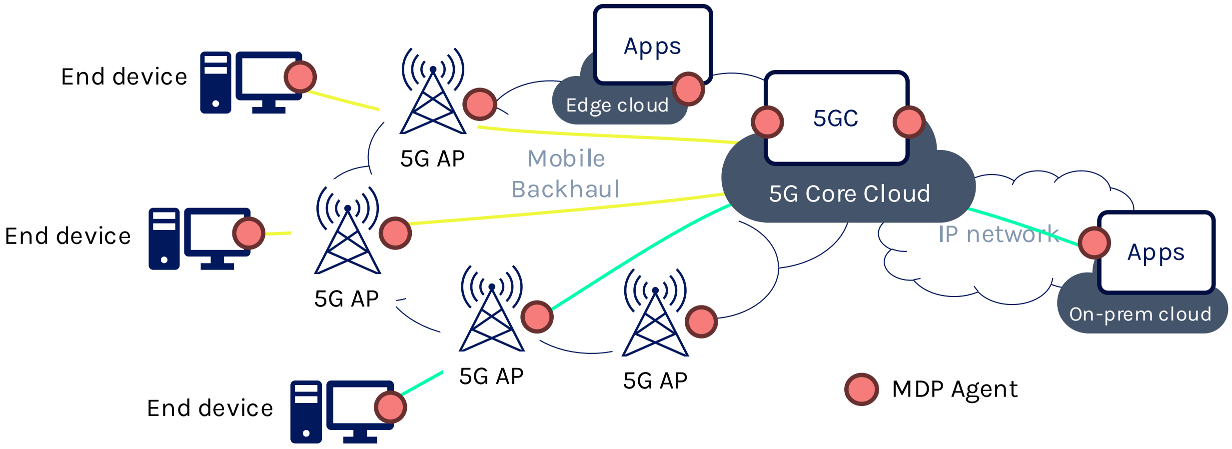


Figure 2-6: Example MDP Agent deployment scenario for 5G SA network

2.1.5 System integration and verification

The implementation and the integration of the components of the PREDICT-6G system in the Nokia Open Lab follows the principles and specifications set forth in D1.2 [2], D3.1 [3]. It is a service-based architecture. where MSs can interact with each other using HTTP/JSON based interfaces and domain level MSs interact with the technology domains using technology specific interfaces.

Through the integration of the components, the PREDICT-6G system in the Nokia Open Lab can automatically create, assure, and release E2E deterministic services in a multi-domain setup including non-deterministic domains. The system is able collect topology, capability information and measurements from the domains. The system allows for the user to request deterministic service between the selected end points and to define deterministic requirements. The SW components necessary to create the E2E deterministic service over non-deterministic domains are dynamically initialized, configured and deployed in the user plane.

The next subsections will describe the E2E time sync and E2E monitoring integrations.

Funded by the European Union

E2E integration of time synch service

As Figure 2-7 shows, the Nokia Open Lab has three-time synch domains in: (1) the 5G SA network with gPTP clock, (2) a ToP dongle with gPTP clock plugged into a router and (3) the MDP Agents implementing gPTP functionalities.

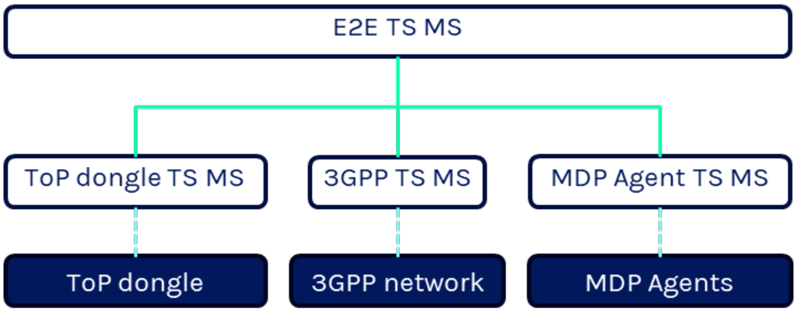


Figure 2-7: Integration of the time synch services

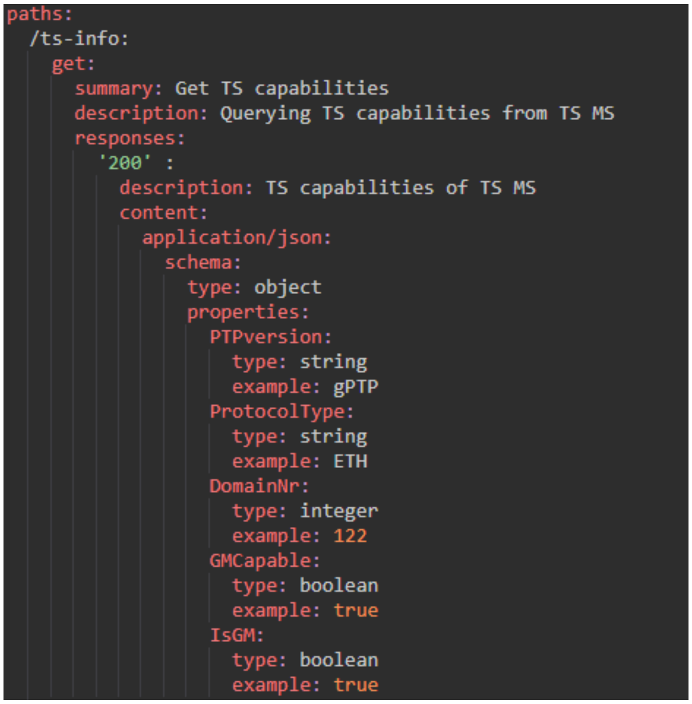
As defined in D1.2, section 9.4 [2]. in the first step of the cross-domain time synchronization procedure, the E2E TS MS queries the domain time synch MSs about their time synch capabilities using the TS OpenAPI (Figure 2-8). Upon receiving the query from the E2E TS MS, the domain TS MSs query the technology domains for the required information using technology specific interfaces. In the Nokia Open Lab time synch setup, the MDP Agents do not have Grand Master (GM) capable clock, thus they can only act as GM followers. The 3GPP domain may act GM or as follower, whereas the ToP dongle can act as GM only. Once this information is collected by the domain TS MSs, they send this to the E2E TS MS in the body of the response sent to the capability query request. Based on the information collected from the domain MSs, the E2E MS decides that 3GPP and MDP domain will act as follower and the ToP dongle will act as GM in the setup. The E2E MS TS uses the clock configuration OpenAPI (Figure 2-9) to ensure that the clock configuration in the different domains are aligned. At the end of the procedure, the cross-domain clock configuration in the Nokia Open Lab setup is configured correctly and works as expected.

Funded by
the European Union

Parameter	**Description**
ID	TIME_SYNC_INTEGRATION
Description	The integration test shows that using the Time Synch OpenAPI can setup a cross-domain time synch setup.
Management domain services	E2E Time Synch Management MS, Time Sync MS
Technology domain (TD)	3GPP, ToP dongle, MDP Agent
Specific MD Interface & methods	Time Synch OpenAPI
Result	SUCCESSFUL At the end of the test, the cross-domain clock configuration in the Nokia Open Lab setup is configured correctly and works as expected.
Date	2H/2024
Observations and comments	No observations

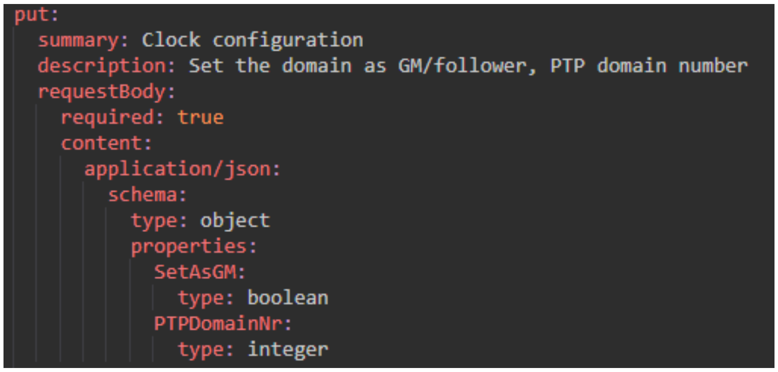
Table 2-1: E2E Time Synch integration in Nokia Open Lab

Funded by the European Union



```
paths:
  /ts-info:
    get:
      summary: Get TS capabilities
      description: Querying TS capabilities from TS MS
      responses:
        '200' :
          description: TS capabilities of TS MS
          content:
            application/json:
              schema:
                type: object
                properties:
                  PTPversion:
                    type: string
                    example: gPTP
                  ProtocolType:
                    type: string
                    example: ETH
                  DomainNr:
                    type: integer
                    example: 122
                  GMCapable:
                    type: boolean
                    example: true
                  IsGM:
                    type: boolean
                    example: true
```

Figure 2-8: Open API for TS capabilities query



```
put:
  summary: Clock configuration
  description: Set the domain as GM/follower, PTP domain number
  requestBody:
    required: true
    content:
      application/json:
        schema:
          type: object
          properties:
            SetAsGM:
              type: boolean
            PTPDomainNr:
              type: integer
```

Figure 2-9: Open API for TS configuration

E2E monitoring

Figure 2-10 shows the E2E monitoring setup integrated in the Nokia Open Lab. The measurement collection system uses MQTT, Telegraf and influxDB components for implementing the necessary functions both on domain and E2E level. The technology domains report the measurements to respective domain MSs, e.g., the 3GPP network and the related MDP Agents send measurements to the 3GPP Measurement Collection MS through the technology specific APIs. The measurement reports received by an MS are forwarded to the MQTT broker and using the Telegraf agent it is finally sent to the influxDB.

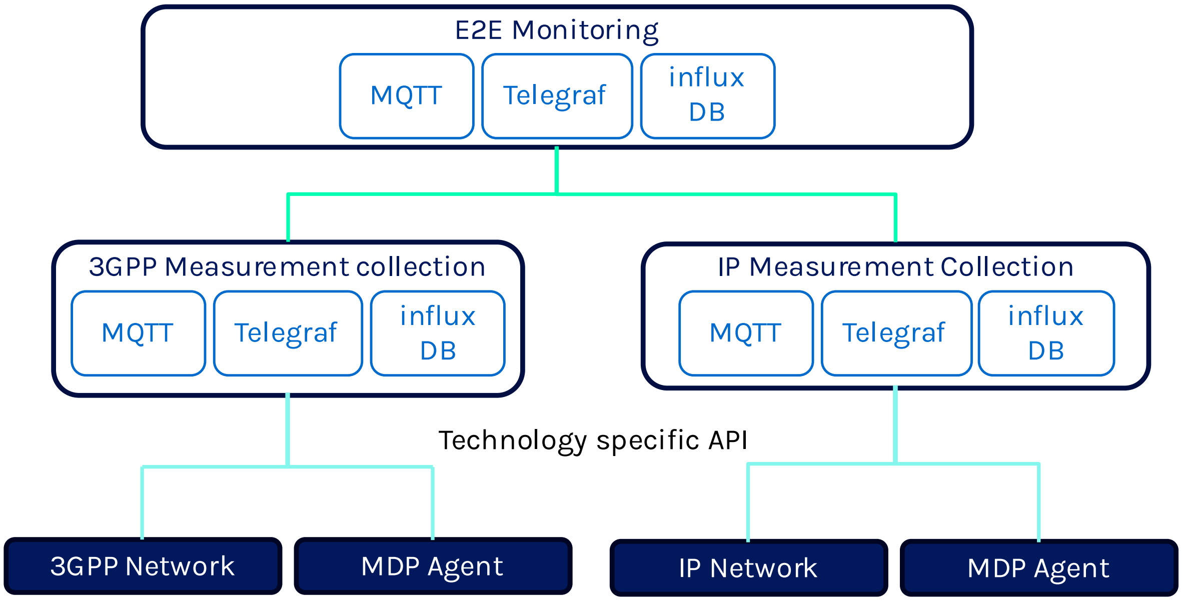


Figure 2-10: E2E monitoring integration

After the measurements are stored in the influxDB, depending on the configuration and requirements, the domain level MS either forwards the data to the E2E Monitoring MS or it further processes it, e.g., creating derived KPIs or aggregated measurements which can be forwarded to the E2E Monitoring MS when it is scheduled or requested directly. Figure 2-11 shows an example throughput measurement from the Nokia Open Lab PREDICT-6G system collected via the implemented measurement collection setup.

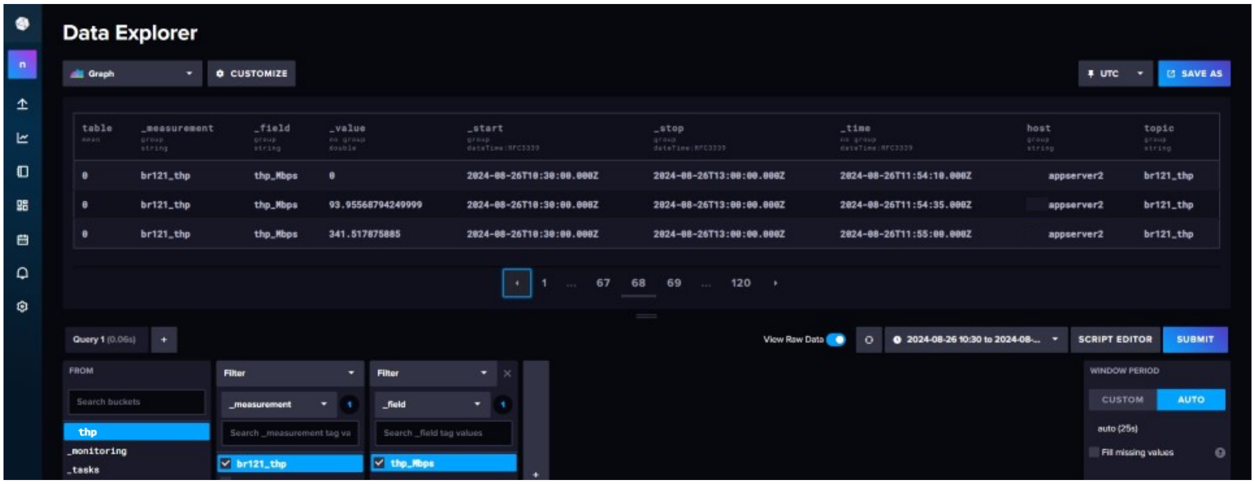


table mean	_measurement group string	_field group string	_value no group double	_start group dateTime:RFC3339	_stop group dateTime:RFC3339	_time no group dateTime:RFC3339	host group string	topic group string
0	br121_thp	thp_Mbps	0	2024-08-26T10:30:00.000Z	2024-08-26T13:00:00.000Z	2024-08-26T11:54:10.000Z	appserver2	br121_thp
0	br121_thp	thp_Mbps	93.95568794249999	2024-08-26T10:30:00.000Z	2024-08-26T13:00:00.000Z	2024-08-26T11:54:35.000Z	appserver2	br121_thp
0	br121_thp	thp_Mbps	341.517875885	2024-08-26T10:30:00.000Z	2024-08-26T13:00:00.000Z	2024-08-26T11:55:00.000Z	appserver2	br121_thp

Figure 2-11: Throughput measurement in Nokia Open Lab setup

Parameter	**Description**
ID	E2E_MONITORING_INTEGRATION
Description	The integration test shows that the setup is able to collect the measurements from the domains via domain specific interfaces and report it to the E2E Monitoring MS via the measurement collection API.
Management domain services	E2E Monitoring MS, Measurement Collection MS
Technology domain (TD)	3GPP, IP
Specific MD Interface & methods	Measurement collection API
Result	SUCCESSFUL The integrated setup was able to collect domain level and E2E level measurements.
Date	2H/2024
Observations and comments	No observations

Table 2-2. E2E monitoring integration test in Nokia Open Lab

Funded by the European Union

2.2 5TONIC Lab

2.2.1 Description of the lab environment

Infrastructure and technologies provided by Ericsson in 5TONIC5TONIC Laboratory:

3GPP 5G system SA:

Ericsson provides the Stand Alone 5G System that supports the experimentation at 5TONIC5TONIC laboratory.

- **5GS infrastructure:**

RAN. The **main** area of coverage is the experimentation room (**X3**, showed at Figure 2-13), which is the main experimentation facility of 5TONIC5TONIC, and it is where the concurrent experiments are executed. The goal of this area is not only to have access to 5G technology but also to have access to the experimentation tools like for example the KPI framework or the graphical dashboard of the experiments.

In this experimentation room it is deployed indoor and outdoor antennas for covering the experimentation facility as well as the outdoor space of 5TONIC.

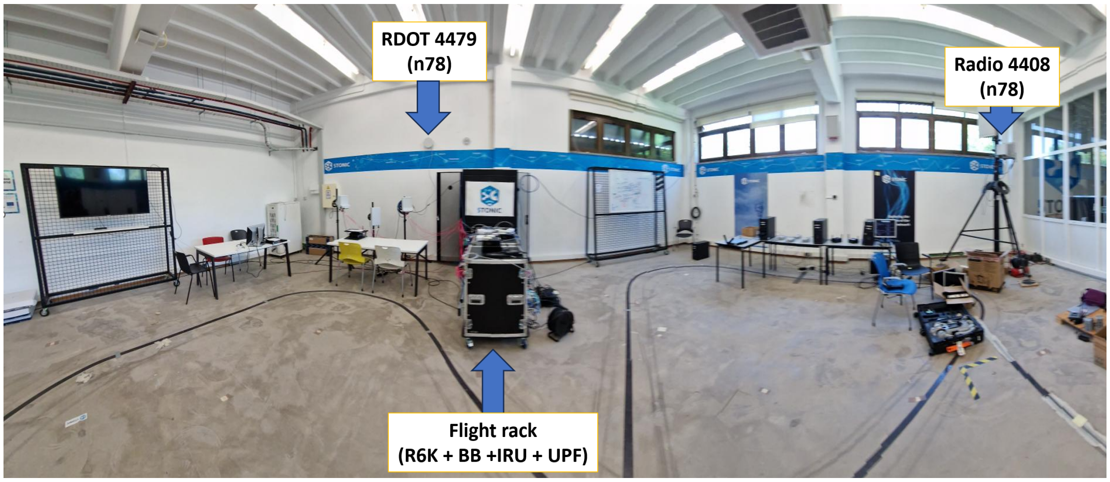


Figure 2-12. 5TONIC experimentation facility

The antennas provide mid-band (3,5 GHz) 5G NR SA coverage at band n78. The below diagram (Figure 2-13) also shows the GNSS active antenna, used for time synchronization of the RAN equipment, including gNB and routers.

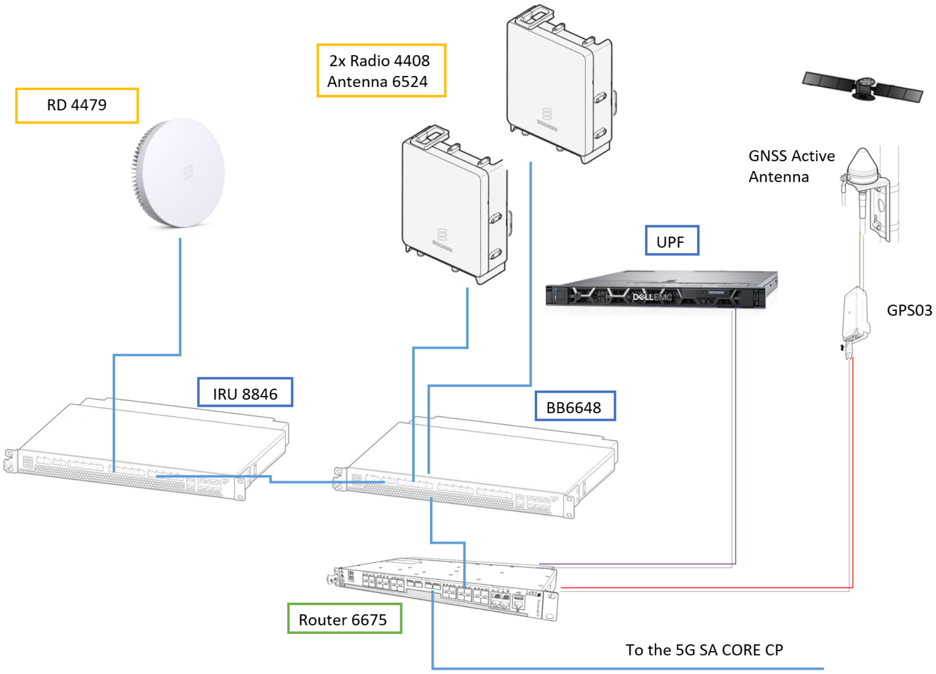


Figure 2-13. Diagram of X3 5G solution at 5TONIC

5G Core. The 5G Core is deployed at the 5TONIC datacenter and it contains the basic Network Functions required for supporting the 5G System:

- Basic NF: NRF, NSSF
- Subscriber NF: UDM, UDR
- Control NF: AMF, SMF, PCF
- User Plane NF: UPF
- Exposing: NEF

The 5G Core is the cloud-native version of Ericsson 5GC and runs on top of a Kubernetes deployment.

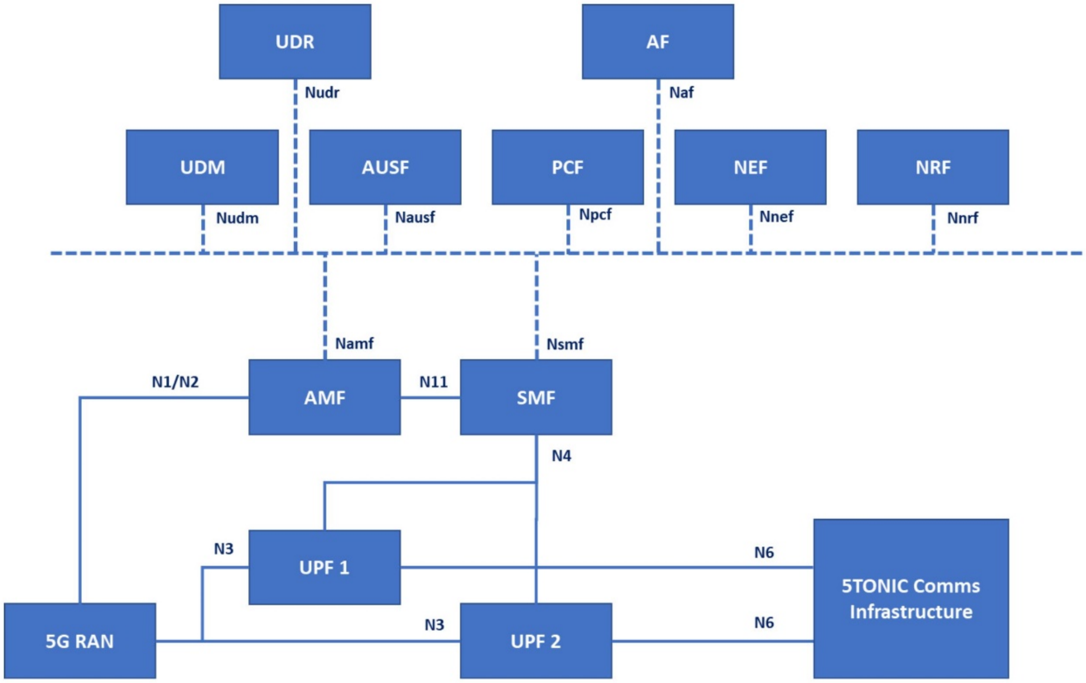


Figure 2-14. Diagram of 5G Core solution at 5TONIC

Portable System. Ericsson developed at 5TONIC a portable system, inspired in the concept of Non-Public Network, which is able to provide 5G NR coverage to sites outside 5TONIC premises but using the 5G Core and the experimental infrastructure of 5TONIC. The design takes the advantage of the new capabilities provided by 5G, such as CUPS and User Plane flexibility, to deploy at vertical premises only the equipment required for providing the access network (gNB) and the local break-out of the user plane (UPF, transmission routers).

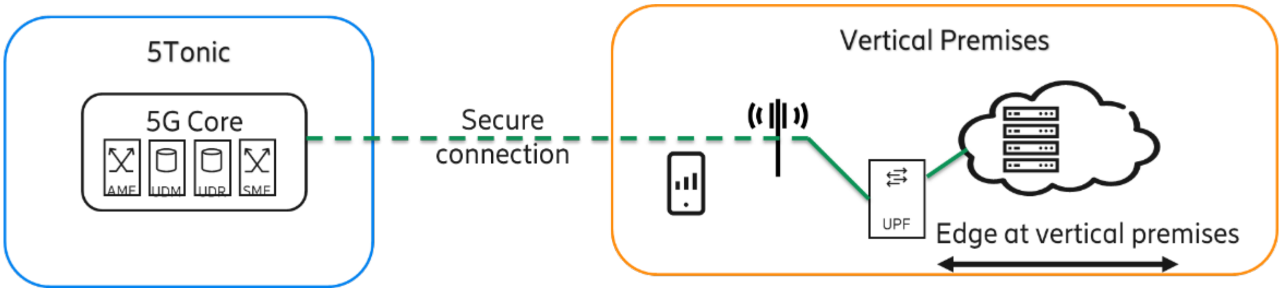


Figure 2-15. Diagram of NPN deployment

The on-premises equipment uses a secure connection towards 5TONIC facilities in order to get access to the control plane part of the 5G Core. With this, it is possible to extend the 5TONIC 5G coverage to external premises and to provide the same capabilities that 5TONIC has, including all the experimentation infrastructure.

- **5GS capabilities:**

eMBB default service. Provides higher data rates, better bandwidth, higher throughput, increased reliability and lower latency as well as improved multimedia functionality for the end user.

Exposure. Provides access to the exposed network services and capabilities of the 5G network for external consumption by enterprises, third party applications or developers.

Ericsson has developed a network exposure API to be consumed by any AF within 5TONIC domain, that currently offers a set of NPN related functionalities categorized as follows:

  - Analytics
  - Experimentation and validation support
  - Experimentation and optimization support
  - Network slicing

Network Slicing. Enables the multiplexing of virtualized and independent logical networks on the same physical network infrastructure.

Analytics and Intelligence. Provides a recommendation of network resources configuration based on the KPIs collected from the network.

Experimentation

Ericsson provides tools that are useful for the realization of the experimentation at 5TONIC laboratory.

- **KPI framework components**

Ericsson has developed a Key Performance Indicator framework that allows to collect and visualize KPIs from the 5G System. The key components of the framework are the following:

Software probe. The software probe is a component that extracts KPIs from the end-user traffic with the granularity of flow for IP traffic. A flow is identified by a tuple of origin IP address, destination IP address, origin Port, destination Port, type of protocol. This approach allows to obtain KPIs related to the application flow, which provides insights about the performance of a specific application in the 5G System.

The software probe can be deployed in any Linux-based system (tested in Ubuntu servers and Raspberry PI), Mac, and Windows, native or virtual, and it is configured to capture the application traffic from a system network interface. There are two deployment options: (i) when it is possible, the probe is deployed in the System Under Test (e.g.,, in the Application host or in the client host) (ii) the probe can be deployed in an independent system that receives a copy of the application traffic using port mirroring.

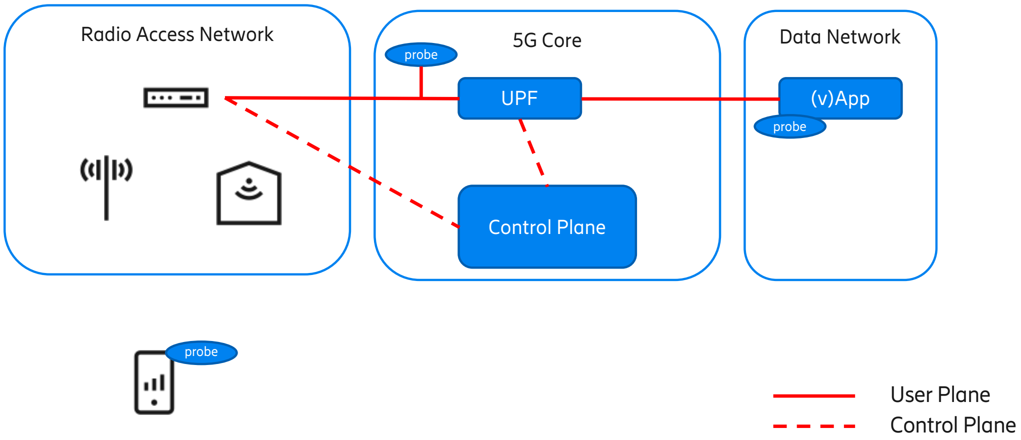


Figure 2-16. Software probe deployment options

In the figure above (Figure 2-16) we can see two probes running on the SUT host in the (v)APP and UE sides, and another probe that receives a port mirroring of N3 interface (user plane). In addition to the deployment options, the software probe supports the following capabilities: L2 interfaces, GTP tunnelling, L3 interfaces (VPN, e.g.,, tun/tap).

The software probe exposes a REST API that allows to control the probe, including the definition of the BFP filter, which defines the application traffic.

The software probe can generate the KPIs Round-Trip Time, One-Way Delay and Throughput, for both uplink and downlink. Other KPIs like Jitter and Reliability can be generated from the data exposed by the software probe in the real-time database.

Baseband and Routers collectors. Besides the software probe, statistics coming from the basebands (gNB) nodes and the routers are used as metric sources. As part of the KPI framework, we deployed several collectors that retrieve information from these systems: Throughput information per router interface (including VLAN interfaces) and cell status.

Real-time database. All the information generated by the metric sources is stored in an influxDB database, with the timestamp of the metric. This information is used for the derivation of KPIs and for the visualization.

Visualization. The visualization system is based on Grafana, which allows the creation of dashboards tailored for the use cases under test.

Active measurement. In addition to the passive measurement of the KPIs, we provide at 5TONIC a set of tools for performing active measurements of the KPIs, including net-tools (ping and iperfs) and Speed Test alternatives.

2.2.2 Description of the PoCs

These are the PoCs planned to demonstrate in PREDICT-6G in 5TONIC lab:

I. **Multi-Technology Multi-Domain DetNet-Based Integration of MDP and AICP:**

- **Summary:** deployment of a MDP based on the interaction of different technologies such as WLAN, TSN, and 3GPP, using DetNet as entry point, to test a set of complete end-to-end use cases.
- **Goal:** to provide deterministic networking solutions, in different fields such as Industry 4.0, medical applications and autonomous transport systems, across different technological domains including WLAN, 3GPP and TSN.
- **Tests:** Wireless control of mobile robot over 3 TSN and non-TSN access technologies : WiFi, Ethernet and 3GPP.
- **KPIs:** jitter, latency, packet ordering, reliability (packet loss).
- **Components:** IEEE 802.1 TSN switch, 802.11 AP, 3GPP 5G gNB, 3GPP 5G Core (including NW-TT/DS-TT TSN innovations), TSN App, 5G Hat.
- **Technology domains:** IEEE WLAN 802.11, IEEE 802.1 TSN (IEEE 802.1Qbv and FRER) and 3GPP 5G with deterministic capabilities (DS-TT/NW-TT).

Funded by
the European Union

II. Smart Factory:

- **Summary**: execution of the Smart Manufacturing use case using a 3GPP 5G mobile network enhanced with user plane improvements.
- **Goal:** to provide flexible deployments of manufacturing premises using a wireless network with deterministic capabilities to ensure the communication between factory elements with a high reliability and bounded latency.
- **Tests:** Smart Factory integration with commercial 3GPP 5G, Smart Factory integration with 3GPP TSN network, Smart Factory with commercial 3GPP 5G validation, Smart Factory with 3GPP TSN network validation
- **KPIs:** jitter, latency, packet ordering, reliability (packet loss)
- **Components**: PLC Siemens, Virtualized Industrial PC, 3GPP 5G gNB, 3GPP 5G Core (including NW-TT/DS-TT TSN innovations)
- **Technology domains**: 3GPP 5G with deterministic capabilities (DS-TT/NW-TT)

III. Localisation and Sensing:

- **Summary:** generation of meaningful sensing results using a range of sensor data to be processed in the network using AI/ML. The use case also includes the remote control of a vehicular device using haptic gloves.
- **Goal**: establish, monitor, adjust and release end to end flows on the control plane by means of AICP.
- **Tests:** TSN software switch integration, TSN-based packet switching using Data Unit Group IP header extension, TSN AF with AICP, CNC with AICP
- **KPIs**: jitter, latency, packet ordering, reliability (packet loss)
- **Components:** IEEE 802.1 TSN switch, 3GPP 5G gNB, 3GPP 5G Core (including NW-TT/DS-TT TSN innovations), TSN AF (including AICP support), CNC (including AICP support).
- **Technology domains:** IEEE 802.1 TSN and 3GPP 5G with deterministic capabilities (DS-TT/NW-TT).

2.2.3 System level integration

It is followed an integration methodology based on the definition of specific tests to be executed by a subset of partners at some point within the lifetime of the project.

2.2.3.1 Smart Factory integration

Smart Factory demonstration will focus on the evolution from a current on premises wired deployment towards a cloudified solution, where the controllers of the equipment are offloaded to a cloud environment and the connectivity between the factory elements is provided through a mobile 3GPP 5G network enhanced with user plane improvements.

The Smart Factory setup proposes a 5G deployment integrated as communication network in the factory. A Non-Public Network, that provides the wireless connectivity (5G NR) and the user plane connectivity (5G Core User Plane), is deployed on-premises. The NPN is connected to a Public Network (5G Core), provided by 5TONIC lab, through a dedicated transmission line. This approach allows to have a dedicated user plane with a Radio Access Network that can be shared for other purposes.

The elements involved in the integration are the following:

- **PLC**: SIEMENS equipment provided by GESTAMP
- **Virtualized Industrial PC**: located at the edge and provided by GESTAMP
- **3GPP 5G commercial network**: 5TONIC's 5G network provided by Ericsson.
- **TSN-Translators** (DS-TT and NW-TT): 3GPP 5G TSN switch, developed by Ericsson, to provide the deterministic enhancements for the 3GPP network at Layer 2.

The components integrated include:

- **General TSN technologies**: virtualized Industrial PC with real time capabilities.
    - **Functionality:** near real time capabilities over a hypervisor
    - **Leftovers**: improve real time capabilities over containers
- **3GPP TSN**: TSN-Translators
    - **Functionality:** Asynchronous Traffic Shaper (IEEE 802.1Qcr), frame replication (FRER), synchronous traffic scheduler (IEEE 802.Qbv) and de-jittering
    - **Leftovers**: end to end testing integration and verification.

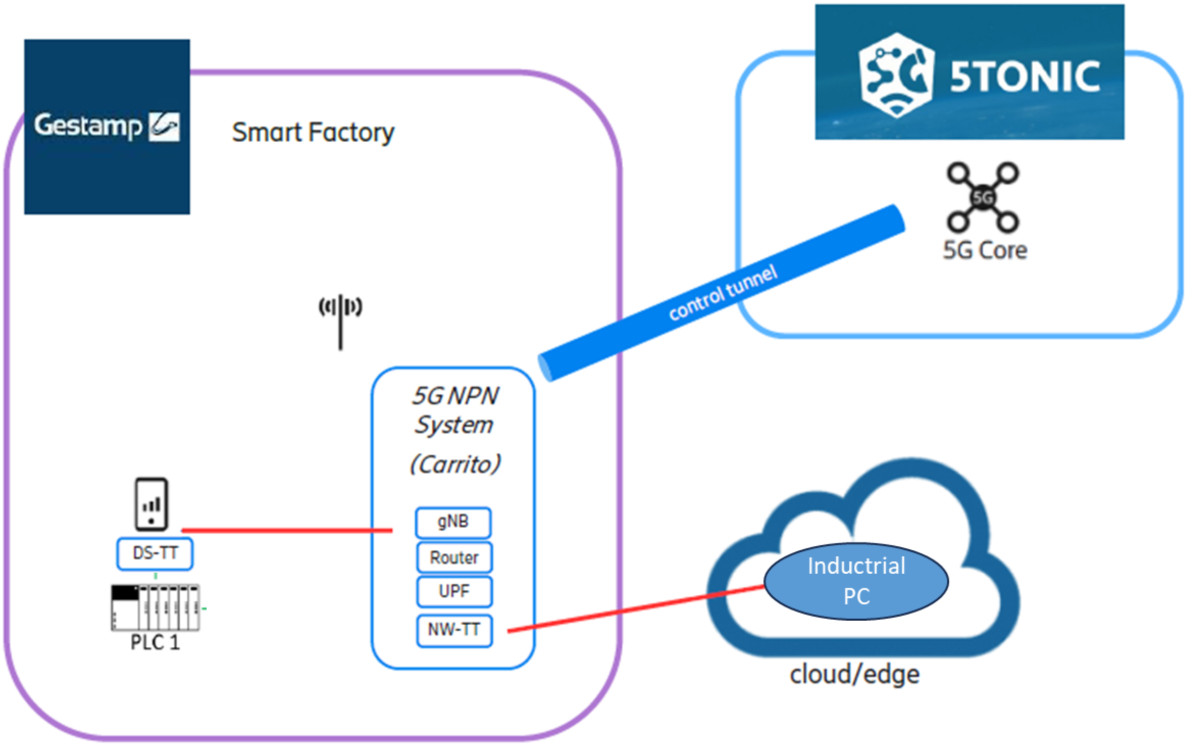


Figure 2-17. 5TONIC experimentation facility

The integration testing follows a stepwise approach where initially, Smart Factory integration will use a 3GPP 5G commercial network and finally a 3GPP 5G commercial network enhanced with TSN capabilities (i.e.,: NW-TT/DS-TT).

Parameter	Description
ID	SMART_FACTORY_INTEGRATION_5G_COMMERCIAL
Description	Enable smart factory communication through a 3GPP 5G commercial mobile network
Management domain services	Topology Exposure, Capability Exposure, Resource Exposure
Technology domain (TD)	3GPP

Specific TD Interfaces	topology_exposure, capability_exposure, resource_exposure
Targeted period	2024-H2

Table 2-3. Integration test Smart Factory with 5G commercial

Parameter	**Description**
Test ID	SMART_FACTORY_INTEGRATION_5G_COMMERCIAL_RESULTS
Result	(SUCCESSFUL/PARTIALLY SUCCESSFUL/NOT SUCESSFUL)
Date	2024-H2
Observations and comments	Details to aid the resolution of partially or not successful tests

Table 2-4. Integration test Smart Factory with 5G commercial results

Parameter	**Description**
ID	SMART_FACTORY_INTEGRATION_5G_TSN_ENABLED
Description	Enable smart factory communication through a 3GPP 5G TSN enabled mobile network
Management domain services	Topology Exposure, Capability Exposure, Resource Exposure
Technology domain (TD)	3GPP
Specific TD Interfaces	topology_exposure, capability_exposure, resource_exposure
Targeted period	2024-H2

Table 2-5. Integration test Smart Factory with 5G TSN enabled network.

Parameter	**Description**
Test ID	SMART_FACTORY_INTEGRATION_5G_TSN_ENABLED_RESULTS
Result	(SUCCESSFUL/PARTIALLY SUCCESSFUL/NOT SUCESSFUL)

Funded by the European Union

Date	2024-H2
Observations and comments	Details to aid the resolution of partially or not successful tests

Table 2-6. Integration test Smart Factory with 5G TSN enabled network results.

2.2.3.2 MDP integration

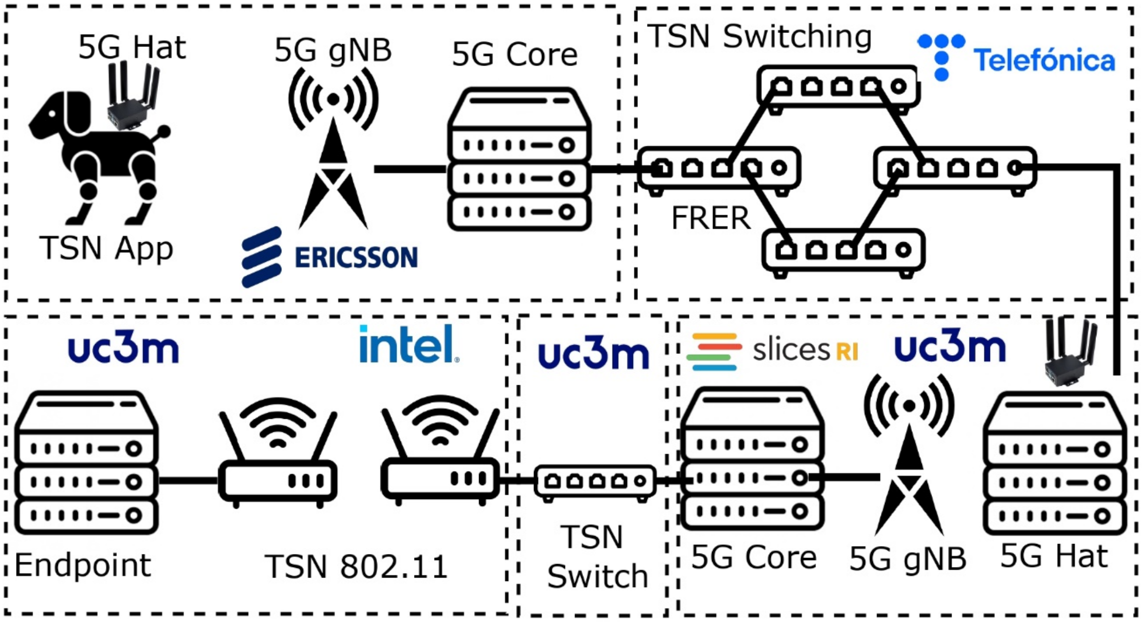


Figure 2-18. Setup of the Multi-Domain, multi-technology dataplane testbed

The Multi-Domain Data Plane (MDP) is designed to provide deterministic networking solutions across different technological domains. This includes integrating technologies such as IEEE 802.11, IEEE 802.1 TSN, and 3GPP 5G, using the IETF DetNet overlay data plane. The goal of MDP is to ensure coherence and precision in dynamic and critical environments like Industry 4.0, medical applications, and autonomous transport systems.

The implementation involves various components and configurations to allow for remote control of a mobile robot dog across heterogenous TDs :

- **TSN Ethernet Domain (UC3M):** Utilizes RELYUM-TSN-BRIDGE4 switches in a diamond topology, supporting IEEE 802.1Qbv (Time-Aware Shaper) and FRER (Frame Replication and Elimination for Reliability). The configuration ensures high reliability and low latency.
- **3GPP 5G Domain (UC3M)**: Deployed using OpenAirInterface (OAI) to provide a robust and flexible 5G SA network. Enhancements has been made over it in order to enable TSN ensuring deterministic communication over 5G.
- **3GPP 5G Domain (ERICSSON):** TSN 3GPP 5G capable connectivity, with TELF TSN Switches on core, Qbv capable.
- **DetNet Integration (UC3M):** Employing IETF DetNet standards to ensure deterministic networking over IP-based networks. DetNet is used as the endpoints of the E2E as well as frontier nodes between domains, in charge of not only performing switching operations, but acting as translators for enabling TSN in the domain (NW-TT/DS-TT like). This includes using techniques like Packet Replication, Elimination, and Ordering Functions (PREOF).
- **802.11 WiFi (INTEL):** WiFi system capable of doing Qbv for scheduling as well as AS for time sync over the air.

Parameter	Description
ID	REMOTE_CONTROL_MOBILE_ROBOT_ACROSS_DOMAINS
Description	Enable a remote operator to control a mobile robot across WiFi, cellular wireless and ethernet networks.
Technology domain (TD)	IEEE 802.11 (WiFi), 3GPP, Ethernet TSN
Specific TD Interfaces	IEEE DetNet
Targeted period	2024-H2

Table 2-7. Integration test robot remote control across technology domains

Parameter	Description
Test ID	REMOTE_CONTROL_MOBILE_ROBOT_ACROSS_DOMAINS_RESULTS
Result	SUCCESSFUL
Date	2024-H2

Observations and comments	Details to aid the resolution of partially or not successful tests

Table 2-8. Integration results robot remote control across technology domains

- MDP innovations for non-deterministic networks

The integration of Time-Sensitive Networking (TSN) traffic into non-deterministic networks like OpenAirInterface (OAI) 3GPP 5G employs key techniques to ensure bounded latency and minimal packet loss. One primary method involves dynamic adaptation of the Modulation and Coding Scheme (MCS). By adjusting the MCS to target a lower Block Error Rate (BLER) without relying on retransmissions, this approach ensures reliable TSN traffic transmission with minimal jitter, even under challenging wireless conditions.

Additionally, adaptive traffic throttling prioritizes TSN traffic by managing the overall traffic entering the 5G network. During network congestion, this mechanism buffers non-critical traffic to preserve the required bandwidth for TSN flows. This prioritization helps maintain consistent latency and reliability for TSN packets.

Lastly, frame resource allocation and de-jittering techniques synchronize TSN packet transmissions with the radio frame. This ensures that packets are consistently scheduled within the same subframe, reducing jitter and aligning with the stringent requirements of TSN. Together, these methods allow non-deterministic networks to handle TSN traffic with the necessary low latency and high reliability.

- MDP functional architecture

The proposed architecture extends IETF DetNet to enable end-to-end deterministic services across multi-domain environments. It integrates IEEE 802.1 TSN, 5G TSN, and IP networks into a unified control and data plane framework to ensure low latency, minimal jitter, and high reliability. The AI-driven Multi-Stakeholder Inter-domain Control-Plane (AICP) provides the necessary intelligence to manage deterministic paths dynamically, using AI/ML algorithms to allocate resources, predict network load, and manage SLAs. By abstracting technology-specific complexities, the AICP offers a unified control interface that orchestrates deterministic services across technologies like TSN and 5G seamlessly.

The Multi-Domain Data Plane (MDP) ensures that deterministic traffic can traverse heterogeneous networks while maintaining strict quality-of-service (QoS) guarantees. It employs DetNet Layer-3 capabilities to create deterministic paths using Packet Replication, Elimination, and Ordering Functions (PREOF), preventing packet loss and ensuring consistent data delivery across domains. The MDP coordinates closely with the AICP to maintain minimal latency and high reliability for deterministic paths. The architecture also includes Management Services (MS) responsible for tasks like Time Synchronization, Path Computation, and Service Automation. These services ensure precise timing for time-sensitive applications and optimize deterministic paths across domains.

To maintain deterministic services across different network domains, the architecture incorporates gateways (Predict-6G DetNet Extended routers) at domain borders. These gateways translate QoS requirements and service parameters, ensuring traffic retains its deterministic properties when transitioning between technologies like TSN and 5G. PREOF mechanisms ensure high reliability by replicating packets across multiple paths and eliminating duplicates, thus avoiding packet loss and out-of-order delivery. Designed to be modular and extensible, the architecture includes model-driven APIs that facilitate the integration of future technologies like 6G, ensuring deterministic services can scale to meet the demands of complex multi-domain environments without sacrificing performance.

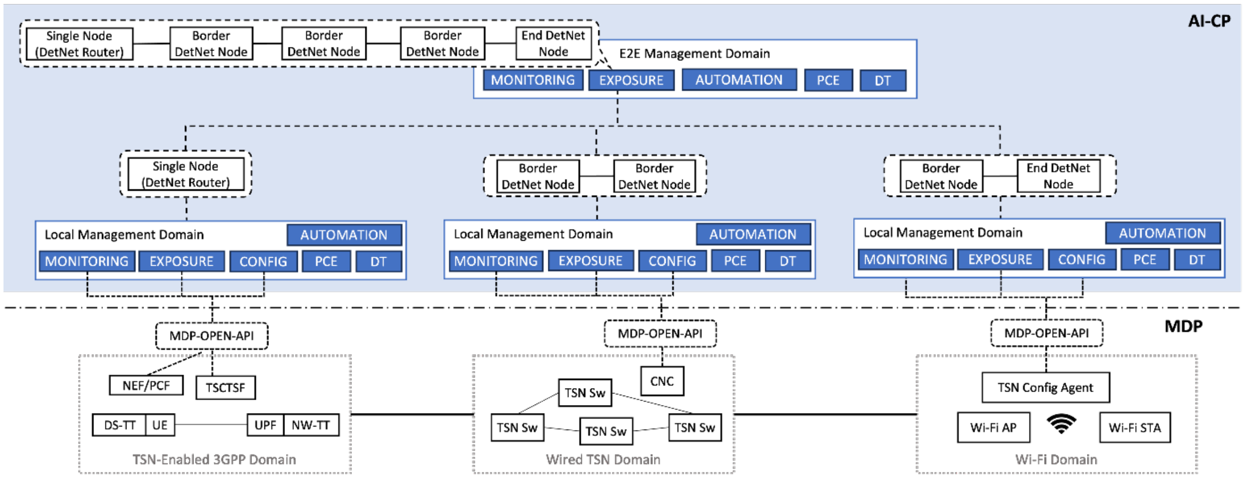


Figure 2-19. Functional architecture of the multi technology data plane

2.2.4 Localisation and Sensing integration

The Localisation and Sensing use case described in D1.2 [2] and D4.1 [1] will be integrated and validated at 5TONIC. Figure 2-20 depicts the testbed topology to integrate MDP and AICP components and to validate the technological advances in an end-to-end fashion. The testbed is composed of two technology domains, i.e., a DetNet-enabled 3GPP domain and a DetNet domain which 3GPP refers to as the Data Network (DN). The 3GPP domain is composed of two

UEs and one UPF while the DetNet domain is composed of a single DetNet router and the application deployed on a dedicate compute host. On the top, the AICP is illustrated which implements a Data Repository (Monitoring Collection MS), Model Management (MLOps orchestrator MS) and Network Management (Service Provisioning MS). Note, Figure 2-19, purposely illustrates terms from SDOs such as 3GPP (SA), IETF (DetNet)/IRTF (NMRG) for better alignment with skilled in the art readers. The MDP in the Localisation and Sensing testbed implements the Data Unit Group innovation, described in D2.2 [4], as part of a DetNet router. The 3GPP domain exposes itself via the NEF as a single DetNet router with three ports, i.e., UE1, UE2 and UPF, the information exposed by the NEF includes capabilities as well as the topology. Furthermore, the NEF exposes run-time monitoring data about link latencies and DUG-specific data points to the AICP. The same exposure capabilities of information apply to the DetNet domain, based on the collected monitoring information, the main objective is to configure the DUG-enabled DetNet behaviour based on end-to-end Service Flow characteristics for traffic from UEs to the application server and back. The actual traffic pattern put upon the testbed will be decided at a later stage. The DUG-enabled DetNet routers implement basic DUG drop behaviour to ease any deterministic packet delivery bottlenecks. The AICP is controlling this DUG drop rate using the collected monitoring data and applies ML techniques to predict packet delivery behaviour on the Data Plane.

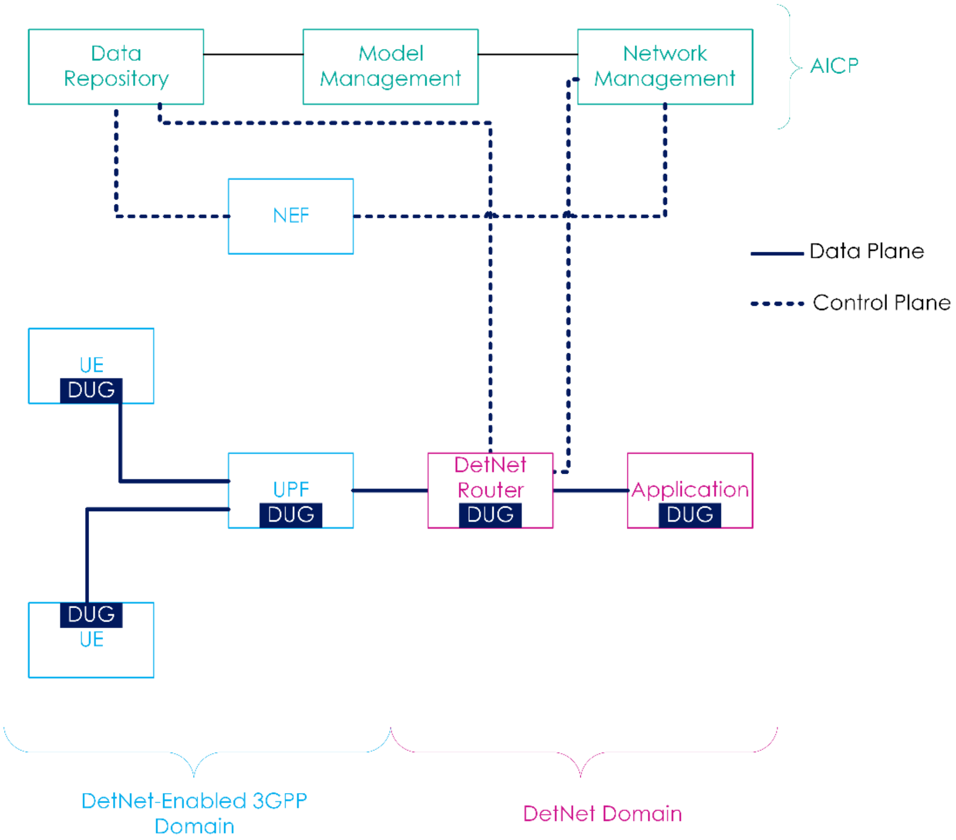


Figure 2-20: Localisation and Sensing testbed topology

The table below summarised the targeted validation experiments.

Table 2-9: Validation test for Localisation and Sensing Use Case

Parameter	Description
ID	LOCALISATION_SENSING
Title	Localisation and Sensing
Use Case	Localisation and Sensing
Description	This PoC uses the DUG concept in a Localisation and Sensing use case where sensing data from UEs will traverse a 3GPP and DetNet domain before reaching the application server where they can be processed in a timely fashion. The sensory data is then being used in a collision avoidance application to stop a

	robot on the factory floor if unexpected objects are in the direction of travel.
MD & MS Involved	◆ E2E Monitoring, E2E Path computation, E2E Resource Manager, E2E Topology Exposure, E2E Service Exposure ◆ Measurement collection, Path computation, Resource Configuration, Resource Exposure, Topology exposure, Capability exposure
TD Involved	DetNet (3GPP, TSN)
Integration Tests	◆ TSN software switch integration ◆ TSN-based packet switching using Data Unit Group IP header extension. ◆ DetNet-enabled 3GPP technology domain integration with AICP ◆ DetNet technology domain integration with AICP
Targeted period	2024-H2
Laboratory and setup	See the content above
KPIs and validation	Jitter, latency, packet ordering, reliability (packet loss) are targeted
PoC Leader (INTERNAL)	Sebastian Robitzsch (IDE)

2.2.5 System level verification

It is followed the definition of specific validation tests, to be executed by a subset of partners at some point within the lifetime of the project, to demonstrate how the PREDICT-6G technologies enable a reliable, time engineered and predictable 6G network. It is also included results from the published paper.

Funded by the European Union

2.2.5.1 Smart Factory verification

Parameter	Description
ID	SMART_FACTORY_VERIFICATION_5G_COMMERCIAL
Description	The objective is to measure all the KPIs defined by Smart Factory use case with a 3GPP 5G commercial mobile network
E2E and MD specific MS involved	Measurement Collection, Service Automation
Technology domains involved	3GPP
Integration	SMART_FACTORY_INTEGRATION_5G_COMMERCIAL
Targeted period	2024-H2
Laboratory setup	The laboratory must be configured as described for Smart Factory demonstration with 3GPP 5G commercial mobile network
KPIs and validation	Smart Factory: Latency: <= N ms Jitter: <= N ms Packet order == N% Packet lost == N% Resilience >= N%

Table 2-10. Validation test Smart Factory with 5G commercial

Parameter	Description
ID	SMART_FACTORY_VERIFICATION_5G_TSN_ENABLED
Description	The objective is to verify that all KPIs defined by Smart Manufacturing use case are met
E2E and MD specific MS involved	Measurement Collection, Service Automation

Technology domains involved	3GPP
Integration	SMART_FACTORY_INTEGRATION_5G_TSN_ENABLED
Targeted period	2024-H2
Laboratory setup	The laboratory must be configured as described for Smart Factory demonstration with 3GPP 5G TSN enhancements (DS-TT/NW-TT)
KPIs and validation	Smart Factory: Latency <= 5 ms Jitter <= 2 ms Packet order == 100% Packet lost == 0% Resilience >= 99,9999%

Table 2-11. Validation test Smart Factory with 5G TSN enabled network

2.2.5.2 MDP verification

Parameter	**Description**
ID	REMOTE_CONTROL_MOBILE_ROBOT_ACROSS_DOMAINS
Description	The objective is to test the effectiveness of IEEE DetNet integration
Technology domains involved	IEEE 802.11 (WiFi), 3GPP, Ethernet TSN
Integration	REMOTE_CONTROL_MOBILE_ROBOT_ACROSS_DOMAINS
Targeted period	2024-H2
Laboratory setup	The 3 access technology domains must be configured to allow MPLS and ensure the correct working of IEEE DetNet

Funded by the European Union

KPIs and validation	Remote robot control (TSN Traffic): Latency: <= 20 ms Jitter: <= 13 ms Packet order == 85% Packet lost == 0% Resilience >= 99.9%

Table 2-12. Validation test for Remote Control of mobile robot across domains

2.2.6 Ethernet TSN Service Provisioning Integration

An initial integration targeting the support for automated provisioning of deterministic services in an Ethernet TSN domain has also been performed. This integration involved five different components of the specific AICP MDP domain, namely the Service Automation, Path Computation, Resource Configuration and Measurement collection. In Figure 2-21, we depict the MDP components involved in this integration. As detailed in D1.2 [2] the scope of each of these modules is as follows:

- **Service Automation**, responsible for receiving the lifecycle management requests from the E2E layer of the AICP and for triggering and coordinating the operations of with the rest of the components of the MDP.
- **Path Computation**, containing the logic to determine the route to be traversed by the service flows considering the deterministic constraints requested from the E2E layer.
- **Digital Twin,** responsible for estimating the KPIs
- **Resource Configurator**, supporting resource retrieval, management and configuration APIs for the configuration of the Ethernet TSN domain.
- **Data Collection and Management of the Measurement collection MS**, implementing the logic for the dynamic creation of the Ethernet TSN domain monitoring, and on-demand configuration of the service metrics with the required access control policies.

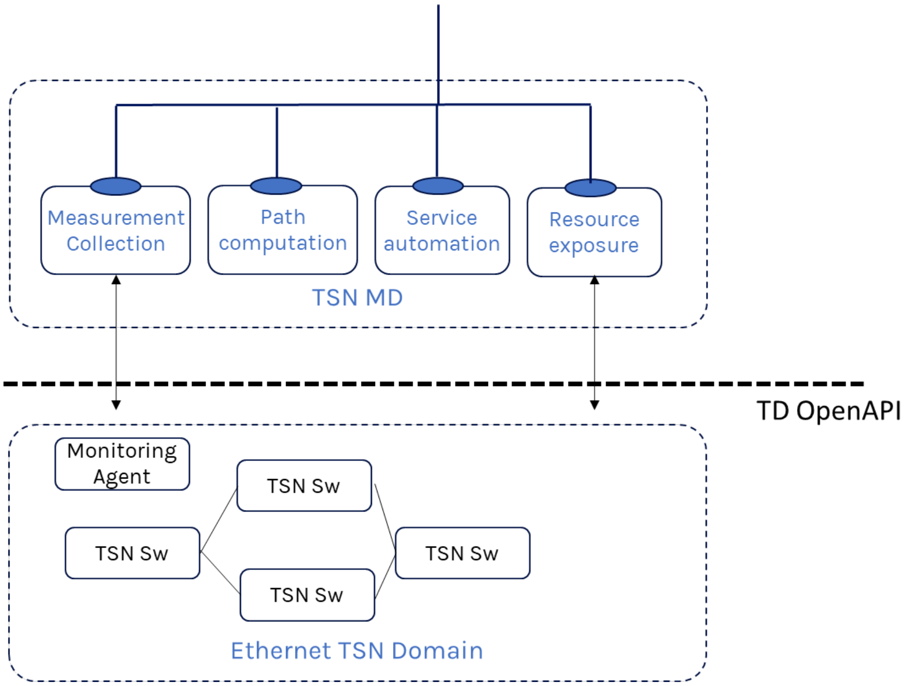


Figure 2-21. Ethernet TSN Service Provisioning Integration MD components

The overall workflow of this integration test, depicted in Figure 2-22, is mainly divided in three stages: (i) Determining the specific path to be allocated; (ii) Allocate the network resources to provision the service; (iii) configure the required data sources and monitoring metrics for the service. During the first phase, the Service Automation processes the request and triggers the Path Computation, which returns the selected path from the analysis of the service QoS characteristics and of the topology information of the TSN domain. In the second phase the Resource Configuration, upon receiving the request from the Service Automation module, performs the Ethernet TSN specific operations for provisioning of selected path and configuring the TSN scheduler. Finally, in the third phase, the Service Automation sends a request to the measurement collection to create a TSN data source with the targeted metrics.

Funded by
the European Union

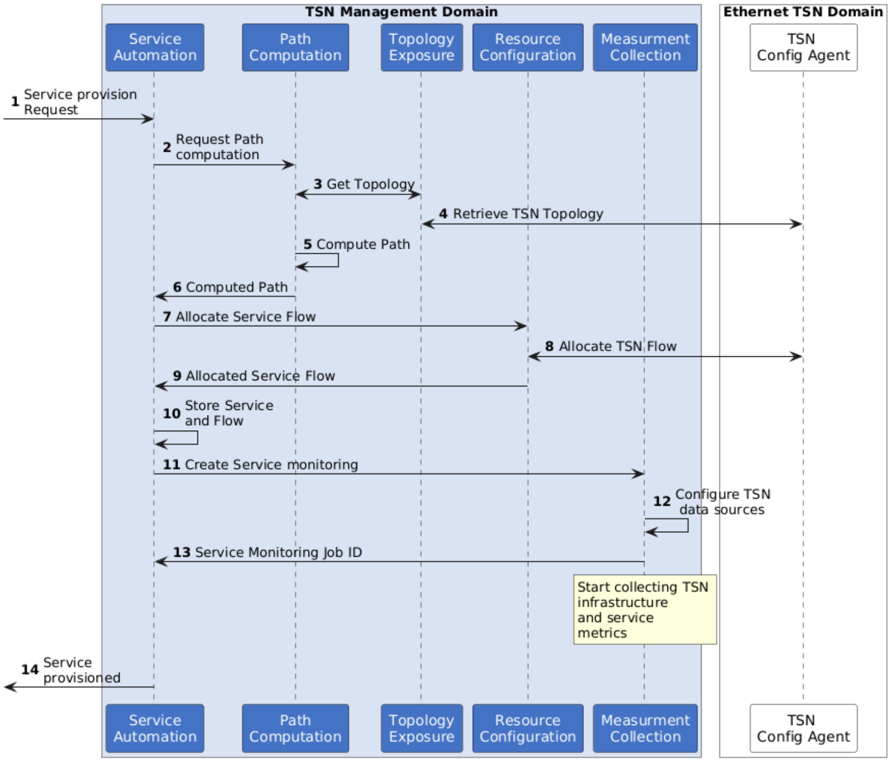


Figure 2-22. Ethernet TSN Service provisioning integration workflow

In Table 2-13 we provide the report for this integration test. More details regarding the specific integration activities done on each module are provided in the following subsections.

Parameter	Description
ID	SERVICE_PROVISIONING_ETHERNET_TSN
Description	This test confirms the proper creation of the collector plugins to retrieve the monitoring data associated to a specific service from a e Ethernet TSN TD.
Management domain services	Service Automation Resource Configuration Path Computation Measurement collection

Technology domain (TD)	Ethernet TSN domain
Specific MD Interface & methods	data_collection.configuration
Reference to TD interfaces	• This test used a TD monitoring agent implementing the API defined in [5] (see section 2.2.4) • This test used the TD Resource Configuration API.
Result	SUCCESSFUL The Data Collection and Management of the measurement collection module was able to retrieve data from an emulated agent stub.
Date	07/2024
Observations and comments	No observations

Table 2-13. SERVICE_PROVISIONING_ETHERNET_TSN Test report

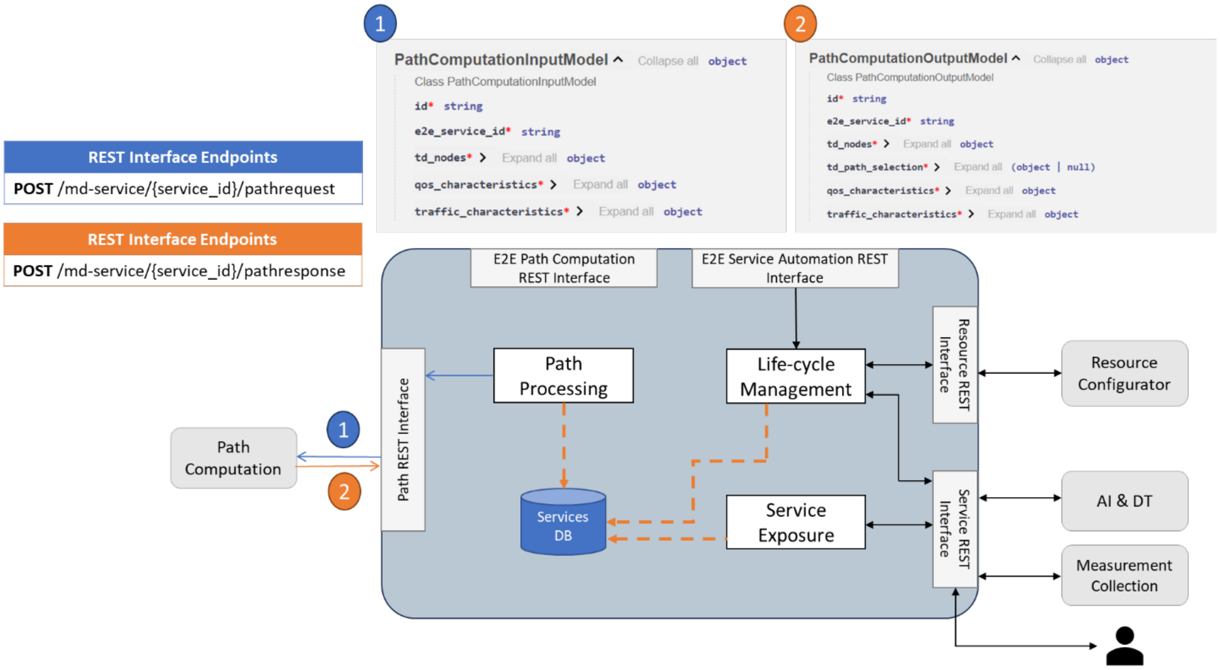
Service Automation

Service Automation plays a crucial role in the service provisioning and decommissioning phases, since it is in charge of processing lifecycle management requests and triggering and coordinating the operation of the rest of AICP MS. Consequently, the onboarding of the Service Automation requires of specific integrations with several MS such as Path Computation, Resource Configuration and Data Collection and Management.

Service Automation – Path Computation.

Service Automation needs to be integrated with the Path Computation MS to request the computation of the path for a specific service, as well as for receiving the result of path computation process. The integration between both modules is performed via a specific HTTP REST interface, named Path REST interface, which is allocated within the Service Automation design. This interface will connect the Path Processing submodule, which is part of the Service Automation, and the Path Computation MS, which has been emulated in these initial tests.

The Path REST interface has been implemented in the shape of REST API, where two specific endpoints have been made available for requesting and receiving the result of the path computation process:



REST Interface Endpoints
POST /md-service/{service_id}/pathrequest

REST Interface Endpoints
POST /md-service/{service_id}/pathresponse

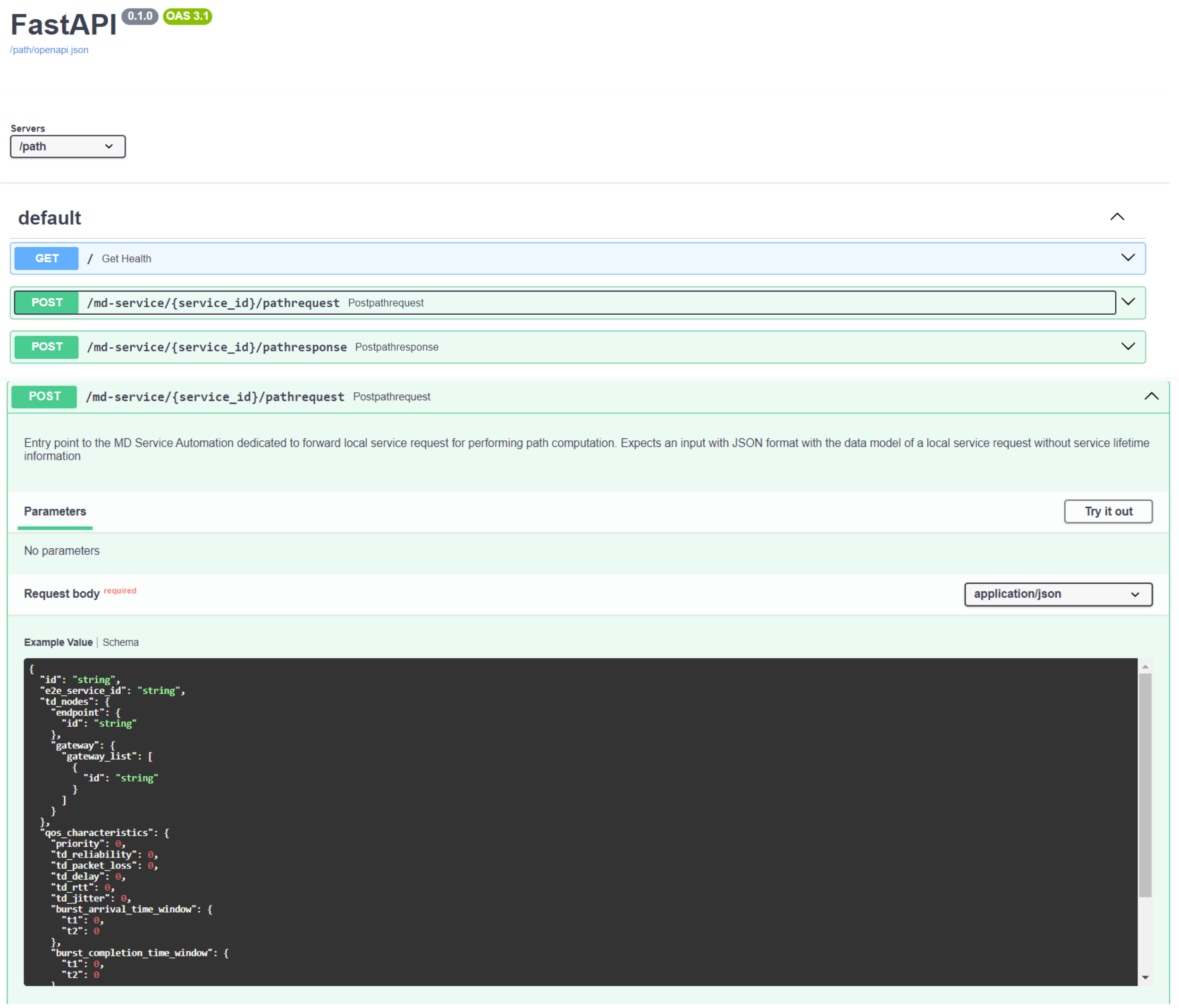
① PathComputationInputModel ^ Collapse all object
Class PathComputationInputModel
id* string
e2e_service_id* string
td_nodes* > Expand all object
qos_characteristics* > Expand all object
traffic_characteristics* > Expand all object

② PathComputationOutputModel ^ Collapse all object
Class PathComputationOutputModel
id* string
e2e_service_id* string
td_nodes* > Expand all object
td_path_selection* > Expand all (object | null)
qos_characteristics* > Expand all object
traffic_characteristics* > Expand all object

Figure 2-23. Service Automation – Path Computation Integration Workflow

1. **/md-service/{service_id}/pathrequest.** This endpoint is called via a POST method by sending a JSON object in the body, following a Pydantic PathComputationInputModel data model, from the Service Automation towards the Path Computation MS. This data model integrates the id of the MD Service, the id of the E2E Service related to the MD service, the information of the nodes where the path must be computed, and the QoS and traffic characteristics to be considered for the path computation. Once received this information, the Path Computation MS performs the computation of the path, considering the nodes and the QoS and traffic characteristics specified.
2. **/md-service/{service_id}/pathresponse.** This endpoint is also called via a POST method by sending back from the Path Computation to the Service Automation MS a JSON object in the body, following a Pydantic PathComputationOutputModel data model. This data model is composed of the same fields as PathComputationInputModel but adding td_path_selection data, which specifies the path computed by the Path Computation MS.

The integration of both MS has been tested in an emulated FastAPI + Swagger environment where both endpoints have been called to test the performance of the API REST Interfaces:



Entry point to the MD Service Automation dedicated to forward local service request for performing path computation. Expects an input with JSON format with the data model of a local service request without service lifetime information

No parameters

```
{
  "id": "string",
  "e2e_service_id": "string",
  "td_nodes": {
    "endpoint": {
      "id": "string"
    },
    "gateway": {
      "gateway_list": [
        {
          "id": "string"
        }
      ]
    }
  },
  "qos_characteristics": {
    "priority": 0,
    "td_reliability": 0,
    "td_packet_loss": 0,
    "td_delay": 0,
    "td_rtt": 0,
    "td_jitter": 0,
    "burst_arrival_time_window": {
      "t1": 0,
      "t2": 0
    },
    "burst_completion_time_window": {
      "t1": 0,
      "t2": 0
```

Figure 2-24. Service Automation – Path Computation FASTAPI + Swagger Interface

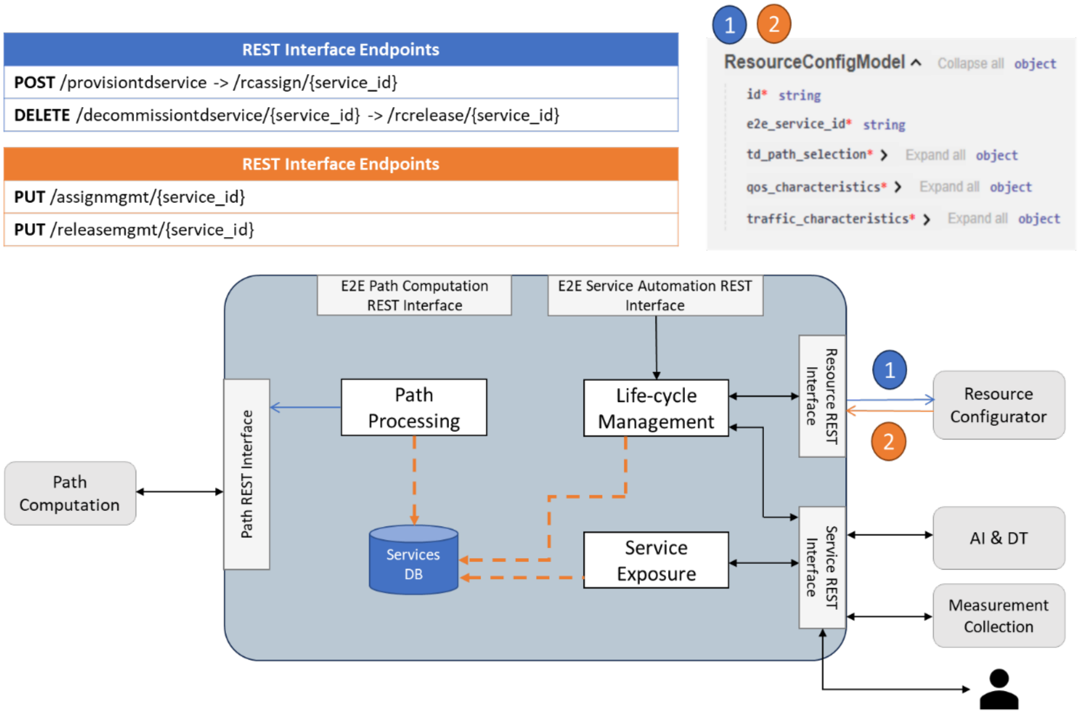
Parameter	Description
ID	INTEGRATION_TEST_SERVICE_AUTOMATION_PATH_COMPUTATION
Description	Internal test performed to validate the integration and communication between Service Automation and Path Computation. This integration is implemented to address two information flows: i) send the service request from the E2E Path Computation towards the Path Computation MS via the Service Automation for computing the path, and ii) forward the Path Computation result performed towards the E2E Path Computation MS. In both cases, the Path Computation REST Interface is used to ensure the communication between both MS. To perform this test, Eviden implemented the Service Automation and a mock-up version of the Path Computation since the latter will be really implemented by UPC.
Management domain services	Service Automation Path Computation
Technology domain (TD)	Ethernet TSN domain
Specific MD Interface & methods	Interface: Path Computation REST Interface defined as part of the MD Service Automation MS. Method: POST at /md-service/{service_id}/pathrequest and /md-service/{service_id}/pathresponse
Reference to TD interfaces	None
Result	SUCCESSFUL The Service Automation MS was able to retrieve data from the Path Computation mock-up.
Date	03/2024
Observations and comments	This test will be extended in the future to verify the integration with the real implementation of the MD Path Computation MS.

Table 2-14. Service Automation – Path Computation Integration Test

Service Automation – Resource Configuration

Once integrated with the Path Computation, Service Automation needs to communicate with the Resource Configuration MS to trigger or release the allocation of resources in the nodes specified for the selected path. The integration between Service Automation and Resource Configuration is performed via a specific HTTP REST interface, named Resource REST interface. This interface will connect the Lifecycle Management submodule, which is located inside the Service Automation, and the Resource Configuration MS, which for the purpose of these initial tests has also been emulated.

The Resource REST interface has been implemented in the shape of REST API, where four specific endpoints have been made available for provisioning and decommissioning services by assigning and releasing resources, as well as for updating the resource assignment and release after the configuration:



REST Interface Endpoints
POST /provisiontdservice -> /rcassign/{service_id}
DELETE /decommissiontdservice/{service_id} -> /rcrelease/{service_id}

REST Interface Endpoints
PUT /assignmgmt/{service_id}
PUT /releasemgmt/{service_id}

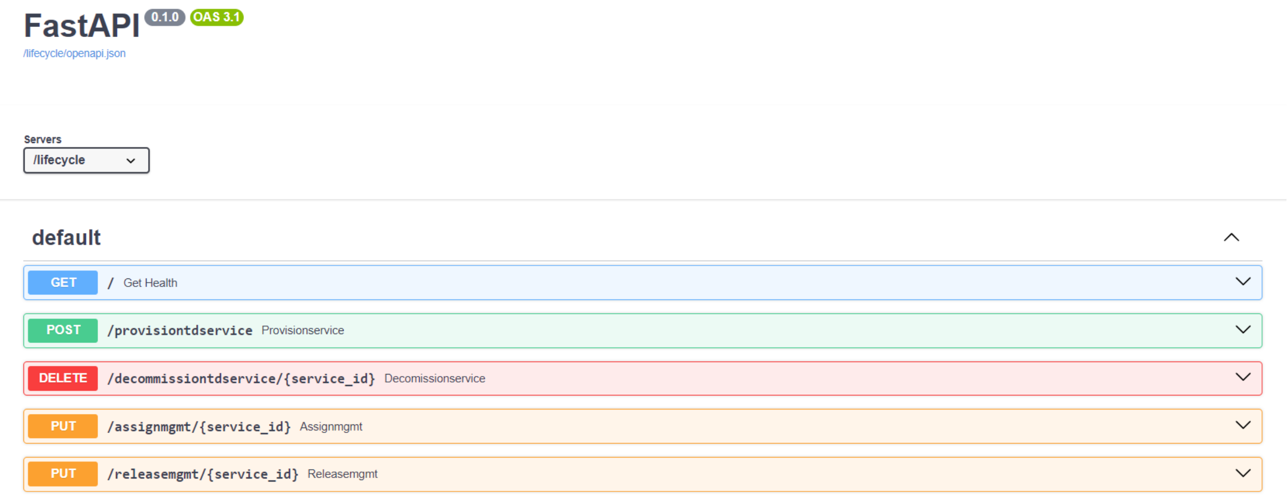
```
ResourceConfigModel ^   Collapse all  object
    id*  string
    e2e_service_id*  string
    td_path_selection* >   Expand all  object
    qos_characteristics* >   Expand all  object
    traffic_characteristics* >   Expand all  object
```

Figure 2-25. Service Automation – Resource Configuration Integration Workflow

1. **/provisiontdservice.** This endpoint is called by a POST method at the Service Automation for provisioning a deterministic MD service. Calling this endpoint leads to a second POST call to the **/rcassign/{service_id}** endpoint at the Resource Configuration MS, leading to start the configuration of the resources at the different nodes of the path previously selected by the Path Computation MS. The POST method is called by sending a JSON object following a ResourceConfigModel data model, which encompasses the id of the MD Service, the id of the E2E Service related to the MD service, the path selected including the information of all nodes involved, and the QoS and traffic characteristics to be considered for the resource configuration. Once received this body, the Resource Configuration MS configures the resources considering the QoS and traffic characteristics required.
2. **/assignmgmt/{service_id}.** This endpoint is called via a PUT method, which is triggered at the Resource Configuration module for updating the information associated to the MD Service at the Service Automation MS. The endpoint is usually called immediately after performing the resource configuration via the endpoint 1. The PUT call is done by sending a JSON object in the body following a ResourceConfigModel data model as for the previous endpoint. As a main difference, in this case, the *td_path_selection* field will include the information of the resources assigned to each node belonging to the path. Once received the body at the lifecycle management submodule of the Service Automation, the submodule updates the database to save the updated resource information and set the status of the service as “provisioned”.
3. **/decommissiontdservice/{service_id}.** This endpoint is called via a DELETE method, which is triggered at Service Automation for decommissioning an existing MD deterministic service. The call to this endpoint leads to a second POST call to the **rcrelease/{service_id}** endpoint at the Resource Configuration MS, which starts the release of the resources at the different nodes of the path selected specified. The call to the DELETE method is done by specifying the identifier of the MD service to be decommissioned. Based on this identifier, the lifecycle management submodule retrieves the data related to the MD Service and triggers the POST at the Resource Configuration MS to release the resources assigned to the specific service.
4. **/releasemgmt/{service_id}.** This endpoint is called via a PUT method, which is triggered at the Resource Configurator for updating the information associated to the MD Service after its decommissioning. This endpoint is usually called immediately after performing the resource configuration via the endpoint 3. The PUT call to this endpoint is done by sending a JSON object in the body following a ResourceConfigModel data model as for the previous endpoint. As a main difference, in this case, the td_path_selection field will include the information of the resources released in each node belonging to the path. Once

received this information at the lifecycle management submodule of the Service Automation, the submodule updates the database to save the released resource information and set the status of the service as “decommissioned”.

The integration of both MS has been tested in an emulated FastAPI + Swagger environment where all endpoints have been called to test the performance of the API REST Interfaces:



FastAPI 0.1.0 OAS 3.1

/lifecycle/openapi.json

Servers

/lifecycle

default

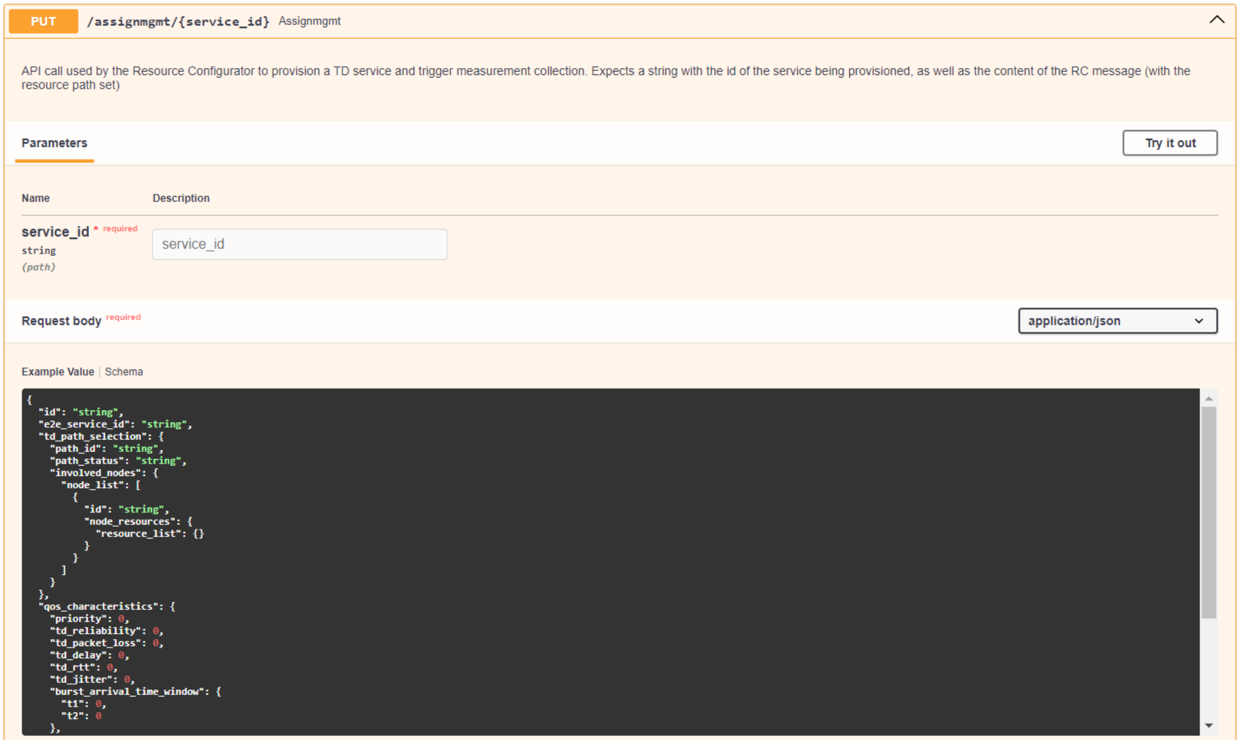
GET / Get Health

POST /provisiontdservice Provisionservice

DELETE /decommissiontdservice/{service_id} Decomissionservice

PUT /assignmgmt/{service_id} Assignmgmt

PUT /releasemgmt/{service_id} Releasemgmt



API call used by the Resource Configurator to provision a TD service and trigger measurement collection. Expects a string with the id of the service being provisioned, as well as the content of the RC message (with the resource path set)

service_id * required
string
(path)

service_id

Request body required

```
{
  "id": "string",
  "e2e_service_id": "string",
  "td_path_selection": {
    "path_id": "string",
    "path_status": "string",
    "involved_nodes": {
      "node_list": [
        {
          "id": "string",
          "node_resources": {
            "resource_list": {}
          }
        }
      ]
    }
  },
  "qos_characteristics": {
    "priority": 0,
    "td_reliability": 0,
    "td_packet_loss": 0,
    "td_delay": 0,
    "td_rtt": 0,
    "td_jitter": 0,
    "burst_arrival_time_window": {
      "t1": 0,
      "t2": 0
    },
```

Figure 2-26. Service Automation – Resource Configuration FASTAPI + Swagger Interface

Parameter	**Description**
ID	INTEGRATION_TEST_SERVICE_AUTOMATION_RESOURCE_CONFIGURATION
Description	Internal test performed to validate the integration and communication between Service Automation and Resource Configuration. This integration is implemented to address two information flows: i) perform the provisioning or decommissioning of the MD Service by requesting the assignment or release of resources to the Resource Configurator, and ii) return the MD Service updated with the resources assigned or released after the previous request. In both cases, the Resource REST Interface is used to ensure the communication between both MS.

	To perform this test, the Service Automation MS and a mock-up version of the Resource Configurator have been implemented.
Management domain services	Service Automation Resource Configuration
Technology domain (TD)	Ethernet TSN domain
Specific MD Interface & methods	Interface: Path Computation REST Interface defined as part of the MD Service Automation MS. Method: POST /provisiontdservice/, PUT /assignmgmt/{service_id}, DELETE /decommissiontdservice/{service_id}, PUT /releasemgmt/{service_id}
Reference to TD interfaces	None
Result	SUCCESSFUL The Service Automation MS was able to retrieve data from the Resource Configuration Computation mock-up.
Date	03/2024
Observations and comments	This test needs to be extended in the future to verify the integration with the real implementation of the Resource Configurator MS.

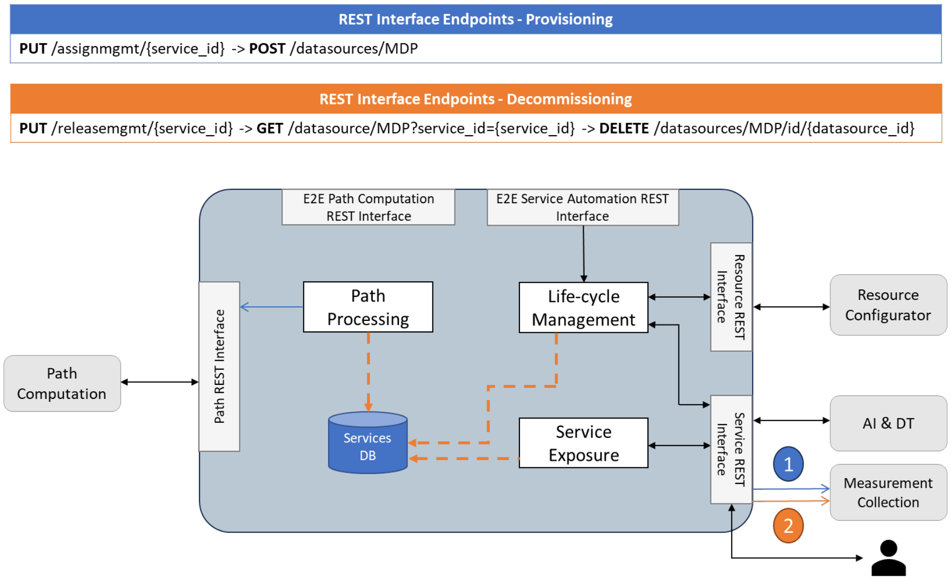
Table 2-15. Service Automation – Resource Configuration Integration Test

Service Automation – Data Collection and Management

After completing the integration with the Resource Configurator, Service Automation MS will interact with the Measurement collection MS to start or stop the monitoring of the service already provisioned. The integration between Service Automation and Measurementcollection is performed by defining a specific function “triggerMonitoring”, which is invoked by calling PUT methods on the “**/assignmgmt/{service_id}**” and **“/releasemgmt/{service_id}”** endpoints defined as part of the Resource Configuration process.

During the service provisioning phase, the “triggerMonitoring” function is called after updating the service information in ServicesDB, in order to start monitoring the provisioned service. To start monitoring this specific service, Service Automation MS calls via a POST method the **"/datasources/MDP/"** endpoint on Measurement collection MS.

In the case of the decommissioning phase, the "triggerMonitoring" function is called after updating the service information, to stop the monitoring of the already decommissioned service. To stop the monitoring, Service Automation first calls via GET method the **"/datasource/MDP?service_id={service_id}"** endpoint to get the datasource_id linked to the specific service. Once obtained the datasource id, Service Automation calls via DELETE method **"/datasources/MDP/id/{datasource_id}"** endpoint, which stops the monitoring.



REST Interface Endpoints - Provisioning
PUT /assignmgmt/{service_id} -> **POST** /datasources/MDP

REST Interface Endpoints - Decommissioning
PUT /releasemgmt/{service_id} -> **GET** /datasource/MDP?service_id={service_id} -> **DELETE** /datasources/MDP/id/{datasource_id}

Figure 2-27. Service Automation – Resource Configuration Integration Workflow

The integration of both MS has been tested in an emulated FastAPI + Swagger environment where all endpoints have been called to test the performance of the API REST Interfaces.

Parameter	Description
ID	INTEGRATION_TEST_SERVICE_AUTOMATION_DATA_COLLECTION
Description	Internal test performed to validate the integration and communication between MD Service Automation and Measurement collection MS. This integration is implemented as part of two information flows: i) trigger or

	stop the measurement collection of the service once the service provisioning or decommissioning has been completed, and ii) return the confirmation of the measurement collection triggering or detention from the Measurement collection MS towards the Service Automation. In both cases, the Service REST Interface is used to ensure the communication between both MS. To perform this test, the Service Automation MS and a mock-up version of the Data Collection and Management MS have been implemented.
Management domain services	Service Automation Measurement collection MS
Technology domain (TD)	Ethernet TSN domain
Specific MD Interface & methods	**Interface:** Service REST Interface defined as part of the MD Service Automation MS. **Endpoints called by MD Service Automation:** **POST** /datasources/MDP/ with body containing (Service_ID, QoS and Traffic Characteristics) **GET** /datasource/MDP?service_id=*service_id* **DELETE** /datasources/MDP/id/*datasource_id*
Reference to TD interfaces	None
Result	SUCCESSFUL The Service Automation MS was able to send and retrieve data from the Data Collection and Management mock-up.
Date	07/2024
Observations and comments	This test needs to be extended in the future to verify the integration with the real implementation of the Data Collection and Management MS.

Table 2-16. Service Automation – Data Collection and Management Integration Test

Resource Configuration

In Ethernet TSN networks, the Resource Configurator adjusts network configurations dynamically, using IEEE 802.1Qbv [6] standard based on the WCTT analysis, ensuring compliance with the required KPIs. Once the network simulation identifies the optimal configuration, it is exported in Yang XML format and uploaded to Ethernet TSN devices using the NetConf [7]protocol. This process enables the real-time adjustment of network resources and schedules, allowing the system to maintain high performance and reliability.

Ethernet Resource Configurator use the RTAW-Pegase API [8] to compute the optimal configuration. When the Resource Configurator receives a service provisioning request (see Figure 2-28), via HTTP Rest interface, it loads the API into a JAVA script module. At the time, the script retrieves the topology and the routing for setting up the simulation. The characteristic of the flow/s are then processed to start with the simulation. This process involves another JAVA script that works in another thread, listening in case a new service provisioning request comes. When the simulation finish, the same thread handling the flows flash up the YANG XML into the ethernet switches, via a REST HTTP interface at them and using NetConf [7] protocol.

```
        {
                "vlan_id": "2",
                "type": "TSN",
                "name": "Flow1",
                "arrival_mode": "Periodic",
                "time": 20,
                "size": "500",
                "sender": "PC_Talker",
                "priority": "7",
                "creation_type": "Unicast",
                "receivers": ["PC_Listener"],
                "latency_constraint": "1"
        },
        {
                "type": "BE",
                "name": "Flow2",
                "arrival_mode": "Periodic",
                "time": 10,
                "size": "1000",
                "priority": "1",
                "sender": "PC_Talker",
                "creation_type": "Unicast",
                "receivers": ["PC_Listener"],
                "latency_constraint": "100"
        },
```

Figure 2-28 Service provisioning input to Resource Configurator.

Table 2-17 and Table 2-18 showcase the two main tests involved in the process of integrating these modules. Internal test, such as WCTT calculation and the computation of the optimal configurations, where performed before by the RTAW-Pegase proprietaries.

Parameter	**Description**
ID	SERVICE_PROVISIONING_REQUEST_PROCESSED
Description	This test confirms the proper creation of a thread to process all flows received from a single service provisioning request
Management domain services	Resource Configuration
Technology domain (TD)	Ethernet TSN domain
Specific MD Interface & methods	POST /provisionservice/,
Reference to TD interfaces	This test used a TD interface referenced at the text
Result	SUCCESSFUL The Resource Configurator is able to process each flow.
Date	06/2024
Observations and comments	No observations

Table 2-17 Service provisioning request processed by the Resource Configurator test

Parameter	**Description**
ID	EXPORT_XML_TO_DEVICES
Description	This test confirms the proper export to the devices involved
Management domain services	Resource Configuration
Technology domain (TD)	Ethernet TSN domain

Specific MD Interface & methods	POST /yangxmltodevice/,
Result	SUCCESSFUL The Resource Configurator is able to export the configurations and send them to the devices
Date	07/2024
Observations and comments	No observations

Table 2-18 Configuration export to the devices.

Path Computation

The Local Path Computation MS is integrated in this PoC and is responsible for calculating the path or paths that will support the requested connectivity service. To this end, the Path Computation collects data plane topological information from the Resource Configuration MS. As explained before, the Service Automation MS requests the path computation through a REST interface, and the Path Computation MS replies to such request with the computed route whose feasibility will be afterwards assessed by the Digital Twin.

Measurement collection

The Data Collection and Management (of the Measurement collection MS) integration tested in this scenario follows the same approach of the one described in Section 0 .The main difference is in this case the metrics from the infrastructure are retrieved through a Data Collector which implements a HTTP REST client for the retrieval of the data. The monitoring agent on the TD side, implements a HTTP server agent following a custom API based on the 3GPP TS 29.122 [9]. The specification of the API is available in OpenAPI format in [10].. For the scope of this integration, an emulated infrastructure agent was used following the same API specification and generating emulated data for the requested service metrics.

In this integration test, the request for the creation of the data source and the configuration of the access control policies are performed automatically and transparently by the Service Automation component (as part of the workflow depicted in Figure 2-29). The verification of the successful integration was performed inspecting the data available in the internal InfluxDB using Grafana, as shown in Figure 2-30, and the data pushed into the Kafka message bus for near-real time consumption, depicted in Figure 2-29. It is worth to highlight, as shown in the figures, the identifier of the service is used for the establishment of the Kafka bus topics and for the names of the InfluxDB buckets.

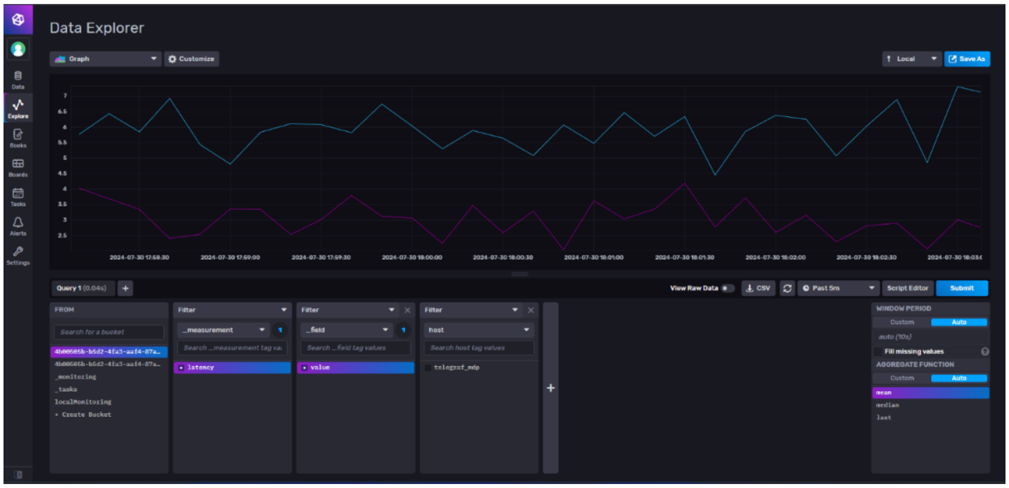
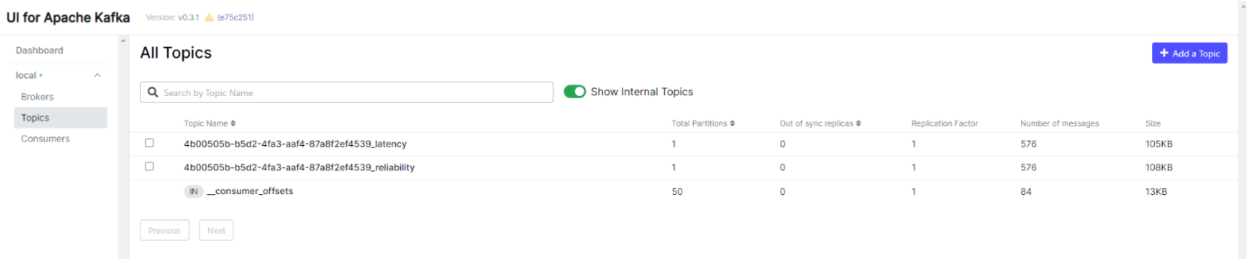


Figure 2-29. Ethernet TSN monitoring data in InfluxDB



Topic Name	Total Partitions	Out of sync replicas	Replication Factor	Number of messages	Size
☐ 4b00505b-b5d2-4fa3-aaf4-87a8f2ef4539_latency	1	0	1	576	105KB
☐ 4b00505b-b5d2-4fa3-aaf4-87a8f2ef4539_reliability	1	0	1	576	108KB
IN __consumer_offsets	50	0	1	84	13KB

Figure 2-30. Ethernet TSN Monitoring data in Kafka bus

In Table 2-19 and Table 2-20 we provide the report of the integration test for this specific component.

Parameter	**Description**
ID	DATA_COLLECTION_CONFIGURATION_ETHERNET_TSN

Description	This test confirms the proper creation of the collector plugins to retrieve the monitoring data associated to a specific service from a e Ethernet TSN TD.
Management domain services	Measurement collection
Technology domain (TD)	Ethernet TSN domain
Specific MD Interface & methods	data_collection.configuration
Reference to TD interfaces	This test used a TD monitoring agent implementing the API defined in [10]
Result	SUCCESSFUL The Data Collection and Management module was able to retrieve data from an emulated agent stub.
Date	07/2024
Observations and comments	No observations

Table 2-19. Ethernet TSN Data Collection Configuration test report

Parameter	**Description**
ID	DATA_COLLECTION_RETRIEVAL_ETHERNET_TSN
Description	This test confirms the proper retrieval of the monitoring data associated to a specific service from a e Ethernet TSN TD.
Management domain services	Measurement collection
Technology domain (TD)	Ethernet TSN domain

Funded by the European Union

Specific MD Interface & methods	data_collection.retrieval
Reference to TD interfaces	This test used a TD monitoring agent implementing the API defined in [10]
Result	SUCCESSFUL • The data from the Ethernet TSN domain was retrieved from the Data collection and management module using the InfluxDB querying interface and the Kafka bus. • The return codes from the northbound API matched the expected ones.
Date	07/2024
Observations and comments	No observations

Table 2-20. Ethernet TSN Data Collection retrieval test report

2.2.7 Validation of Data Unit Groups Implementation

This section describes the validation of the Data Unit Groups (DUGs) concept presented in D2.2 [4].

Validation Methodology

Following the detailed description in D2.3 [11] of the implementation using Express Data Paths (XDP) hooks as part of the extended Berkley Packet Filter (eBPF) implementation of the Linux community, this section now focuses on the validation methodology and initial results of leveraging XDP to implement a DUG-enabled router. Figure 2-31, below, depicts two nodes with the hostnames nuc34 and nuc239 and are configured as shown. The XDP object is loaded into interface eth0 on the nuc34 node. Both interface eth0 and eno1 are configured with IPv6 addresses and Iperfv6 is used to measure the impact of XDP on UDP and TCP network performance.

Figure 2-31. Validation Set-Up for Data Unit Group Implementation

Iperf was configured as followed on the server:

```
iperf –s –V –i 1
```

Where the arguments have the following meaning:

- -s: Acting as the iperf server to send traffic to clients
- -V: Iperf operating with IPv6
- -i 1: Report network performance metrics every second

For the client, iperf was configured as followed:

```
iperf –c fe80::56b2:3ff:fe19:c5ae%eth0 –V –i 1 –t 80
```

Where the arguments have the following meaning:

- -c: Act as the iperf client followed by the IPv6 address of the server
- -V: Iperf operating with IPv6
- -i 1: Report network performance metrics every second
- -t 80: Run an experiment for 80s

The reason why 80s are chosen is to provide average values for a 60s-long experiment where the first 10 and last 10s of the 80s-long experiment are removed from the data set to allow the system to operate under stable conditions with no impact of any rate control or non-network resource scaling to adjust to the newly added system load.

In addition to testing UDP and TCP using iperf, the following methodology was applied. Network performance can be measured by transmitting and receiving data between devices using iperf in user space. Together with XDP hook loaded into the kernel, high-performance packet processing is enabled right at the network interface level. The question to answer first was the impact of XDP on the overall network performance compared to no XDP hook loaded. This has been achieved by developing two XDP hook variants and explained here in further detail what functionality was implemented:

No XDP Hook: iperf uses the conventional network stack to measure TCP and UDP performance without any XDP hook present.

XDP Pass: A loaded XDP software is set up to pass packets to the conventional network stack without alteration. With this configuration, the overhead caused by simply installing the XDP hook can be compared.

XDP with Data Unit Group (DUG) Logic: In this instance, the Data Unit Group (DUG) logic is used by the XDP to manage packets. By using logic to control data unit groups before sending packets to the conventional network stack or other processing channels, the XDP program processes packets at the network interface level.

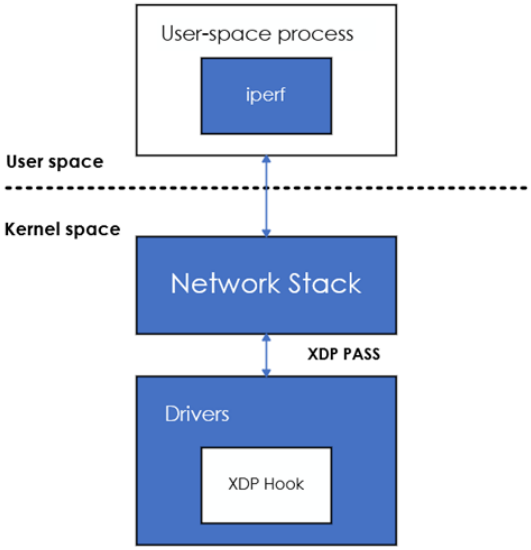


Figure 2-32. Placement of XDP Hook Implementing Data Unit Group Concept in Linux Environment

Each of these scenarios can be compared to see how XDP impacts network performance. The three scenarios—the No XDP Hook scenario acts as a baseline, the XDP Pass scenario illustrates the overhead associated with implementing XDP, and the XDP with DUG Logic scenario exhibits the performance measurement possible through network interface-level packet processing.

Validation Results

Given that UDP is connectionless and eliminates the overhead of creating and maintaining connections, it typically performs better than TCP when tested using iperf with no XDP, XDP Pass, and XDP with DUG (Data Unit Group) logic. There is a small but discernible increase in bandwidth and stability in the values in scenarios with XDP Pass and XDP with DUG when compared to no XDP. This is because, even with minimal gains in bandwidth, XDP enables early packet processing at the network interface level, lowering latency. Figure 2-33. below illustrates the average performance results for TCP and UDP, each measured over 23 iterations over 60s of independent measurements.

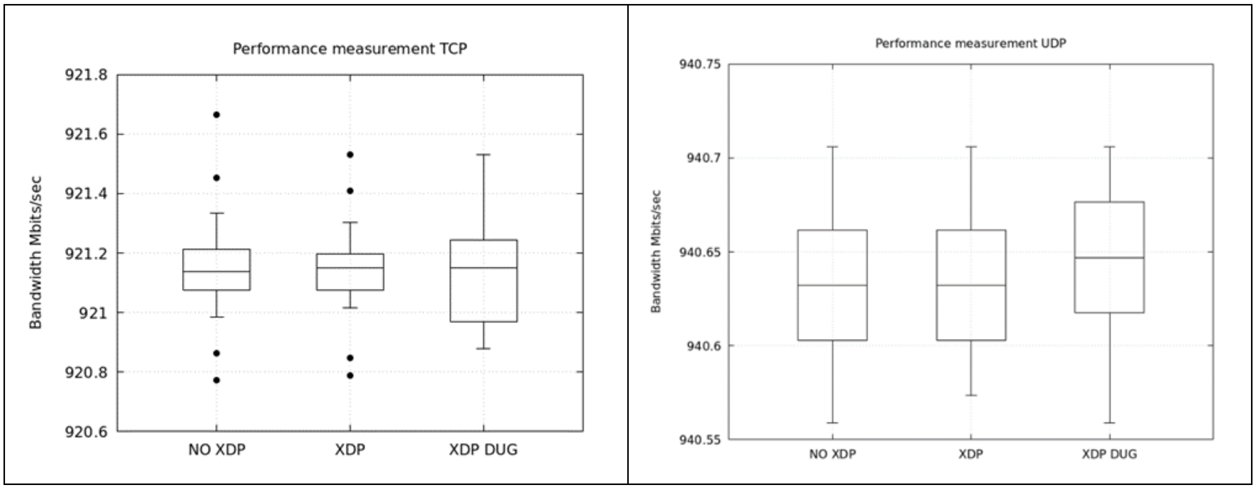


Figure 2-33. Performance Comparison Results for XDP Using TCP and UDP – Iperf Averages

Given UDP has less overhead and is connectionless, it typically performs better than TCP. Furthermore, because UDP doesn't require the same connection management procedures as TCP, its performance figures are typically more stable. Alternatively, TCP performance holds like earlier findings, with its built-in overhead resulting in a somewhat lower and more variable performance than UDP. Although the advent of XDP—whether in Pass or with DUG logic—helps to further stabilize these values, the performance characteristics of TCP and UDP remain significantly distinct. Figure 2-34 below illustrate the raw iperf data plotted each measured over 23 iterations within an 80-second time limit.

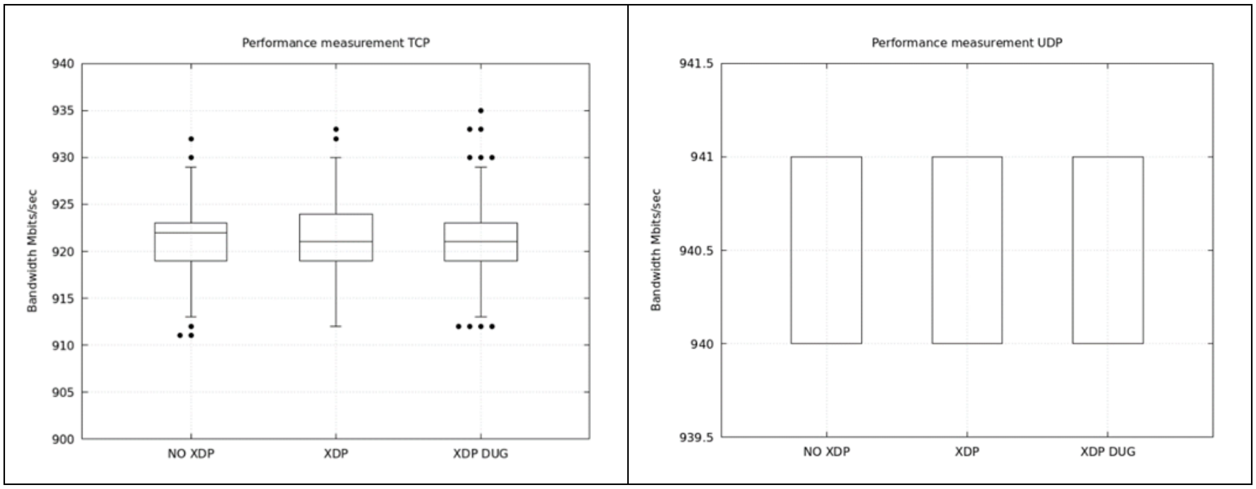


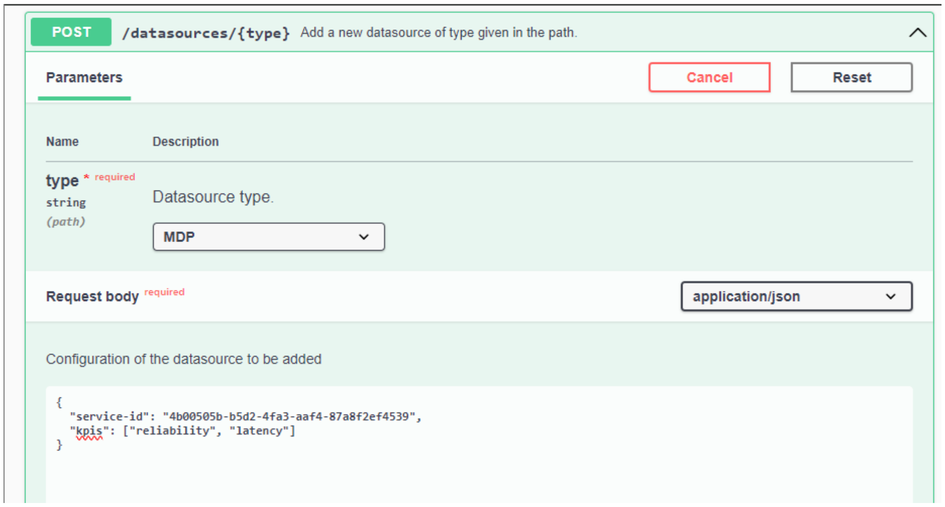
Figure 2-34. Performance Comparison Results for XDP Using TCP and UDP – Aggregated per Second Throughput Numbers across all Experiments

2.2.8 Sensing-Enabled 3GPP Domain Data Collection and Management Integration

The Data Collection and Management, as described in D3.2 [12] is the AICP component responsible for the retrieval and storage of the monitoring metrics required for the service. This component supports a northbound management interface enabling the dynamic creation and configuration of data sources (based on the specific technology domain), and the dynamic creation of monitoring jobs which specify the metrics required and the access credentials. The module uniforms data collected from distinct and diverse data sources and exposes them both in a near real-time fashion and in time series through a northbound query interface, The integration with the Sensing-enabled 3GPP domain targeted the verification of the automated collection of the latency and sensing information from the domain.

The setup used for this integration comprises a Localisation and Sensing network, used in the scope of the additional use cases targeted by the PREDICT-6G described in D1.1 [13], running at Interdigital laboratories and two virtual machines (VM) running at Nextworks premises. One VM runs an InfluxDB where all the information coming from the sensing enabled 3GPP domain is pushed, and the other VM contains the software components of the Data Collection and Management.

The integration tests executed verified the correct operation of the data source creation and establishment of the per metric access control credentials using the REST-based northbound API (we refer to D3.2 [12] for further details regarding the API exposed by this module). In Figure 2-35 we depict the request used for the data source creation. As mentioned previously, the data source creation request allows the specification of the metrics to be collected and the service instance identifier, which is used to label the incoming data. This request triggers the internal logic which creates a data collector container, which retrieves the data from the InfluxDB where the monitoring information is pushed.



`/datasources/{type}` Add a new datasource of type given in the path.

type * required
string
(path)

Datasource type.

MDP

Request body required

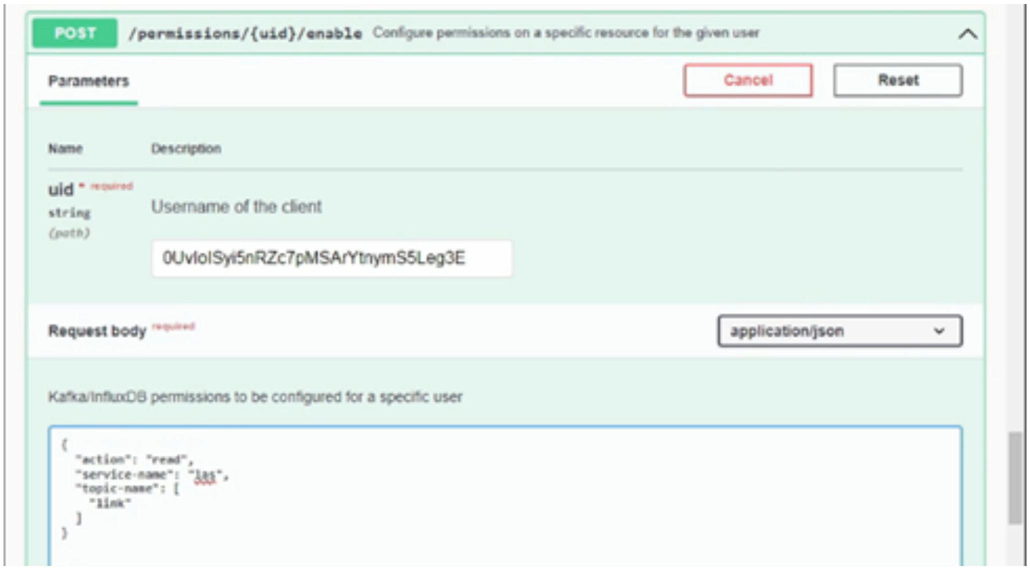
application/json

Configuration of the datasource to be added

```
{
  "service-id": "4b00505b-b5d2-4fa3-aaf4-87a8f2ef4539",
  "kpis": ["reliability", "latency"]
}
```

Figure 2-35. Data Collection and Monitoring data source creation request

In Figure 2-36 we depict the request used for establishing the access control policies applied to a user. The example in Figure 2-35 enables the specified user to read from a Kafka topic where the data is pushed for near-real time consumption of data, and for the querying of the stored data.We refer to D3.2 [12] for more details regarding the access control mechanisms of the Data Collection and Management module.



Username of the client

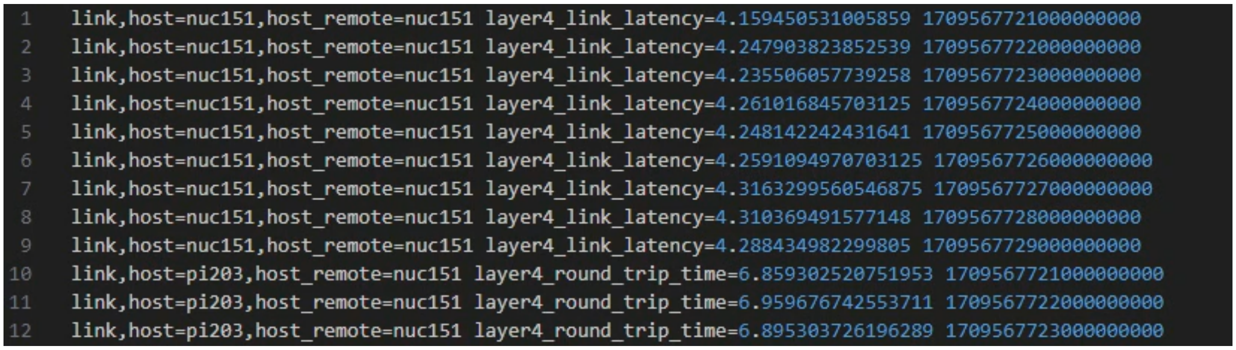
0UvloISyi5nRZc7pMSArYtnymS5Leg3E

Kafka/InfluxDB permissions to be configured for a specific user

```
{
  "action": "read",
  "service-name": "las",
  "topic-name": [
    "link"
  ]
}
```

Figure 2-36. Data Collection and Monitoring access control establishment request

The successful integration was confirmed using the near-real time and querying interfaces to retrieve data using different user credentials. In we provide a screenshot of the information retrieved from the sensing enabled 3GPP domain and available to be consumed on the Data Collection and Management module. The report of the integration in Table 2-21 summarizes the outcomes of this integration.



```
link,host=nuc151,host_remote=nuc151 layer4_link_latency=4.159450531005859 1709567721000000000
link,host=nuc151,host_remote=nuc151 layer4_link_latency=4.247903823852539 1709567722000000000
link,host=nuc151,host_remote=nuc151 layer4_link_latency=4.235506057739258 1709567723000000000
link,host=nuc151,host_remote=nuc151 layer4_link_latency=4.261016845703125 1709567724000000000
link,host=nuc151,host_remote=nuc151 layer4_link_latency=4.248142242431641 1709567725000000000
link,host=nuc151,host_remote=nuc151 layer4_link_latency=4.2591094970703125 1709567726000000000
link,host=nuc151,host_remote=nuc151 layer4_link_latency=4.3163299560546875 1709567727000000000
link,host=nuc151,host_remote=nuc151 layer4_link_latency=4.310369491577148 1709567728000000000
link,host=nuc151,host_remote=nuc151 layer4_link_latency=4.288434982299805 1709567729000000000
link,host=pi203,host_remote=nuc151 layer4_round_trip_time=6.859302520751953 1709567721000000000
link,host=pi203,host_remote=nuc151 layer4_round_trip_time=6.959676742553711 1709567722000000000
link,host=pi203,host_remote=nuc151 layer4_round_trip_time=6.895303726196289 1709567723000000000
```

Figure 2-37. Sensing enabled 3GPP metrics retrieval from the Data Collection and Management

Funded by
the European Union

Parameter	Description
ID	DATA_COLLECTION_CONFIGURATION_SENSING_3GPP
Description	This test confirms the proper creation of the collector plugins to retrieve the monitoring data associated to a specific service from a e Sensing enabled 3GPP domain.
Management domain services	Data collection and management
Technology domain (TD)	Ethernet TSN domain
Specific MD Interface & methods	data_collection.configuration data_collection.retrieval
Reference to TD interfaces	This test used an InfluxDB where the infrastructure elements of the Sensing enabled 3GPP domain pushed information [10]
Result	SUCCESSFUL The Data Collection and Management module was able to retrieve data and enforce the access control policies for the data retrieval.
Date	01/2024
Observations and comments	No observations

Table 2-21. Sensing enabled 3GPP Data Collection integration report

3 Updated integration and roadmap

The integration and validation activities for PREDICT-6G system were planned in D4.1 [1] on a 3-month basis cycles. Starting from January 2024 (M13) with the first cycle, the current deliverable reports the progress at the end of the 3third cycle of integration and validation activities.

The methodologies used were based on Agile methodologies combined with Continuous Integration/Continuous Deployment (CI/CD) and incremental validation. Each cycle was divided in 3 SCRUM sprints. The CI/CD was based on the usage of Git and Git pipelines.

Figure 3-1 lists the planned activities and highlights the cycle the closed at M21.

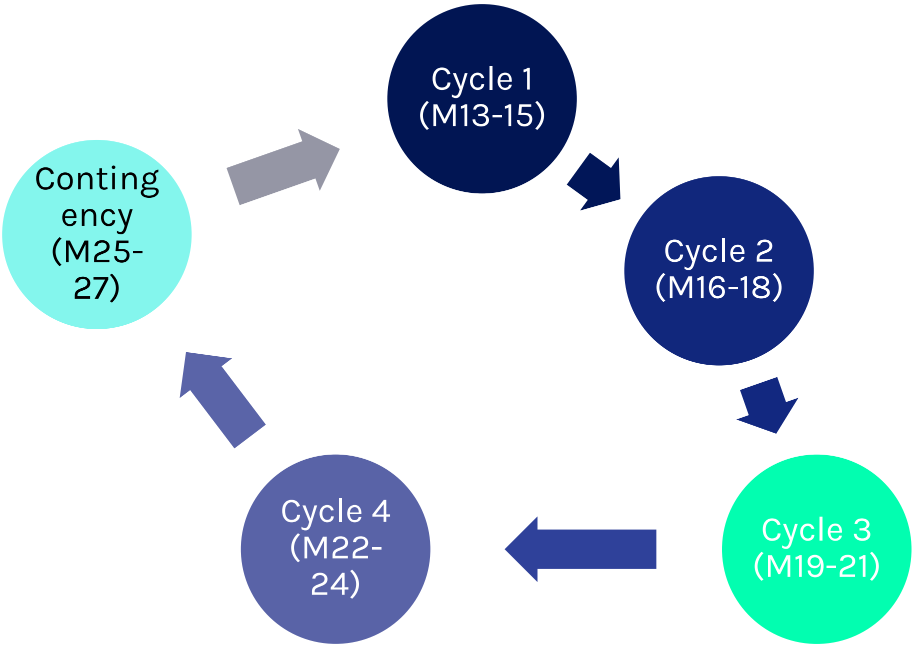


Figure 3-1: Updated roadmap of integration and validation activities

During the first three cycles PREDICT-6G partners involved in WPs 2, 3 & 4 have worked closely to deliver, integrate, and deploy the functionalities of PREDICT-6G system. Each Cycle runs in parallel with the development cycles described in D3.2 [12].

The activities performed during M13-M21 include:

- detailed presentation of technical activities, scheduling of releases and analytical framework for the development, integration and deployment of PREDICT-6G system (Cycle 1)

- test plan for all use cases and the release of the fist versions of data plane and control plane components (Cycle 2)
- activities referring to integration tools both data plane and control please components (Cycle 3).

Table 3-1 below summarized the list of components (and functionalities) and the validation tests performed for each PoC.

Component, functionalities	Test performed	PoC	Open Lab
Time Sync Interfaces to AICP	E2E Time Synch integration in Nokia Open Lab	Deterministic services for critical communications	Nokia Open Lab
Measurement collection Interfaces to AICP	E2E monitoring integration test in Nokia Open Lab	Deterministic services for critical communications	
Service Automation General TSN technologies 3GPP TSN	Integration test Smart Factory with 5G commercial	Smart Factory	5TONIC Lab
Service Automation General TSN technologies 3GPP TSN	Integration test Smart Factory with 5G commercial results	Smart Factory	
Service Automation General TSN technologies 3GPP TSN	Integration test Smart Factory with 5G TSN enabled network	Smart Factory	
Service Automation General TSN technologies 3GPP TSN	Integration test Smart Factory with 5G TSN enabled network results	Smart Factory	
Exposure Services - Integration of MDP	Integration test robot remote control across technology domains	Multi-Technology Multi-Domain DetNet-Based Integration of MDP and AICP	

Funded by the European Union

Resource Configuration	Integration results robot remote control across technology domains	Multi-Technology Multi-Domain DetNet-Based Integration of MDP and AICP	
Service Automation General TSN technologies	Validation test Smart Factory with 5G commercial	Smart Factory	
Service Automation General TSN technologies 3GPP TSN	Validation test Smart Factory with 5G TSN enabled network	Smart Factory	
Dynamic re-configuration of TSN-based networks	SERVICE_PROVISIONING_ETHERNET_TSN Test report	Multi-Technology Multi-Domain DetNet-Based Integration of MDP and AICP	5TONIC Lab
Path Computation	Service Automation – Path Computation Integration Test	Multi-Technology Multi-Domain DetNet-Based Integration of MDP and AICP	
Service Automation	Service Automation – Resource Configuration Integration Test	Multi-Technology Multi-Domain DetNet-Based Integration of MDP and AICP	
Service Automation	Service Automation – Data Collection and Management Integration Test	Multi-Technology Multi-Domain DetNet-Based Integration of MDP and AICP	
Resource Configuration	Service provisioning request processed by the Resource Configurator test	Multi-Technology Multi-Domain DetNet-Based Integration of MDP and AICP	5TONIC Lab
Resource Configuration	Configuration export to the devices.	Multi-Technology Multi-Domain DetNet-	

		Based Integration of MDP and AICP	
Digital twin - Assess TSN over Wi-Fi	Ethernet TSN Data Collection Configuration test report	Multi-Technology Multi-Domain DetNet-Based Integration of MDP and AICP	
Digital twin - Assess TSN over Wi-Fi	Ethernet TSN Data Collection retrieval test report	Multi-Technology Multi-Domain DetNet-Based Integration of MDP and AICP	
Data Collection and Management	Sensing enabled 3GPP Data Collection integration report	Localisation and Sensing	

Table 3-1. List of tests performed per PoC

The results of the final release will be included in D4.3 (due in M30) and the activities will focus on the delivery of the integrated prototype and validating it is components.

4 Conclusions

This report presents the first version of the PREDICT-6G integrated prototype and the initial outcomes of WP2 and WP3 tools deployed in the two Open Labs are included in this report.

The integration activities focusing on testing the behaviour of the AICP and the enhanced user plane activities are detailed in this report. Thus, in the four demonstrations (PoCs) anticipated the capabilities of the PREDICT-6G system are tested and the results are presented.

In next period, all PREDICT-6G components and functionalities will undergo validation in the Open Labs. The results of the validation will be included in D4.3 (M30).

5 References

[1] PREDICT-6G Consortium, “D4.1 Integration and validation plans and Open labs design,” 2024.

[2] PREDICT-6G Consortium, “D1.2 PREDICT-6G framework architecture and initial specification,” 2023.

[3] PREDICT-6G Consortium, “D3.1 Release 1 of AI-driven inter-domain network control, management, and orchestration innovations,” 2023.

[4] PREDICT-6G, “D2.2: Implementation of selected release 1 PREDICT-6G MDP innovations,” 2023.

[5] PREDICT-6G Consortium, “DMP-OpenAPI,” 2024.

[6] IEEE, “Std 802.1Q-2014.,” 2015.

[7] R. E. a. M. B. a. A. B. a. J. Schönwälder, “Network Configuration Protocol (NETCONF),” RFC Editor, 2011.

[8] RealTime-at-Work, “RTaW-Pegase,” 2024.

[9] ETSI, “3GPP TS 29.122 version 16.7.0 Release 16,” 2020.

[10] OpenAPI, “OpenAPI Specification v3.1.0,” 2021.

[11] Predict-6G Consortium, “D2.3 Release 2 of PREDICT-6G MDP,” 2024.

[12] PREDICT-6G Consortium, “D3.2 Implementation of selected release 1 AI-driven inter-domain network control, management and orchestration innovations,” 2024.

[13] PREDICT-6G Consortium, “D1.1 Analysis of use cases and system requirements,” 2023.

Funded by
the European Union

6 Appendixes

n/a