D3.1

Release 1 of AI-driven inter-domain network control, management, and orchestration innovations

NXW

Funded by
the European Union

Revision v0.1

Work package	WP3
Task	T3.1, T3.2, T3.3, T3.4
Due date	30-09-2023
Submission date	30-09-2023
Deliverable lead	NXW
Version	V1.0
Authors	Pietro G. Giardina, Matteo Ravalli, Giada Landi (NXW) Luis Velasco, Marc Ruiz, Davide Careglio, Salvatore Spadaro, Fernando Agraz (UPC) José Luis Cárcel (ATOS) Rafael Rosales (INT) Sebastian Robitzsch, Renan Krishna (IDE) Constantine Ayimba (UC3M) Péter Szilágyi, Tamás Kárász, Szabolcs Nováczki, Zoltán Vincze, Csaba Vulkán, Gergely Hajgató (NOK) Carla Fabiana Chiasserini, Somreeta Pramanik (POLITO) Marta Blanco, Jesús Folgueira, Rafael Cantó (TID)
Reviewers	Marc Mollà (ERC) Valerio Frascolla (INT) Antonio De La Oliva (UC3M)

Abstract

Given the relevancy time-critical services are acquiring for different vertical sectors, standards bodies (such as IETF and IEEE) started working on network architectures capable to support the deterministic and reliable data transmission. Some solutions have been emerging although bounded to specific technological domains and characterized by limitations in terms of interoperability, preventing the provisioning of deterministic service across multiple domains (technological and administrative). In this document, the initial functional design of the AI-driven multi-stakeholder inter-domain control plane (AICP), the PREDICT-6G control plane solutions to seamlessly integrate a data plane building of a set of multi-technology deterministic domain, is discussed. AICP relies on 4 key technological enablers: i) AI/ML, for smart resource allocation, ii) Digital Twins, for system and (deterministic) service' KPI predictably, iii) Data Collection and Management, for continuous monitoring of the services, and iv) Inter-domain orchestration for the management of E2E deterministic service lifecycle. The topic of how to represent the different technologies, building the underlying networks, inside the AICP is also discussed, as that is a topic that can be considered crucial for the proper lifecycle management of E2E deterministic services.

Keywords

Network Determinism, Predictability, Reliability, Time-sensitiveness, TSN, DetNet, RAW, Control Plane, Cross-domain, Data and Network modelling

Document revision history

Version	Date	Description of change	Contributor(s)
V0.1	08-05-2023	Initial ToC	NXW
V0.2	31-08-2023	First complete draft	WP3
V0.3	11-09-2023	Document ready for internal review	NXW, NOK, UPC
V0.4	22-09-2023	Reviewed version (internal)	Internal reviewers
V1.0	29-09-2023	Official delivery version	WP3

Disclaimer

Copyright notice

Document information

Nature of the deliverable		[R]
Dissemination level		
PU	Public, fully open. E.g., website	✓
CL	Classified information as referred to in Commission Decision 2001/844/EC	
SEN	Confidential to PREDICT-6G project and Commission Services	

* Deliverable types:

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

DEM: demonstrator, pilot, prototype, plan designs.

DEC: websites, patent filings, press and media actions, videos, etc.

OTHER: software, technical diagrams, etc.

Table of Contents

List of figures

List of tables

Acronyms and definitions

3GPP	Third Generation Partnership Project
AICP	AI-driven Multi-stakeholder Inter-domain Control-Plane
AI/ML	Artificial Intelligence/ Machine Learning
BGP-LS	Border Gateway Protocol – Link State
DetNet	Deterministic Networking
DoA	Description of Action
DT	Digital Twin / Digital Twinning
E2E	End-to-End
IP	Internet Protocol
LLDP	Link Layer Discovery Protocol
MD	Managed Domain
MDP	Multi-domain Data-Plane
MF	Management Function
MOPS	Millions of Operations Per Second
MS	Management Service
QoS	Quality of Service
SDO	Standards Developing Organization
WP	Work Package

Table of partners

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

1 Executive summary

Time-critical services are becoming more and more relevant for several vertical services, in line with the vision of Industry 4.0, which requires deterministic and reliable information delivery mechanisms for a full industry digitalization. Several Standard Developing Organizations (SDOs) such as IEEE and IETF, are working to standardize system architectures capable to fulfil the need of the different verticals sectors: as a consequence, some solutions targeting the creation and management of deterministic networks are progressively emerging. Such solutions are, nevertheless, characterized usually bounded to specific technological domains, a characteristic that creates limitations in their interoperability and prevents the possibility to provision cross-domain end-to-end (E2E) deterministic services. The solution proposed by PREDICT-6G is to design and implement a specific control plane architecture, AICP (AI-driven multi-stakeholder inter-domain control plane), capable to seamlessly interact with a data plane integrating heterogenous multi-technological and multi-administrative domains. AICP relies on 4 key technological enablers: i) AI/ML, for smart resource allocation, ii) Digital Twins (DTs), for system and deterministic service Key Performance Indicator (KPI) predictably, iii) Data collection and management, for continuous monitoring of the services, and iv) Inter-domain orchestration for the management of E2E deterministic service lifecycle.

The PREDICT-6G vision for AICP is that of a hierarchical control plane, with several local control platforms extending and integrating existing technological control planes, supervised by an E2E control platform for the management of E2E determinist services.

This document discusses WP3 investigations on AICP and initial outcomes, which represent the key results of the deliverable:

- *AICP functional architecture design.* The control architecture follows a service-centric approach, where all the functionalities are modelled as loosely coupled services, that provide a high-level of flexibility and scalability. Services are logically grouped per technological enabler.
- *Network and Data abstraction.* The AICP is designed to integrate and control different technologies. This requires that each network and the related data collected for monitoring purposes must be normalized to common models in order to provide a unified view to the E2E control platform.

2 Introduction

The continuous advancement on network technologies and the progressive digitalization of the industries generates an increasing interest of different vertical sectors around time-critical services and applications. Although some technical solutions are already available, the different level of maturity and the numerous technologies employed create limitation in their interoperability, at both technical and administrative level. In this document, a technical (or technological) domain is a piece of network characterized by one dominant technology (e.g., TSN), while an administrative domain is a piece of network administrated by a single entity (e.g., Telco operator). An administrative domain can employee multiple technologies. Given the level of complexity and heterogeneity that the interconnected domain can reach, the possibility of deploying E2E deterministic services, and their lifecycle management in general, is a challenging problem that needs to be solved at the management and control plane.

This document describes the initial design of the PREDICT-6G AICP, which aims to fill the current gap that prevents the provisioning and maintaining of E2E deterministic services across heterogenous technical and administrative domains. The key concepts characterizing the AICP, and its architecture design are discussed in section 3, including the set of current network technological domains planned to be covered: 3GPP, DetNet/IP and Wi-Fi. The central part of the document (sections 4, 5, 6, and 7) describes the AICP key technological enablers, and their set of functionalities modelled as services, in line with the concept of service-centric (or service-based) architectures. The AI/ML, for smart resource allocation and predictability, is discussed in section 4 while, section 5 describes the concept of DT and its usage to make prediction on the trend of determinist services' KPIs. Section 6 describes the mechanism enabling monitoring and data collection, for the continuous monitoring of service's parameters used to feed both AI/ML processes and DTs, the two make predictions/take decisions on if/how to reconfigure the underlying system to guarantee E2E service determinism and reliability. Such decisions are enforced frictionless by specific inter-domain orchestration and network control services, described in Section 7. Section 8 explains how the AICP modules are mapped on the technological domains and abstracts the different network technologies and the related data, providing also initial thoughts on the possibility of modelling network domain that are completely non-deterministic. Finally, section 9 reports examples of operation workflows, with the scope of clarify possible interactions between the different services during the execution of common operations, e.g., E2E deterministic service provisioning.

3 AICP in a nutshell

The PREDICT-6G system is designed to operate across multiple technological and/or logical domains, which must be functioning in a cooperative manner to fulfil the deterministic requirements against the E2E communication. Here a technological domain refers to a network segment that implements network functions (user-plane, control-plane, management-plane) according to a set of standards, such as 3GPP, or IETF DetNet. An administrative or logical domain is a network segment that is governed by one entity such as an operator. An end-to-end communication system may consist of multiple administrative domains that are implemented with the same network technology (e.g., two 3GPP networks), however due to their different governance and thus separate control/management interfaces, they are integrated to the PREDICT-6G system as separate domains. The interplay of domains implies the need for a higher control entity that manages the different technologies regardless of their standalone support of determinism. This responsibility is delegated to the AICP, which was introduced in (PREDICT-6G/D1.1/9, 2023) and is further described in this section.

3.1 Key Concepts

The AICP is designed with a service-centric approach wherein the management capabilities are realized by management services (MSs) and exposed through well-defined interfaces. Each MS must have a clear scope and can realize multiple capabilities to fulfil that scope. Moreover, each MS must be self-contained, i.e., operable on its own. The latter requirement fosters the extensibility of the PREDICT-6G system: the addition of a new MS cannot affect the operation of the already deployed ones.

The fact that MSs can be deployed dynamically into an already operating system also means that an implementation of the system does not necessarily include all the available MSs, but the AICP can start new services or rescale/terminate existing ones on demand. This nature guarantees the adaptability of the PREDICT-6G system architecture against on-the-fly changes of the communication environment.

Besides, exposing the system capabilities through interfaces allows to integrate new technological domains in a non-intrusive way, where the communication between the new technology domain and the PREDICT-6G system can be contained within the interface instead of making changes in the technological domain itself.

3.2 Architecture

The architecture of the PREDICT-6G system is depicted in **Figure 3-1** taken from (PREDICT-6G/D2.1/3, 2023) with a focus on the scope and the boundaries of the AICP and the multi-domain data plane (MDP). The managed entities are supervised by management functions (MFs) that are realizations of one or more MSs that are organized in management domains (MDs).

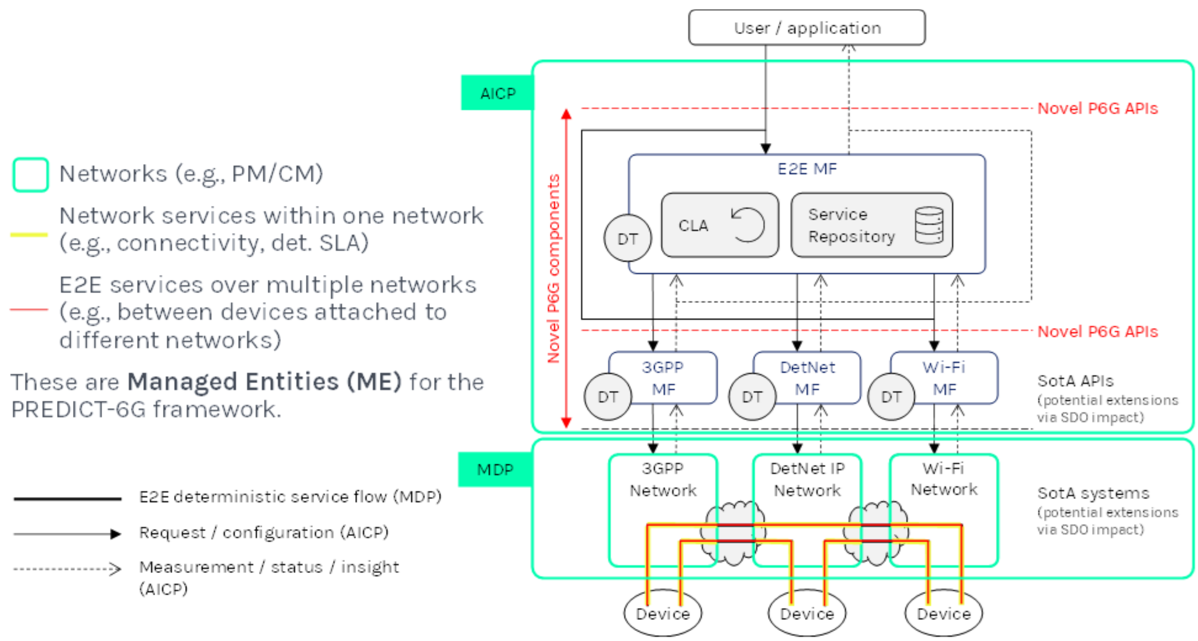

Figure 3-1 PREDICT-6G system architecture composed of the AICP and of the MDP.CLA: closed-loop automation.

The scope of a MS can be domain-specific or E2E, which implies the need for a hybrid, hierarchical-federated architecture for the AICP wherein both lateral and vertical relations are allowed between MSs as emphasized in **Figure 3-2**. Communication with MSs is possible through interfaces that must be partly unified inside the project architecture to support common tasks, e.g., MS discovery, MS registration, requesting for data exposure, providing health status on the service itself, etc. However, as MSs differ in nature, some parts of the interfaces are inherently different from each other. The latter holds to interfaces between MSs and managed entities (MEs) as well where the interface must adapt to the peculiarities of the specific ME.

From the viewpoint of the AICP, MDs join MSs depending on their scope, which can be only one among:

- Technology-specific
- E2E
- Inter-domain integration

E2E and inter-domain integration MDs must be unique in a PREDICT-6G system, but there can be as many technology-specific MDs as needed for the specific network deployment.

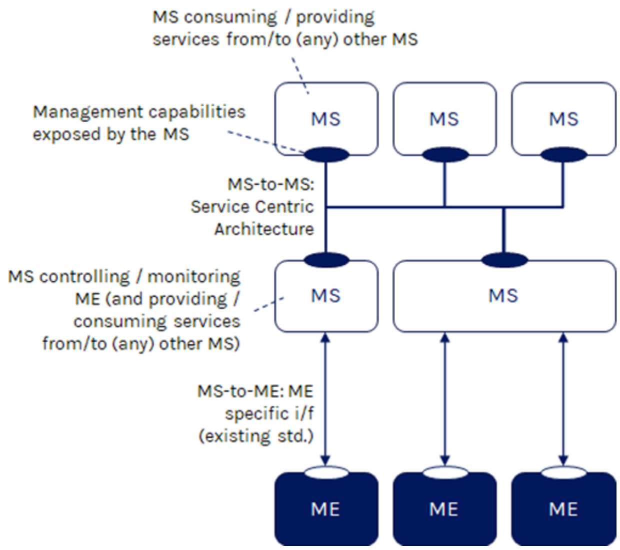

Figure 3-2 The hybrid architecture of the PREDICT-6G AICP: both lateral and vertical inter-MS relations are allowed depending on the role of the specific MS.

Technology-specific MDs also differ from the other ones in the fact that MSs in these domains have southbound interfaces tailored to the managed entity instead of utilizing the partly unified interface-scheme mentioned before. 3GPP, DetNet and Wi-Fi are three technology domains considered so far to be part of the PREDICT-6G system, but the PREDICT-6G system itself can be extended to other technological domains as well owing to the service-based architecture. As the alignment of the communication is all done through the ME-specific interfaces, the PREDICT-6G system is flexible enough, in principle, to provide control, performance, failure management, etc. even to yet not existing technological domains with the development of the corresponding MS and its interface.

The E2E MD joins all the MSs that provide capabilities for the creation and management of deterministic services spanning multiple technological domains and/or networks. Hence, these MSs interact with all the technology-specific MSs deployed in the PREDICT-6G system. Moreover, the MD is responsible to report the health status of the overall network and take preventive steps to avoid the degradation of the E2E service performance.

The Inter-domain integration MD provides capabilities for deploying and orchestrating the PREDICT-6G system components, such as the capability to register and discover the available MSs in run-time, and the deployment and initialization of required services to an operable state. The latter functionality plays a key role in setting up such MSs that must be aligned to the whole of the network. Such an example is traffic replication on disjoint paths, where plainly joining the standalone subsystem capabilities do not yield the expected E2E result; hence, additional requests must be settled with the integrated technological domains before starting the deterministic services. The MSs in this MD communicate with all the other MDs by nature.

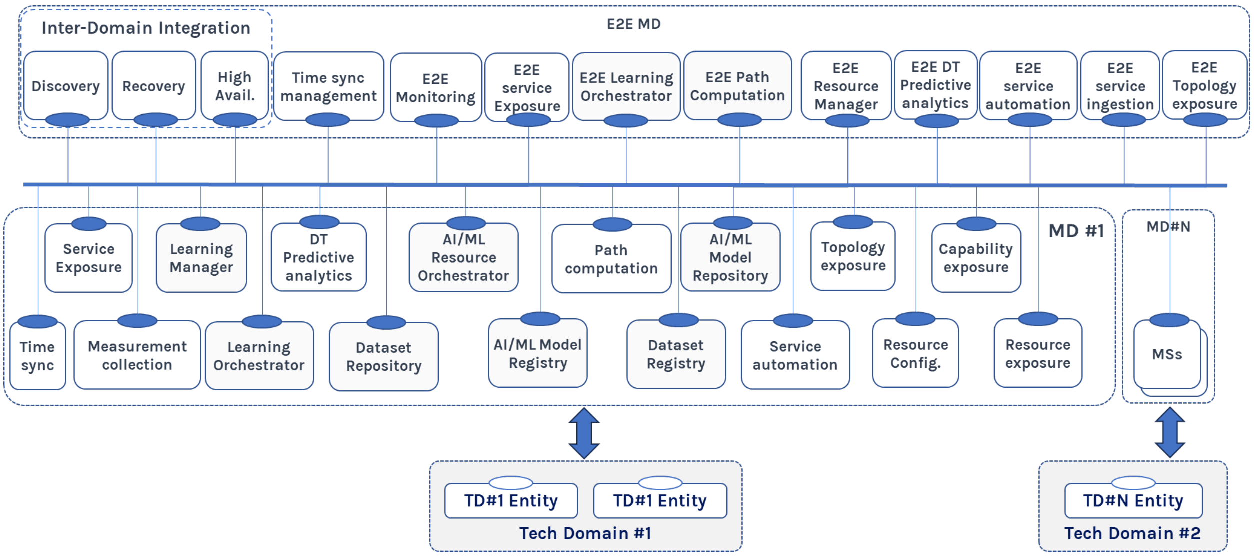

Figure 3-3 Relation between MDs and MSs.

The type of the MD defines not only the scope and the interfaces used in the MSs, but the relation compared to the other MDs as well. E2E and inter-domain integration MDs are on the top of the hierarchy while the technology-specific domains are on the bottom. The relation between the MDs is reflected on the contained MSs as well. MSs in the same MD are to fulfil the MD's scope in a federated manner implying that the MSs are on the same hierarchical level inside the MD. An example of a possible deployment is depicted in **Figure 3-3** with the various MDs highlighted.

3.3 Control loop automation

A distinguished set of services are responsible for maintaining the Service Level Agreement (SLA) inside the PREDICT-6G system forming a closed control loop (CCL). Conforming to the key concepts of the project, CCLs are designed to operate in an autonomous manner with minimal user intervention. Hence, user interaction is requested only when an unresolvable conflict arises. In any other cases the CCL must be able to:

1. forecast or detect the problem,
2. decide on the solution,
3. execute the solution.

without letting down the network traffic. Despite having a central CCL entity, the full responsibility of control is broken down to smaller ones, each having a dedicated CCL. With this approach, the 3 main deterministic measures (reliability, predictability, and time-sensitiveness) can be decoupled and can be maintained on their own even if other criteria are not met with the SLA.

As a principle, every conflict must be resolved in the smallest domain possible, i.e., if a conflict can be resolved both in a technology-specific domain or with an E2E solution (like reconfiguring multiple technology domains), the former solution must be preferred. Hence, both technology-domain-specific CCLs and E2E CCLs are to be presented in the PREDICT-6G system.

The possibly different scopes of the different CCLs impose different requirements against the control performance. A set of control KPIs are proposed in (PREDICT-6G/D1.1/A, 2023) to measure that control performance. However, that list is more of a starting point for creating CCL-specific metrics rather than an exhaustive list of available control KPIs that must be implemented.

3.4 Service request

PREDICT-6G is targeted to achieve such a level of autonomous working that the initiation of an E2E deterministic service could be taken not only by an operator but even by an end-user itself. In any case, the main components of the service request are the:

a. Service endpoints,
b. QoS characteristics,
c. Traffic characteristics,
d. Traffic flow template,
e. Service lifetime.

All these components must be interpreted by the AICP.

Service endpoints can be in the same domain, different domains with the same type or different domains with different types. Regardless on the type of the domain, each endpoint must be clearly identifiable e.g., by an IP address or a Fully Qualified Domain Name (FQDN). Supporting determinism is not a must for E2E service endpoints but it is for inter-domain service endpoints, as depicted in **Figure 3-4**.

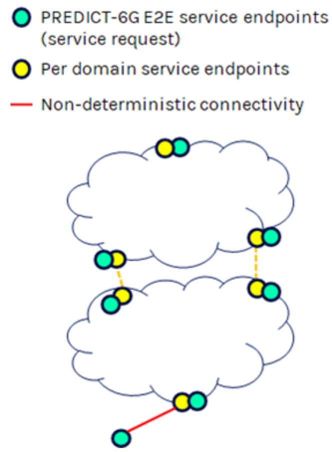

Figure 3-4 Supporting determinism is not a must for E2E service endpoints.

A group of QoS characteristics may reflect one of the main targets of the PREDICT-6G project, e.g., the characteristics describing reliability are depicted in **Figure 3-5**.

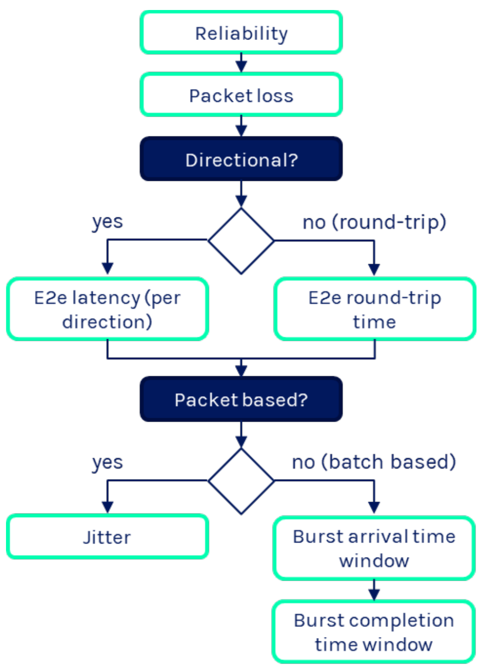

Figure 3-5 A group of QoS characteristics describing the system's reliability.

Traffic characteristics describe the direction and the periodicity of the flow as per **Figure 3-6**.

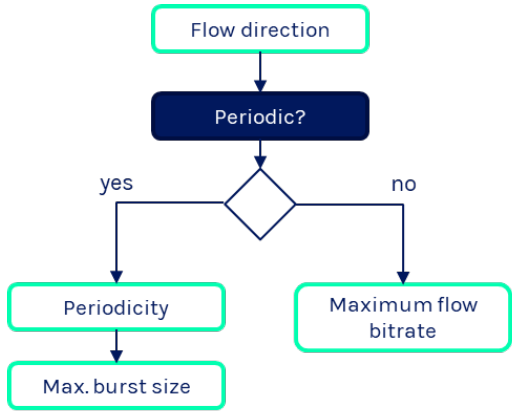

Figure 3-6 *The descriptors of the traffic characteristics.*

The identification of the User-Plane (U-plane) packets and their assignments to traffic flows are executed by matching each of the packets against traffic flow templates (TFTs). A service request may contain multiple TFTs, where each TFT may describe an uplink or downlink packet flow. All packets matching any of the TFTs associated with a service are considered to be part of the service, and the treatment defined in the SLAs are applied to the packets. TFTs are expected to realize the Berkeley Packet Filtering mechanisms.

A service request must contain a service lifetime with a mandatory start time (that can be immediate or some forthcoming date) and an optional termination time. If no termination time is defined, the service may be provided until its termination is not requested.

3.5 Target technological domains

While the PREDICT-6G system is designed to be adaptable to any kind of technological domains, the following concrete technological domains are to be integrated in its first release:

- 3GPP mobile communication domain,
- DetNet/IP domain,
- Wi-Fi network domain.

A domain of these types is handled as a managed entity inside the PREDICT-6G network as depicted in **Figure 3-7**.

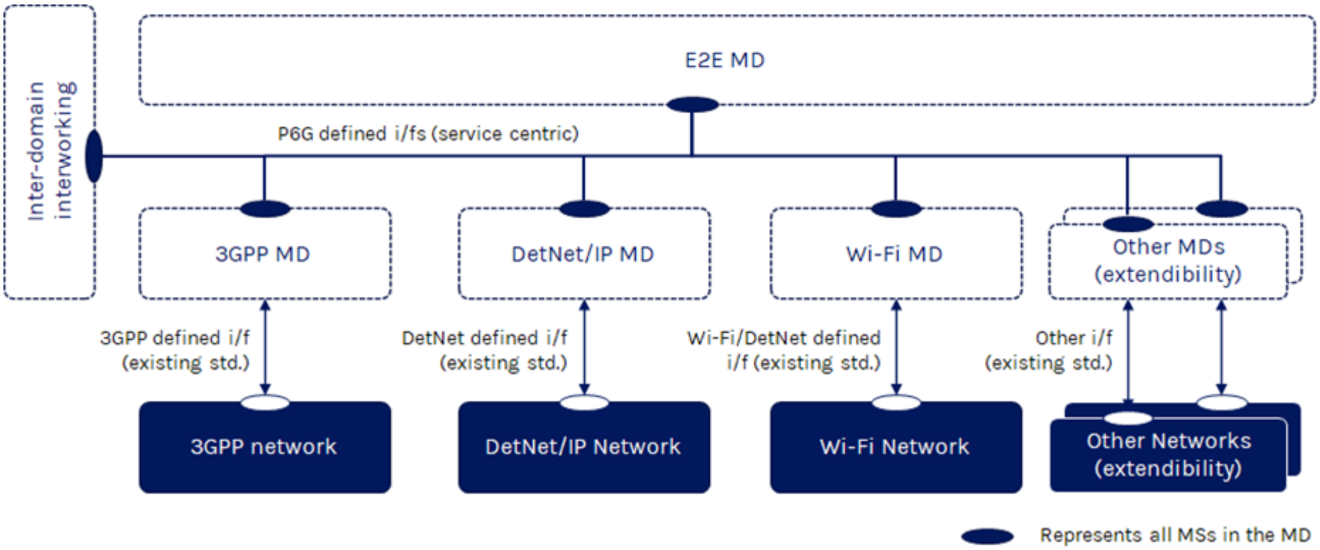

Figure 3-7 Interworking of the technology-specific management domains in the PREDICT-6G network.

4 AI solutions for multi-domain control

4.1 Scope

Given the increasing complexity of communication networks and the applications that they are called to support, the use of AI/ML models has become essential for effective and efficient decision making and the prediction of the system dynamics. In particular, in the case of TSNs it is mandatory to determine an E2E system configuration that guarantees deterministic data delivery, in spite of the heterogeneous network media and technologies as well as the variegate traffic requirements. AI (and ML)-based MD and MS are therefore essential components for realizing the concept of TSNs.

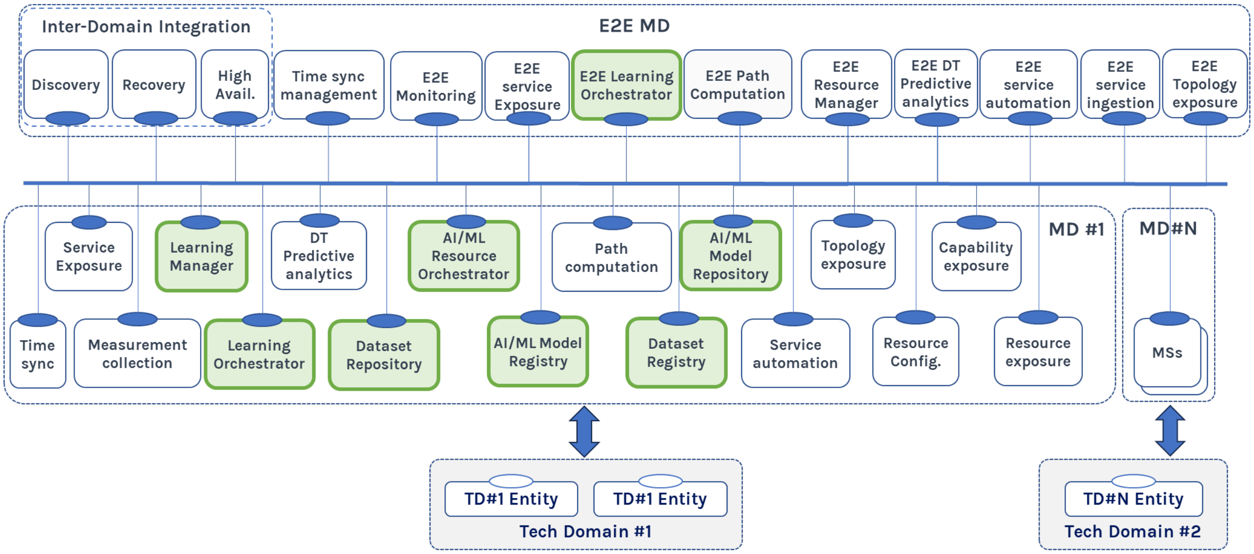

Figure 4-1 *MDs and MSs related to AI-based prediction and decision making (highlighted in green).*

However, it is worth noticing that, as AI/ML operations become pervasive, hence an increasing availability of computational resources is required and a growing amount of data has to be collected, an emerging trend is to use distributed and decentralized approaches that allow for the exploitation of both the computational resources and the data owned by different network nodes. Distributed/decentralized ML poses several challenges, as set forth below:

1. While it enables the exploitation of data and resources at the different nodes, it requires that the adopted AI/ML models, and the structure thereof, are properly tuned to the capability of the available heterogeneous network nodes.
2. The nodes' coordination that has to be ensured for distributed ML operations (i.e., training and inference) comes at a cost of communication bandwidth, additional processing, and, ultimately, energy consumption, at the network/computing nodes.
3. Data collection may become cumbersome, and further increase the consumption of communication bandwidth and energy, especially in dynamic environments and time-varying operational conditions. Furthermore, data ownership/privacy requirements may need to be satisfied, which prevents sharing information among the different entities playing a role in the completion of AI/ML operations.

To address the above challenges, the following MSs are envisioned:

- AI/ML Model Repository, which stores the different types of AI/ML models to be used for a given task.

- AI/ML Model Registry, recording the AI/ML model characteristics, e.g., structure, format of required input and provided output, complexity level in Million Operations Per Second (MOPS), possible dataset that have been used for training (if any), last training/update operation (if any).
- Datasets Repository, storing the datasets to be used for such AI/ML operations as model training.
- Dataset Registry, including the datasets characteristics (e.g., size, domain, input features, privacy requirements, data statistics information).
- Resource Orchestrator for AI/ML Operations (AI/ML Resource Orchestrator for short), an entity that operates locally within each technological domain, orchestrating computational and networking resources within the corresponding domain.
- Learning Manager, interacting with the AI/ML Resource Orchestrator, as well as with the AI/ML Model and Dataset Repositories and Registries, in order to retrieve an AI/ML model/dataset whenever needed and to keep the Repository/Registry up to date.
- Learning Orchestrator, with two types of such MSs being envisioned: an MS that is local to the domain layer (Technological Domain Learning Orchestrator) and an MS that is supervising the overall learning process across different domains (E2E Learning Orchestrator) whenever more than one domain shall be involved. A hierarchical structure is therefore foreseen, where the E2E Learning Orchestrator is responsible for the higher-layer, E2E orchestration and interacts with the Technological Domain Learning Orchestrators.

We remark that the Learning Manager, the Repositories and the Registries are instead logically unique entities, although they can be implemented through distinct physical ones. An example of interactions between the different AI-based MSs is described in section 9.3.

4.2 Management Services

4.2.1 Learning Orchestrator

4.2.1.1 Overview

The Learning Orchestrator has to determine: (i) the AI/ML model (e.g., type of model full version, compressed version), (ii) possibly, the data to be used, and (iii) the nodes to be exploited. It also triggers the (pre-)training or update of AI/ML models whenever needed, based. As mentioned, there exists a local Learning Orchestrator, which acts within a given technological domain, and an E2E Learning Orchestrator that supervises the overall learning process across different domains. Thus, whenever an AI/ML service involves multiple domains, the E2E Learning Orchestrator instructs the local ones, specifying which task should be performed within each domain and, if any, the target performance associated with the requested AI/ML service. The local Learning Orchestrators will then deliver the output of the local AI/ML operation to the E2E Learning Orchestrator, which assembles the received outputs and ultimately provides the AI/ML service. For the sake of brevity, the description below refers to a generic Learning Orchestrator.

4.2.1.2 Exposed Functionalities

The Learning Orchestrator makes use of the AI/ML models and datasets repositories and registries, and the information it receives from the AI/ML Resource Orchestrator about the computing/network nodes. It receives queries for AI/ML services from the E2E Service Automation, makes the decisions listed in Section 4.2.1.1, and then exposes such decisions to the AI/ML Resource Orchestrator. It then gets from the latter the AI/ML operation output (e.g., inference, or (pre-)trained ML model) – operation that has been executed under the control of the AI/ML Resource Orchestrator itself and returns such output (i.e., the requested AI/ML service) to the E2E Service Automation. The set of functionalities provided by the Learning Orchestrator MS is the same at both local and E3E and is reported in **Table 4-1**.

Table 4-1. *List of functionalities provided by the Learning Orchestrator*

Functionalities	Support (M|O)	Description
Get AI/ML model information	M	Provide a mechanism to get an AI/ML model information from the AI/ML Model Registry
Get dataset information	M	Provide a mechanism to get dataset information from the Dataset Registry
Get node information	M	Provide a mechanism to get information about capability of the network/computing nodes from the AI/ML resource Orchestrator
Determine suitable version of AI/ML model, datasets, paradigm, nodes	O	Execute algorithm for identifying the datasets, the (possibly compressed) version of the AI/ML model, the participating nodes, the resources, and the paradigm to be used for the task at hand
Trigger AI/ML Operations	M	Instruct the AI/ML Resource Orchestrator about how the AI/ML operation has to be executed

4.2.1.3 Inputs and Outputs

Inputs (from):

- AI/ML model and datasets registries: Information about AI/ML models and related datasets
- (E2E) Service Automation: AI/ML service (update/train/inference) request
- AI/ML Resource Orchestrator: information on network/computing nodes and their capability

Outputs (to):

- AI/ML Resource Orchestrator: identification of AI/ML model, datasets, AI/ML operation paradigm, nodes to involve and resources to be allocated.
- (E2E) Service Automation: AI/ML service (update/train/inference).

4.2.2 AI/ML Resource Orchestrator

4.2.2.1 Overview

Each AI/ML Resource Orchestrator is in charge of implementing the AI/ML service according to the Learning Orchestrator's instructions. To this end, it has to interact with and instruct the Learning Manager and the network/computing nodes to be involved in the AI/ML operation.

4.2.2.2 Exposed Functionalities

The AI/ML Resource Orchestrator has to interact with (i) the Learning Orchestrator for receiving the instructions on how to implement the AI/ML service (operation to be executed, AI/ML paradigm, target learning time and quality/inference time and quality, nodes to involve, resource to allocate), and to return the output of the related AI/ML operation, (ii) with the Learning Manager to properly operate on the Model/Dataset Repositories and Registries, (iii) with the network/computing nodes to instruct them on how to implement the AI/ML operation execution and get from them the result of such operation. The set of functionalities provided by the Resource Orchestrator for AI/ML Operations MS is reported in **Table 4-2**.

__Table 4-2.__ List of functionalities provided by the AI/ML Resource Orchestrator

Functionalities	Support (M|O)	Description
Get information on network or computing nodes, their	M	Provide a mechanism to collect information from network/computing nodes, their capability, and ability to collect data/owned datasets

capability and the datasets they own/can build		
Get AI/ML models/datasets	M	Provide a mechanism to get AI/ML models/datasets from corresponding Repositories through the Learning Manager
Instruct nodes	M	Provide a mechanism to instruct network/computing nodes about how to perform an AI/ML operation
Allocate resources	M	Instruct selected nodes on resources to be allocated for the AI/ML operation
Return AI/ML operation result	M	Return the AI/ML service to the Learning Orchestrator
Push AI/ML models and related information	M	Provide a mechanism to push AI/ML model and related information to the Learning Manager to be inserted in the AI/ML Model Repository and Register
Push datasets and related information	M	Provide a mechanism to push datasets and related information to the Learning Manager to be inserted in the Dataset Repository and Register

4.2.2.3 Inputs and Outputs

Inputs (from):

- Learning Orchestrator: Trigger and instructions for AI/ML operation
- Network/computing nodes: Information on nodes, their capability and data
- Learning Manager: AI/ML models and Datasets available in Repositories, and related metadata.

Outputs (to):

- Network nodes: Operation to be performed (i.e., training/inference, learning/inference requirements), the paradigm to be used (centralized, type of distributed), nodes to involve, resources to be allocated.

- Learning Orchestrator: AI/ML service (i.e., output of the requested AI/ML operation)
- Learning Manager: AI/ML models and Datasets to be inserted/updated in the Repositories and related metadata to be inserted/updated in the Registries.

4.2.3 Learning Manager

4.2.3.1 Overview

A Learning Manager has to (i) retrieve an AI/ML model/dataset from repository and deliver it to the AI/ML Resource Orchestrator whenever needed; (ii) keep the AI/ML Model and Dataset Repository/Registry updated upon the onboarding of new/updates AI/ML models or datasets.

4.2.3.2 Exposed Functionalities

The Learning Manager gets/pushes AI/ML Models and Datasets from/into the corresponding Repositories and Registries. To this end, beside querying/acting on the Repositories and Registries, it interacts with the AI/ML Resource Orchestrator, which may require/provide such items, as well as with any external user that can provide (already trained/untrained) AI/ML models and/or datasets. The set of functionalities provided by the Learning Manager MS is reported in **Table 4-3**.

Table 4-3. *List of functionalities provided by the Learning Manager*

Functionalities	Support (M|O)	Description
Get AI/ML model and related metadata from external user	M	Provide a mechanism to get an AI/ML model and related information from external user nodes/computing nodes
Get dataset and related metadata	M	Provide a mechanism to get a dataset and related information from external user
Insert/update AI/ML model in Model Repository	O	Provide a mechanism to add/update an AI/ML model (version) – received from the AI/ML Resource Orchestrator or an external user -- to/in the Model Repository
Insert/update AI/ML model record in Model Register	O	Provide a mechanism to add/update an AI/ML model record to/in the Model Register

Insert/update dataset in Dataset Repository	O	Provide a mechanism to add/update a dataset -- received from the AI/ML Resource Orchestrator or an external user -- to/in the Dataset Repository
Insert/update dataset record in Model Register	O	Provide a mechanism to add/update a dataset record to/in Dataset Register

4.2.3.3 Inputs and Outputs

Inputs (from):

- AI/ML Resource Orchestrator: queries for AI/ML Model/Dataset and related metadata
- AI/ML Model/Dataset Repository: AI/ML model/dataset
- AI/ML Model/Dataset Registry: AI/ML model/dataset metadata
- External User: AI/ML Model/Dataset and related metadata
- AI/ML Resource Orchestrator: newly created/updates AI/ML Model/Dataset and related metadata.

Outputs (to):

- AI/ML Resource Orchestrator: Requested AI/ML Model/Dataset and related metadata
- AI/ML Model/Dataset Repository: push model/dataset
- AI/ML Model/Dataset Registry: push model/dataset metadata.

4.2.4 AI/ML Model Registry

4.2.4.1 Overview

The AI/ML Model Registry includes the records describing the ML models already onboarded on the platform through the Learning Manager (i.e., previously loaded by an external user/another entity, pre-loaded onto the platform, or provided by the AI/ML Resource Orchestrator). Each record reports the model identifier and the model metadata, i.e., static and dynamic model information. Static information specifies, in particular, the model size (possibly the compression factor used to obtain this version of the model), structure, the task is apt to, and the suitable format of the dataset/sample to be used for model training/inference. Dynamic information specifies instead when the model has been last trained, with which dataset(s), the achieved quality (or loss value), the model temporal validity.

4.2.4.2 Exposed Functionalities

The Model Registry provides information on the available (trained/untrained) ML models to the Learning Orchestrator and is updated by the Learning Manager. The set of functionalities provided by the Model Registry MS is reported in **Table 4-4**.

Table 4-4. *List of functionalities provided by the AI/ML Model Registry*

Functionalities	Support (M|O)	Description
Keep AI/ML model record	M	Maintain record (identifier and metadata) of ML models
Provide AI/ML model information	M	Provide the record (identifier and metadata) of the requested AI/ML models to the Learning Orchestrator

4.2.4.3 Inputs and Outputs

Inputs (from):

- Learning Manager: ML model metadata upon model onboarding/update
- Learning Orchestrator: query for ML model metadata.

Outputs (to):

- Learning Orchestrator: ML model metadata.

4.2.5 Dataset Registry

4.2.5.1 Overview

The Dataset Registry includes the records describing the datasets that are available on the platform for ML model training or pre-training. The datasets can be onboarded on the platform through the Learning Manager by an external user/another entity, pre-loaded onto the platform, or created/updated by the AI/ML Resource Orchestrator. Each record reports dynamic information, which may change as the dataset is enhanced with new data samples. Such information includes timestamp indicating when the data set has been created/updated, dataset size, sample format, ownership of the data and conditions of use thereof, details on the context in which the dataset has been created (e.g., domain, type of data sources used).

4.2.5.2 Exposed Functionalities

The Dataset Registry provides information on the available datasets to the Learning Orchestrator and is updated by the Learning Manager whenever changes in a dataset occur or the record of a new dataset has to be created. The set of functionalities provided by the Dataset Registry MS is reported in **Table 4-5**.

Table 4-5. *List of functionalities provided by the Dataset Registry*

Functionalities	Support (M|O)	Description
Keep dataset records	M	Maintain record (identifier and metadata) of the available dataset
Provide dataset information	M	Provide the record (identifier and metadata) of the requested dataset(s) to the Learning Orchestrator

4.2.5.3 Inputs and Outputs

Inputs (from):

- Learning Manager: Dataset metadata upon dataset onboarding/update
- Learning Orchestrator: query for dataset metadata.

Outputs (to):

- Learning Orchestrator: dataset metadata.

4.2.6 Model Repository

4.2.6.1 Overview

The Model Repository includes the ML models already onboarded on the platform through the Learning Manager (i.e., previously loaded by an external user/another entity, pre-loaded onto the platform, or provided by the AI/ML Resource Orchestrator), along with the model identifier.

4.2.6.2 Exposed Functionalities

The Model Repository can be queried by the Learning Orchestrator to get AI/ML models or enhanced with new AI/ML models or updated by the Learning Manager. The set of functionalities provided by the Model Repository MS is reported in **Table 4-6**.

Table 4-6. List of functionalities provided by the Model Repository

Functionalities	Support (M|O)	Description
Store AI/ML models	M	Store all available AI/ML models and their identifier.
Provide AI/ML model	M	Provide the requested AI/ML to the Learning Manager

4.2.6.3 Inputs and Outputs

Inputs (from):

- Learning Manager: query for AI/ML models
- Learning Manager: new/updated AI/ML model

Outputs (to):

- Learning Manager: requested AI/ML model.

4.2.7 Dataset Repository

4.2.7.1 Overview

The Dataset Repository stores the datasets already onboarded on the platform through the Learning Manager (i.e., previously loaded by an external user, provided by a network node, pre-loaded onto the platform, or provided by the AI/ML Resource Orchestrator), along with their identifier.

4.2.7.2 Exposed Functionalities

The Dataset Repository can be queried by the Learning Manager to get datasets or enhanced/updated with datasets provided by external users or network nodes through the Learning Manager. The set of functionalities provided by the Dataset Repository MS is reported in **Table 4-7**.

Table 4-7. List of functionalities provided by the Dataset Repository

Functionalities	Support (M|O)	Description
Store datasets	M	Store all available datasets and their identifier.
Provide dataset	M	Provide the requested dataset to the Learning Manager

Funded by the European Union

4.2.7.3 Inputs and Outputs

Inputs (from):

- Learning Manager: query for dataset
- Learning Manager: new/updated dataset

Outputs (to):

- Learning Manager: requested dataset(s).

5 Digital Twinning as predictability enabler

5.1 Scope

In PREDICT-6G the DT technology is used to provide two main functionalities to support management services related with predictive analytics. On the one hand, the DT predicts the QoS KPIs of a new service request on a candidate path on a single technological domain, as well as predicts the impact on the QoS KPIs to already provisioned services. On the other hand, the DT predicts the E2E QoS KPIs of a new service request crossing several technological domains. In both cases, the main challenges are related to the computation of partial KPIs on a service supported by differentiated technological domains. **Figure 5-1** shows the MSs related to the digital twinning in the AICP.

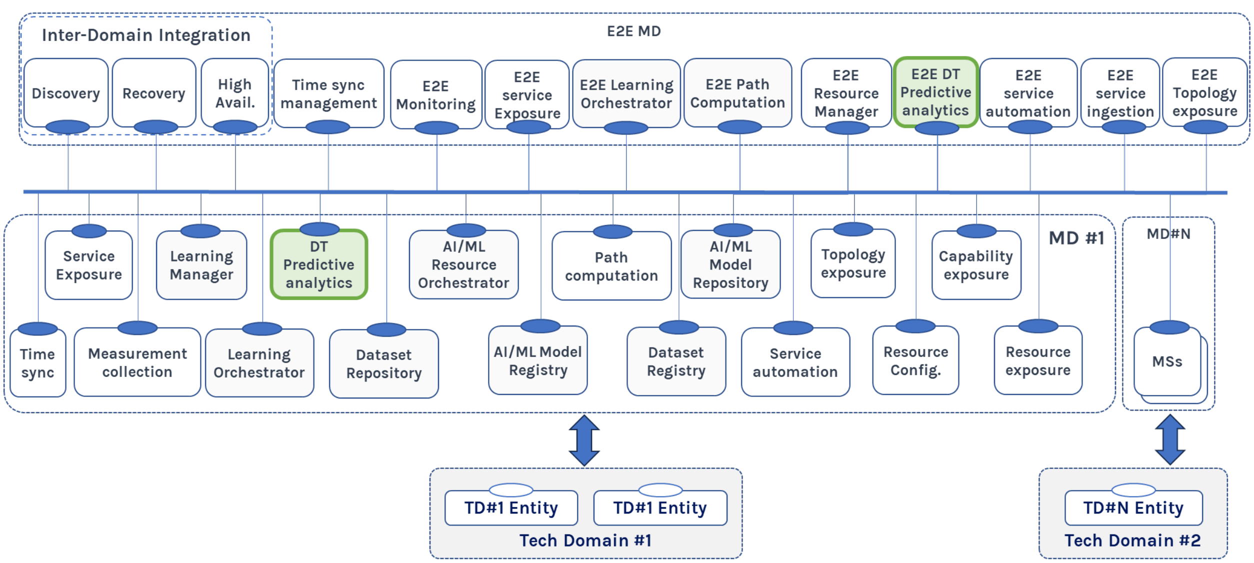

Figure 5-1 MDs and MSs related to DT-based predictive analytics (highlighted in green).

5.2 Management Services

5.2.1 DT Predictive Analytics

5.2.1.1 Overview

This service consists in predicting the KPIs of new and already established traffic flows (TSN, Best effort, etc) on a specific technological domain.

5.2.1.2 Exposed Functionalities

Two main functionalities are identified: first, the DT receives a request (*r*) to compute the KPIs of a traffic flow if it was established on a candidate route given the traffic characteristics that will be supporting. Second, the DT computes the KPIs for the request and the incremental KPIs for the already established traffic flows (*F*).

The following list of KPIs can be measured and quantified by the DT for the flow request under evaluation and the rest of established flows:

- **Throughput** (or data rate): Average volume of traffic (in Mb/s or Gb/s) at the input and/or output of a route (modelled as a queuing system) at a given time.
- **Traffic loss ratio**: Average percentage of traffic that is rejected due to lack of available capacity at a given time.
- **Delay** (or latency): Average of the elapsed time (in ms) that the traffic experiences when it traverses the defined route (modelled as a queuing system) from input to output at a given time.
- **Jitter**: Standard deviation of the elapsed time (in ms) that the traffic experiences when it traverses the defined route (modelled as a queuing system) at a given time.

For each of the described KPIs, two type of measurements can be obtained:

- **Nominal values** i.e., the actual values that the DT estimates for request *r* if it would be established.
- **Delta values** i.e., the difference (increment or decrement) that the DT estimates comparing the KPIs computed before and after establishing request *r*.

Typically, nominal values can be computed for the request *r*, while delta values are computed for the existing set of flows *F*. However, depending on the specific KPI, service type and use case, the set of available KPI metrics and measurements can change. In any case, the DT is always able to provide relevant and accurate QoS estimation data that can be used for decision making.

Table 5-1. *List of functionalities provided by E2E DT Predictive Analytics*

Functionalities	Support (M|O|C)	Description
request to compute the KPIs of a traffic flow	M	**Inputs (from MS/Interfaces)** **Network (*G*)** ◆ *N:* Set of nodes, each with the required parameters e.g., switching/ routing capacity. ◆ *E:* Set of currently active edges (adjacencies) between nodes in *N,* each with the required parameters e.g., distance, maximum capacity ◆ *IF:* Set of network interfaces *IF*, each with the required parameters e.g., edge assignment, speed, time slicing policy **Established TSN and BE flows (*F*) (for each flow *f* ∈ *F*)** ◆ *Type:* Type of service/traffic ◆ *Route:* Current route, defined as a sequence of network interface ◆ *maxTraffic:* Maximum actual traffic ◆ *KPIif*: List of KPI metrics per interface of the flow (if available) **New flow request (*r*)** ◆ *Type:* Type of service/traffic ◆ *Route:* Candidate route to evaluate, defined as a sequence of network interfaces ◆ *maxTraffic:* Maximum expected traffic **Outputs (to MS/Interfaces)** **Request (r)** ◆ KPInom: Expected KPI **Flows (F) (for each flow f ∈ F)** ◆ KPIdelta: List of expected KPI metrics per interface if request r were established

5.2.1.3 Inputs and Outputs

Inputs (from):

- Service Automation: Service DB, and Service provisioning and decommission notifications.
- Topology Exposure: Topology DB and Topology update notifications.
- Measurement Collection: Traffic and KPI Measurements.
- Service Automation: Request to compute the E2E KPIs
- Resource and Capability Exposure: resource characteristics

Outputs (to):

- Service Automation: E2E KPI prediction.

5.2.2 E2E DT Predictive Analytics

5.2.2.1 Overview

This service consists in predicting the E2E KPIs of new traffic flows (TSN, BE, etc)

5.2.2.2 Exposed Functionalities

Two main functionalities are identified: first, the DT receives a request (*r*) to compute the KPIs of a traffic flow if it was established on a candidate route traversing several technological domains, together with the partial KPIs for each of these domains. Second, it computes the KPIs identified and described in Section 5.2.1.2.

Table 5-2. *List of functionalities provided by DT Predictive Analytics*

Functionalities	Support (M|O|C)	Description
request to compute the E2E KPIs of a traffic flow	M	**Inputs (from MS/Interfaces)** **New flow request (*r*)** ◆ *Type:* Type of service/traffic ◆ *Route:* Candidate route to evaluate, defined as a sequence of technological domains ◆ *maxTraffic:* Maximum expected traffic ◆ KPInom: List of expected KPI for each technological domain

		Outputs (to MS/Interfaces) **Request (r)** ◆ KPInom: Expected E2E KPI

5.2.2.3 Inputs and Outputs

Inputs (from):

- Service Automation: Service DB, and Service provisioning and decommission notifications.
- Topology Exposure: Topology DB and Topology update notifications.
- Measurement Collection: Traffic and KPI Measurements.
- Service Automation: Request to compute the E2E KPIs

Outputs (to):

- Service Automation: KPI prediction.

6 Data collection and management

6.1 Scope

The collection of data from the different domains of the Multi-technology Data Plane is crucial to enable both predictability and reliability aspects in PREDICT-6G services. The continuous measurement of data and KPIs provides a timely information that can be exploited by analytic processes (based on both AI/ML and DT) to provide predictions on the future behaviour of the network in terms of determinism, e.g., assessing if the creation of a new service, given the current monitoring values, may affect the performance KPIs of the already running services. KPI variation can also provide important information on the status of the provisioned services and, in case of abnormal deviation, a decision process (e.g., an AI) may decide to enforce specific configurations on the underlying network(s) to preserve services' determinism. **Figure 6-1** is shows the MSs enabling the management and the collection of the data, at both technological and E2E domains.

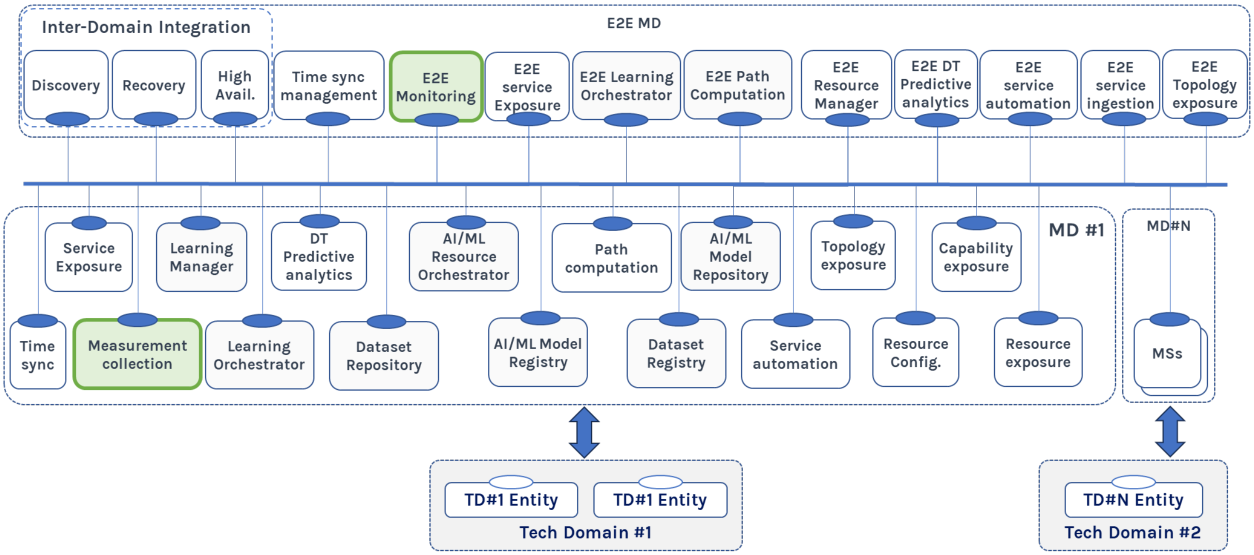

Figure 6-1 MDs and MSs related to data collection and management (highlighted in green).

The data collected at the local domains can be exploited locally and/or used by the E2E Domain for the automatic management of the E2E services. The information collected is technology specific, i.e., it depends on the target technological domain so that, to be used into an E2E service context, it needs to be somehow normalized and abstracted towards common data models (see Sections 8.2). In addition, in order to avoid a complete open access, where any actor (user, application, MS) can access any data from any domain, a proper data management is required, to provide mechanisms for Authentication, Authorization and Access Control. The data can be consumed as soon as collected (real-time data exposure) or stored in form of timeseries, to be used in the future, e.g., for AI/ML training. The storages to maintain such timeseries are not explicitly indicated in the AICP architecture as the MSs refers on the functionalities to manage and collect the data. MSs exploit dedicated storage, although it cannot be excluded, in future refinements of the AICP architectural design, the interaction with external ones, e.g., the Dataset Repository (see Section 4.2.7) for the automatic creation of datasets suitable to train AI/ML models.

6.2 Management Services

6.2.1 Measurement Collection

6.2.1.1 Overview

The Measurement Collection is a service local to the technological domains in charge of collecting measurements of interest for the management of deterministic services, with a certain degree of granularity (e.g., per-service, per-flow, per-tenant). The collected data will be accessed through a specific service interface that can be consumed by other MSs (e.g., E2E Monitoring described below), human users, local applications, etc.

6.2.1.2 Exposed Functionalities

The data collected by different data sources are at the same time stored and exposed towards other entities to be consumed. Optionally, the data can be manipulated to provide statistical metrics, e.g., average, standard deviation, and any other forms of aggregation. The set of functionalities provided by the Measurement Collection MS is reported in **Table 6-1**.

Table 6-1. *List of functionalities provided by Measurement Collection*

Functionalities	Support (M|O|C)	Description
Historical data management	M	Storage and the retention of historical data (e.g., timeseries) collected from the Technological Domain. The data are useful to compile statistics, to train AI/ML models and implement Digital Twins. The data can be also accessed by MSs at the E2E Management Domain (e.g., E2E Monitoring) for analysis at the E2E level. In this case, the data are represented in a uniform manner for any collection domain, as discussed in Section 8.2.1
Real-time data management	M	Real-time exposure of data collected from the different data sources. Useful to trigger event-driven processes, such as control loops, enabling network automation, etc. As per Historical Data, the real-time data can be also collected by the E2E Monitoring and the same consideration on data uniformity are valid.
Data Manipulation	O	Allows service consumers to compute data aggregation statistical functions such as Average, Standard Deviation, Variance, etc.

Service Configuration	M	Configuration of data sources and level of data collection granularity for a given service, i.e., sub-service belonging to an E2E deterministic service. Authorization, authentication, and access control functionalities, to enable differentiated data access (e.g., per user, tenant, application) at different granularity (per-service(s), flow(s)). Mainly directed to System/Network administrators.

6.2.1.3 Inputs and Outputs

Inputs (from):

- Technology domains: Obtain data and relevant information for modelling the performance of the network.
- E2E monitoring: Dynamic configuration of data sources and level of data collection granularity for a given service.

Outputs (to):

- Learning Orchestrator, DT Predictive service: Real-time and historical data for local prediction and decision processes.
- E2E Monitoring: Real-time and historical data related to the local sub-service.

6.2.2 E2E Monitoring

6.2.2.1 Overview

This MS represents the E2E version of the Measurement Collection discussed above. This implies that most of the functionalities exposed are in common between the two MSs, although some differences exist.

The E2E Monitoring (E2EM) is part of the E2E Management Function (in the E2E Management Domain) and has access only to other MSs in both E2E and Local MDs. Therefore, the E2EM cannot collect data directly from the Data Plane and its only data sources are Measurement Collection MSs at the Technological Domains. In this regard, the E2EM requires that the data provided by the different domains are somehow normalized and abstracted in a common format (this task is in charge of the Measurement Collection MS), in line with the idea of abstraction provided by the E2E domain. The usage of a common format makes the data collected to be meaningful at the E2E level (technological-dependent formats are, most probably, meaningful only in the domain of origin) while, the translation offloading to the lower layers, allows avoiding wasting of computational resources at the E2E Management Function and enables the scalability.

6.2.2.2 Exposed Functionalities

As already introduced, the functionalities provided by the E2E Monitoring are basically the same provided by the Measurement Collection, declined in an E2E manner as reflected in the descriptions reported in **Table 6-2**.

Table 6-2. *List of functionalities provided by E2E Monitoring*

Functionalities	Support (M|O|C)	Description
Historical data management	M	Storage and the retention of historical data (e.g., timeseries) collected in the Technological Domain. The data are useful to compile statistics and to train E2E AI/ML models and implement E2E Digital Twins. As described above, the storage can be dedicated or belonging to other MSs.
Real-time data management	M	Real-time exposure of data collected by the different data source. Useful to trigger event-driven processes, such as control loops, enabling E2E network and E2E service management automation.
Data Manipulation	O	Allows service consumers to compute data aggregation functions such Average, Standard Deviation, Variance, etc.
Service Configuration	M	Authorization, authentication, and access control functionalities, to enable differentiated data access (e.g., per user, tenant, application) at different granularity (per-service(s), flow(s)). Mainly directed to System/Network administrators. The policies and permissions set at the E2E level need to be reflected also the Local Management Domains. Indeed, it is important to note that the data consumed by e.g., a tenant at the E2E domain, are collected from the underlying technological domains, accessing the respective Measurement Collection MSs with proper permissions.

6.2.2.3 Inputs and Outputs

Inputs (from):

- Measurement Collection: Real-time and historical data related to the local sub-services. Used for the automatic control of the E2E deterministic services.
- E2E Service Automation: Configuration of the monitoring process at E2E service orchestration. This also implies run-time adjustments e.g., at E2E service update, and termination of monitoring process at service termination.

Outputs (to):

- E2E Learning Orchestrator, E2E DT Predictive service: Real-time and historical data for local prediction and decision processes.
- Measurement collection: Dynamic configuration of data sources and level of data collection granularity for a given service.

7 Inter-domain orchestration and network control

7.1 Scope

As explained in Section 3, to enable the automated creation of E2E deterministic services, the PREDICT-6G AICP architecture relies on a hierarchical approach composed of an E2E Management Framework and several technology specific Management Frameworks for controlling different technology domains (3GPP, Wi-Fi, IP, etc). Through this design, PREDICT-6G aims at complementing the capability of specific technology controllers and creating enablers for managing E2E determinism over multiple technologies.

Nevertheless, to ensure dynamic network control and orchestration capabilities as part of the AICP, it is crucial to address frictionless inter-domain orchestration when integrating heterogeneous 6G deterministic networks. In this section, the inter-domain orchestration and network control framework is described by analysing a set of MSs, depicted in **Figure 7-1** which have been designed for enabling specific technology domains to be orchestrated as part of an E2E deterministic service.

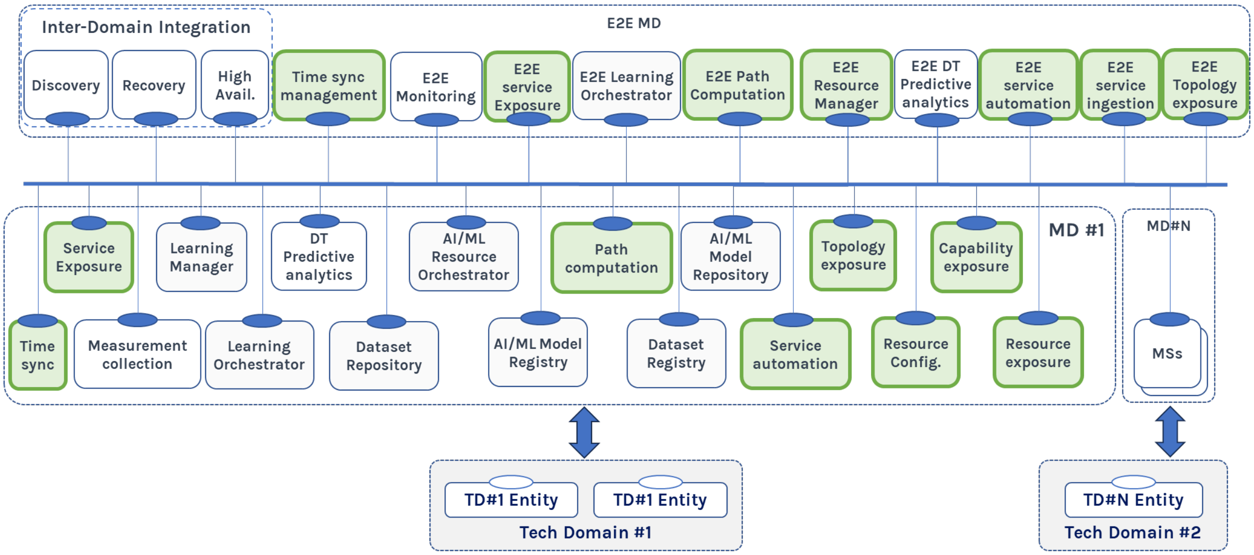

Figure 7-1 MDs and MSs related to Inter-domain orchestration (highlighted in green).

This set of MSs in technology specific and E2E domains are designed to: (i) receive and interpret deterministic E2E service intents from the users of the system; (ii) perform domain selection based on E2E and per domain topology and resource status; (iii) translate E2E requirements to selected domain specific requirements; (iv) perform continuous control loop automation; (v) expose data-plane topology, deterministic capabilities and resources, and (vi) configure network synchronization.

These MSs are directly related to the services detailed in Sections 4, 5 and 6, and their combination allows to add predictability and monitoring functionalities as part of the inter-domain orchestration and network control. In the following section, MSs are described in detail for both technology and E2E domains.

7.2 Management Services

7.2.1 Technology Domain

7.2.1.1 Time Sync

7.2.1.1.1 Overview

The Time Sync service learns time sync capabilities of the technology domain (e.g., supported sync protocols, whether the domain can sync to external Grand Master clock, etc.) and continuously monitors and tracks the status of the time sync in the domain. It reports changes in the time sync status (e.g., synced/not synced, change in Grand Master clock, etc.) to the Time Sync Management service of the E2E domain. Based on the required time sync service attributes received from the Time Sync Management service, it configures the time sync in the technology domain and provides feedback regarding the successfulness of the configuration request.

7.2.1.1.2 Exposed Functionalities

The Time Sync service exposes information to the Time Sync Management service regarding the time sync capabilities and actual time sync status of the technology domain. The set of exposed information describes whether the technology domain can sync to Grand Master clock of other technology domains or can act as Grand Master for other technology domains. It also provides information whether the technology domain is in synced or not synced state and which time domain it uses.

Table 7-1. List of functionalities provided by Time Sync

Functionalities	Support (M|O|C)	Description
Clock Master	O	Whether the management service may configure its managed entity to expose its own clock to other MEs as clock master.
Clock Slave	O	Whether the management service may configure its managed entity to sync its own clock as clock slave to an external clock master.
Time Sync Status	M	Either “synced” or “not synced”. If the status is “not synced”, any timestamped data produced from the MD should be marked accordingly to notify the consumers of the MSs. Lack of time synchronization may also mean reduced capability of the ME to provide or fulfil deterministic services.
Time Domain	M	The identity of the (g)PTP time domain to which the ME synchronizes (if the time sync status is “not synced”, indicate the time domain to which synchronization is or would be attempted).

7.2.1.1.3 Inputs and Outputs

Time Sync service interacts with the Technology Domain via the existing domain specific management interfaces and the Time Sync Management service of the E2E domain via the PREDICT-6G defined APIs.

Inputs (from):

- Technology Domain: time sync information regarding the actual time sync settings and configuration, status of the time sync service in the technology domain.
- Time Sync Management service: required time sync capabilities in the technology domain, time sync configuration.

Outputs (to):

- Technology Domain: time sync status query and time sync configuration commands.
- Time Sync Management service: technology domain time sync status, feedback about time sync configuration request.

7.2.1.2 Service Exposure

7.2.1.2.1 Overview

The Service Exposure exposes towards consumers (e.g., other MSs, applications, users, etc.) information concerning the current status of the services provisioned at a given Technological Domain. The information exposed are mapped by following an information model common to all the Technological domains so that the correspondent E2E services e.g., E2E Service Management (See 7.2.2.7) can collect them easily and give a representation of the related E2E Service. The possibility to retrieve service information in the technology-dependant format can be optionally supported for the purpose of local MSs (e.g., the DT of local network could require such an information)

7.2.1.2.2 Exposed Functionalities

The Service Exposure is a “read-only” MS which manly acts as service information provider and translator, from the local (tech-dependant) service format to the common one.

Table 7-2. List of functionalities provided by Service Exposure

Functionalities	Support (M|O|C)	Description
Obtain Service Information	M	Obtain information related to service conditions and general information of a service running in a specific technology domain such as service type, resources consumed, owner of the service, starting time, etc.
Abstract Service Information	M	Abstract Service information and map this data to a common data model
Exposes Service Information	M	Expose abstracted service information to other MS through a dedicated MS interface. Tech-domain information format can be optionally supported

7.2.1.2.3 Inputs and Outputs

Inputs (from):

- Technology Domain: service information collected from the technology domain, assuming that such domain would provide an interface to retrieve it. Alternatively, service information can be collected in a passive manner from the service provisioner i.e., the Service Automation MS, in charge of provisioning deterministic service in local domains.

Outputs (to):

- E2E Service Exposure: provides the E2E exposer with information related to the composing sub-service.
- Learning Orchestrator, DT-based Predictive Analytics: Local predictive and decision services requires to be synchronized with the current status of deployed deterministic service.

7.2.1.3 Service Automation

7.2.1.3.1 Overview

Service Automation is a MS hosted in each technology domain for ensuring the closed-loop automation of a service after the acceptance of a specific service request performed at the E2E MD.

The closed-loop automation process of a specific service consists of three different stages: (i) service provisioning, where an initial configuration is defined and set in the specific technology domain to meet the requirements of the service request; (ii) service assurance, where a continuous control-loop configuration is set to ensure that the agreed SLA of the service is maintained in any conditions; and (iii) service termination, where the delivery of the service is terminated due to a specific request, which could be made by the user/operator, or due to the expiration of the service lifetime.

7.2.1.3.2 Exposed Functionalities

The Service Automation MS plays a crucial role in AICP for each technology domain since it configures, ensures the SLAs, and terminates services while considering internal and external flows of information. The set of functionalities required for this MS are:

Table 7-3. *List of functionalities provided by Service Automation*

Functionalities	Support (M|O|C)	Description
Local path request ingestion	M	Ingest and forward the local path computation request performed by the E2E Path Computation towards the Path Computation MSs.
Service Provisioning	M	Translate the service request information for setting an initial configuration in the specific technology domain to fulfil the requirements of the service. This process includes allocating path resources, updating resource availability and notifying service provisioning.
Service Assurance	M	Assure a continuous control-loop self-configuration to maintain the SLA of the service once it has been configured. Control-loop configuration requires a

		continuous exchange of information with other MS such as Measurement Collection.
Service Termination	M	Finish the service instantiation under an explicit request or due to the expiration of the service lifetime. Service Termination requires the exchange of information with E2E Service Automation.
Local Conflict Prediction/Resolution	C	Detect, predict, and solve conflicts in the specific technology domain related to the nature of the service. In case conflicts cannot be solved, the MS will have to be able to escalate them to the E2E Service Automation MS.

7.2.1.3.3 Inputs and Outputs

To expose all the aforementioned functionalities, the Service Automation MS will need to interact with a wide set of MS. The set of interactions in the shape of inputs/outputs are:

Inputs (from)

- E2E Service Automation: Service Automation could receive service provisioning and service termination requests from the E2E Service Automation during E2E service provisioning stage in case any conflict occurs or in case the user/operator decides to terminate the service instance.
- E2E Path Computation: Service Automation receives the path computation request from the E2E path computation in order to trigger the local path computation process.
- Path Computation: Service Automation receives the local path selection result performed by the Path Computation.
- DT Predictive Analytics: Service Automation receives the KPI prediction performed by this MS.
- Measurement Collection: Service Automation needs to get information related to service performance (KPIs) for ensuring the fulfilment of the SLAs for each specific service. Service performance is provided at the technology domain by the Measurement Collection MS.
- Resource Configuration: Resource configuration confirms the resource allocation for the specific local path.

Outputs (to)

- Resource Configuration: Service Automation defines the initial service configuration based on the service request. Service configuration needs to be exposed to the Resource Configuration MS for configuring resources in each technology domain to fulfil the service needs.
- E2E Path Computation: Service Automation forwards the local path selection performed in the technology domain.
- E2E Service Automation: Service Automation exposes closed-loop information in each technology domain towards the E2E Service Automation Service for performing service assurance and termination, and in case of the need for escalating local conflicts.
- DT-based Predictive Analytics: Service Automation notifies the service provisioning after allocating the required resources and for the specific sub-service in the local domain. Additionally, requests the computation of the KPIs related to that service for the selected path.

7.2.1.4 Path Computation

7.2.1.4.1 Overview

The Path Computation MS is responsible for calculating routes over its managed specific technology domain. To do this, the Path Computation MS uses the information related to the topology, the data plane resources availability as well as their deterministic capabilities, which is provided by the Topology, Resources and Capability exposure MSs of the technological domain MF. In the framework of PREDICT-6G, the Path Computation MS is invoked by the Service Automation MS during the process of the E2E Service computation and configuration. Once the E2E Service Automation triggers the Service Provisioning process, the E2E Path Computation requests a local path computation to the Service Automation MS for each specific technological domain. Then, the Service Automation forwards the path computation request to the specific Path Computation MS. Upon the reception of such request, the Path Computation MS uses a set of path computation algorithms that are fed with the topology, resource availability and capabilities of the underlying data plane to calculate a set of candidate paths that are likely to fulfil the requirements posed by the requested service. To increase the chances of success on the candidate path computation, the Path Computation MS relies on the AI/ML-based MSs, which implement a set of AI/ML techniques to predict the feasibility or optimality of the computed candidate path. The result will be sent to the Service Automation MS, which will contact the DT Predictive Analysis MS to compute the KPIs associated to such candidate and will decide if they are acceptable. If so, the path will be sent to the E2E domain for final E2E evaluation. Otherwise, another candidate will be computed and analysed.

7.2.1.4.2 Exposed Functionalities

Table 7-4 enumerates and describes the functionalities and communication interfaces that must be implemented in the Path Computation MS.

Table 7-4. List of functionalities provided by Path Computation

Functionalities	Support (M|O|C)	Description
Path Computation Request	M	Receive request from Service Automation MS for performing local path computation
Topology, capability and resource request	M	Request information related to topology, capability and available resources for a specific technology domain
Path Computation execution	M	Perform an initial path computation by leveraging capability, resource and topology information
Path Computation result communication	M	Return path calculation for each of the potential paths

7.2.1.4.3 Inputs and Outputs

<u>Inputs (from)</u>

- <u>Resource Exposure:</u> the service will require APIs to collect the resources available in the technological domain.
- <u>Topology Exposure:</u> the interconnection of the resources at the data plane level is needed to compute the path.
- <u>Capability Exposure:</u> to properly determine what resources can be used for the path to fulfil the requirements of the requested service.
- <u>Service Automation:</u> this MS is the one that sends the path computation request to the Path Computation MS.
- <u>Learning Orchestrator:</u> The feasibility of the computed candidate path requested to this MS prior to KPI evaluation and further configuration.

Outputs (to):

- Service Automation: The Path Computation confirms the selected path to the Service Automation MS.
- Learning Orchestrator: with information related to the computed path for feasibility.
- Resource Exposure: the service requests the resources available in the technological domain.
- Topology Exposure: the interconnection of the resources at the data plane level is requested to compute the path.
- Capability Exposure: the service performs this request to properly determine what resources can be used for the path to fulfil the requirements of the requested service.

7.2.1.5 Resource Configuration

7.2.1.5.1 Overview

This MS translates the E2E service requirements into per-domain resource configurations to facilitate the ingested service.

7.2.1.5.2 Exposed Functionalities

Table 7-5. List of functionalities provided by Resource Configuration

Functionalities	Support (M|O|C)	Description
Configuration status response	C	In response to a configuration request from the E2E management domain, the MS responds with a measure of capability of meeting the KPI
Resource usage monitoring	C	While rendering the ingested service, the MS sends status reports of varying utilization metrics of the resources

7.2.1.5.3 Inputs and Outputs

Inputs (from)

- Resource Exposure: the service will require APIs to adjust the amenable parameters of the resources available in a management domain.

- Topology Exposure: to leverage or curb potential unwanted side effects of a configuration the MS needs to know how the resources are interconnected within a management domain.
- Capability Exposure: to properly determine the suitability of a resource to handle an ingested service, the MS will require information on their capabilities.
- E2E Resource Management: the MS will receive configuration requests from the E2E resource manager.

Outputs (to):

- Technology Domains (MDP) for configuring resources in each domain.

7.2.1.6 Topology Exposure

7.2.1.6.1 Overview

The Topology Exposure service is used to expose topology information to other MS in each technology domain so that the topological view of the relationship among nodes and links in a certain domain can be retrieved.

Depending on the nature of the network elements in the domain there could be already defined mechanisms that could facilitate the automatic retrieval of such information. For instance, for layer-3 network elements (e.g., IP nodes) it is possible to leverage on BGP-LS (Border Gateway Protocol with Link State) for collecting such topological relationships, including some other valuable information like performance metrics, for instance. In the case of layer-2 devices (e.g., Ethernet switches) other protocols can be leveraged, e.g., LLDP (Link Layer Discovery Protocol), to obtain the neighbourhood relation among nodes and built from that the overall topology.

In some other cases, there are no standard mechanisms for the automatic retrieval of the topology of the elements in the domain. In those cases, the view of the domain could be retrieved from complementary management components in place, such as inventory or specific-purpose management or control elements (e.g., SW Defined Networking (SDN) controller, Management System, etc) providing the required topological view.

In order to deliver a complete topology, it will be then necessary to aggregate the information available through different sources and compose an integrated topological view. To do that, it is necessary to define some mechanism that could permit such aggregation so that the correspondence of the information between the boundaries of each domain can be easily related.

7.2.1.6.2 Exposed Functionalities

The Topology Exposure system provides a number of functionalities that can be consumed by some other systems in the architecture. Basically, on the one hand=, the system collects the information from the devices and/or complementary per-domain control and management systems to get knowledge of the topological relationships between network elements. This could include additional information than the links among them, like performance metrics on such links. On the other hand, the Topology Exposure system exposes the retrieved information.

Complementary to that information integration, a process of abstraction could be put in place either for concentrating on relevant information for the purpose of the service request towards the Topology Exposure system, or simply for privacy considerations.

Table 7-6. *List of functionalities provided by Topology Exposure*

Functionalities	Support (M|O)	Description
Retrieve Topology Information	M	Obtain topology information between service endpoints (e.g., available connectivity 1:1, 1:N, N:N) through the topology API developed in the MDP for each technology domain
Abstract Topology Information	M	Abstract topology information and map this data to a common data model
Expose Topology Information	M	Expose abstracted topology information to other MS through a dedicated MS interface

7.2.1.6.3 Inputs and Outputs

Inputs (from):

- Topology-related protocols: in some situations, standard-based protocols could serve for the purpose of retrieving topological information. This is the case of BGP-LS or LLDP.
- Topology API (MDP): alternatively, the topological information in each domain could be retrieved from complementary control and management components in the domain, such as Inventory, SDN Controller or Element Management Systems.
- Path Computation: Accepts topology information request.

Outputs (to):

- Path Computation: Path Computation MS can consume topological information for identifying the proper paths in the network.
- E2E Topology Exposure: the information provided by the Topology Exposure MS will serve as basis for building the overall end to end topological view of the network.
- E2E Resource Manager: the topological information will be used to identify potential issues at the time of configuring resources.

7.2.1.7 Capability Exposure

7.2.1.7.1 Overview

This service is aimed to expose the information related to the deterministic capabilities of the technology domain. Such capabilities may be exposed either to other MSs of the technological domain (e.g., the Path Computation MS), or MSs of upper level MFs such as the E2E Resource Manager (Section 7.2.2.5). To do this, the Capability Exposure MS will rely on the APIs provided by the different technological domains of the MDP of PREDICT-6G (PREDICT-6G/D2.1/3, 2023).

7.2.1.7.2 Exposed Functionalities

Table 7-7 provides a description of the main functionalities that must be implemented in the Capability Exposure MS.

Table 7-7. List of functionalities provided by Capability Exposure

Functionalities	Support (M|O|C)	Description
Obtain deterministic capabilities	M	Obtain deterministic capabilities between service endpoints (PREOF, det. scheduler, multi-path, etc.) through the capability API developed in the MDP for each technology domain.
Abstract deterministic capabilities	M	Abstract deterministic capabilities information and map this data to a common data model

Expose deterministic capabilities	M	Expose abstracted deterministic capability information to other MS through a dedicated MS interface

7.2.1.7.3 Inputs and Outputs

Inputs (from):

- Technology Domains (MDP): This MS will use the interfaces exposed by the data plane of the domain to collect information about its deterministic capabilities.
- Path Computation: Accepts capability information request.

Outputs (to):

- Path Computation: It will use the capabilities information to compute suitable paths that will be further assessed by the DT.
- E2E Resource Manager: It will keep abstracted information about the deterministic capabilities available in the technological domain.

7.2.1.8 Resource Exposure

7.2.1.8.1 Overview

The Resource Exposure provides the current status of available resources in a technological domain. As per other exposure MSs, the consumer of such information can be other MSs, Application, Users, etc, and the information format follows a common data model.

7.2.1.8.2 Exposed Functionalities

Table 7-8. List of functionalities provided by Resource Exposure

Functionalities	Support (M|O|C)	Description
Obtain available resources information	M	Obtain information related to available resources between service endpoints (delay budget, loss, bandwidth, load, multi-path) through the resource API developed in the MDP for each technology domain
Abstract resource information	M	Abstract resource information and map this data to a common data model

Expose abstracted resource information	M	Expose abstracted resource information to other MS through a dedicated MS interface

7.2.1.8.3 Inputs and Outputs

Inputs (from)

- Technology Domains (MDP): information related to available resources in each domain.
- Path Computation: Accepts resource availability information request.

Outputs (to)

- E2E Resource Exposure: Provide status of local resources consumed/available.
- Learning Orchestrator, DT-based Predictive Analytics: Local predictive and decision services requires to be synchronized with the current status of available resources.

7.2.2 E2E Domain

7.2.2.1 Time Sync Management

7.2.2.1.1 Overview

The Time Sync MS is responsible for setting up and maintaining E2E time synchronization. It provides information regarding the capabilities and characteristics of the PREDICT-6G system's time sync to other E2E services. E2E services may also request changes or signal preferences (e.g., which technology domain to act as Grand Master clock) to Time Sync Management service. It continuously interacts with the Time Sync service in the technology management domains to learn about time sync capabilities and actual time sync status in the respective domains. Based on the learned information, it orchestrates the time sync service in the PREDICT-6G system via sending configuration to the Time Sync service of the technology domains.

7.2.2.1.2 Exposed Functionalities

The Time Sync MS ingests time sync information from the Time Sync service of the technology domains. It also ensures a common time synchronization across all technology domains.

Table 7-9. List of functionalities provided by E2E Time Sync Management

Functionalities	Support (M|O|C)	Description
Ingest time synchronization from domains	M	Ingest time synchronization information from the Time Sync MS of each technology domain
Ensure common time synchronization	M	Ensure a common time synchronization across all technology domains

7.2.2.1.3 Inputs and Outputs

The Time Sync MS interacts with the Time Sync service of technology specific management domains and the E2E services via the PREDICT-6G defined APIs.

Inputs (from):

- Time Sync service of the technology specific management domain: time sync information regarding the actual time sync settings and configuration, status of the time sync service in the technology domain, configuration command feedback
- E2E services: query regarding time sync capabilities, request time sync preferences

Outputs (to):

- Time Sync service of the technology specific management domain: time sync status query and time sync configuration commands.
- E2E services: PREDICT-6G system's time sync capabilities and characteristics.

7.2.2.2 E2E Service Ingestion

7.2.2.2.1 Overview

The E2E service ingestions (E2E-SI) manages the requests of E2E services, performed by human users (e.g., Telco Operator's staff) and applications (e.g., industrial controller). It is important to highlight that the E2E service ingestion is a complex MS, representing the entry point of the AICP. In this MS, a service request needs to be parsed and validated and then, in case of success, remapped to specific information models representing the E2E service and, at the same time decomposed in sub-services to be provisioned over the targeted technological domains.

7.2.2.2.2 Exposed Functionalities

Table 7-10. List of functionalities provided by Service Ingestion

Functionalities	Support (M|O|C)	Description
Validate service request	M	Validate the service request performed by the user or the operator. Validation process implies the parsing of the request to check the syntax, the privilege level of the requestors and the exploitation of request resource availability and select domain options functionalities. If all the steps succeeded, the request is fully validated, and the service is ready to be provisioned.

7.2.2.2.3 Inputs and Outputs

Inputs (from):

- Users: Application, Operators, etc. that exploit PREDICT-6G framework to establish E2E deterministic services. Additionally, this service could also receive termination requests from the user/operator that would need to be validated.
- E2E Service Automation: E2E service ingestion receives the confirmation for the E2E service provisioning or decommissioning.

Outputs (to):

- E2E Service Automation: requests E2E service provisioning or decommissioning.
- Users: E2E Service Ingestion confirms the user, application or operator the success or failure of the E2E service provisioning or decommissioning request.

7.2.2.3 E2E Service Automation

7.2.2.3.1 Overview

The E2E Service Automation is a MS designed to manage E2E closed-loop service automation and conflict resolution for all technology domains. This MS supervises the E2E service assurance of all services in each technology domain, tackles conflict resolution for each technology domain and in case of need, escalates conflicts towards the user or the operator for further decisions.

7.2.2.3.2 Exposed Functionalities

In line with the description of the service, the exposed functionalities for this service are detailed below:

Table 7-11. List of functionalities provided by E2E Service Automation

Functionalities	Support (M|O|C)	Description
Request cross-domain E2E path	M	Request and obtain information about the E2E path across the set of domains.
Define and provision the service	M	Definition of the service as E2E object following a specific information model and its decomposition in sub-services to be provisioned and stitched across a set of selected technological domains
E2E Service Assurance	M	Perform E2E service assurance via closed-loop automation for each service running in the different technology domains. To perform E2E service assurance this MS will need to interact with technology specific Service Automation MS.
Notify E2E service notification	M	Inform E2E Learning Orchestrator and E2E DT Predictive Analytics about the service provisioning or decommissioning
Configure E2E monitoring	M	Configure the E2E Monitoring MS after the service provisioning or decommissioning
Store E2E service information	M	Add or remove information related to the service provisioned or decommissioned.
Predict/detect and solve E2E conflicts	C	Detect, predict, and solve conflicts in specific technology domains via E2E actions. In case conflicts cannot be solved, the MS could escalate them to the user/operator.

E2E Service Termination	C	Perform E2E service termination under specific requests of the user/operator in case of conflicts.

7.2.2.3.3 Inputs and Outputs

To expose all the aforementioned functionalities, the E2E Service Automation MS will need to interact with other MS. The set of interactions in the shape of inputs/outputs are:

Inputs (From):

E2E Service Ingestion: E2E Service Automation will ingest the E2E service provisioning or decommissioning request after the validation performed at the E2E Service Ingestion.

E2E Path Computation: E2E Service Automation will receive the E2E Path and Domain selection performed by the E2E Path Computation.

Service Automation: E2E Service Automation will ingest closed-loop information from each technology domain to perform service provisioning, assurance and for detecting local conflicts.

Outputs (To):

Service Automation: During the service provisioning, assurance and the potential service termination, E2E Service Automation needs to interact with each Service Automation MS running in the specific technology domain. Additionally, this MS will also solve local conflicts in technology domains from the E2E perspective.

E2E Path Computation: During the service provisioning, the E2E Service Automation requests the cross-domain E2E path to the E2E Path Computation MS.

E2E Learning Orchestrator: To notify the service provisioning or decommissioning.

E2E DT Predictive analytics: To notify the service provisioning or decommissioning.

E2E Monitoring: Configure E2E monitoring after the confirmation of an E2E service provisioning.

E2E Service Exposure: E2E Service Automation could forward general E2E service information related the service SLAs, the initial service configuration, and other service conditions to the E2E Service Exposure.

E2E Service Ingestion: Inform about the success of the E2E provisioning, termination or update requests.

7.2.2.4 E2E Path Computation

7.2.2.4.1 Overview

The E2E Path Computation MS is responsible for calculating routes over the E2E domain and driving the path computation over technology-specific domains, which translates at the E2E level in the selection of the sequence of technological domains that will implement the E2E path. To do this, the E2E Path Computation MS requests the E2E AI/ML-based MSs to analyse the suitability of the computed domain sequence according to the capabilities of each domain and the characteristics of the requested service. Once the candidate domain sequence is correctly assessed, it is sent to the E2E DT Predictive analytics, which will compute the KPIs associated to an abstracted E2E path traversing the computed domain sequence. This process aims at maximizing the chances a feasible E2E path is finally computed. In the next step, the E2E Path Computation MS Interacts with the Service Automation MS of the selected technological domains to request local path computation and to retrieve local path calculations that will compose the candidate E2E path. Once E2E Path Computation collects the local path of each technological domain, it computes the complete E2E path and sends it to the E2E Service Automation MS for its configuration.

7.2.2.4.2 Exposed Functionalities

Table 7-12 enumerates and describes the functionalities and communication interfaces that must be implemented in the Path Computation MS.

Table 7-12. *List of functionalities provided by E2E Path Computation*

Functionalities	Support (M|O|C)	Description
Path Computation Request	M	Receive request from E2E Service Automation MS for performing path computation
Local Path Computation Request	M	Request, receive and process local path computation from Service Automation and Path Computation MSs in local technology domains.
Path Computation execution	M	Perform an initial path computation by leveraging capability, resource and topology information
Path Computation result communication	M	Return path calculation for each of the potential paths

Select domain options	M	Select the best domain options for deploying the service. The selection process uses as matching criteria the E2E service characteristic requested (e.g., QoS Parameters) to find an E2E path over different technological domains.
Request KPI provisioning feasibility and smart resource evaluation	M	Request the feasibility for the provisioning of specific KPIs and assess the existence of smart resources to meet the E2E service requirements.

7.2.2.4.3 Inputs and Outputs

Inputs: E2E Service Ingestion, Topology Exposure, Resource Exposure, Capability Exposure

Inputs (from)

- E2E Service Automation: this MS is the one that sends the path computation request to the Path Computation MS.
- Service Automation: This MS sends back local path computation results from each technology domain. This information is then used to compute E2E paths.
- E2E Learning Orchestrator: To receive the feasibility assessment of the computed E2E domain sequence.
- E2E DT Predictive analytics: To receive the KPI estimation associated to an E2E path traversing the computed E2E domain sequence.

Outputs (to):

- Service Automation: To request local path computation in each technology domain.
- E2E Learning Orchestrator: To send E2E domain sequence to obtain a feasibility assessment.
- E2E DT Predictive analytics: To send the E2E domain sequence to obtain a KPI estimation of the E2E path across them.
- E2E Service Automation: To send the computed E2E path for its configuration.

7.2.2.5 E2E Resource Manager

7.2.2.5.1 Overview

The E2E resource manager requests resource configuration in all the technology domains that constitute an ingested service.

7.2.2.5.2 Exposed Functionalities

Integrating these functionalities, the MS facilitates resource configuration across pertinent management domains thereby provisioning the ingested service.

Table 7-13. *List of functionalities provided by E2E Resource Manager*

Functionalities	Support (M|O|C)	Description
Resource orchestration	M	In response to a service ingestion, the MS issues configuration requests for resources across the management domains required to deliver a service. For radio resources, multi-access network slicing will constitute one of the approaches to be explored, extending cellular slicing (Wen Wu et al., 2022) to include Wi-Fi radio resources. An AI-based algorithm that receives as input the radio resources available in cellular and wi-fi access domains together with channel state information and the requested SLA for a service will coordinate the creation of network slices to fulfil those service demands. The new AI algorithm is developed and trained to configure the slices such that certain quality of service (QoS) is satisfied for different traffic types. The AI model will be developed and evaluated with the help of a proprietary simulator (not released). Inputs: r – Service throughput requirements l – Service latency requirements h_i – Radio channel state for i_{th} user. Outputs: θ_x – Reserved radio frequency resources for a network slice "x" fulfilling the requested SLA (throughput and latency).

Resource Life Cycle Management	M	The MS handles the commissioning, deployment and release of resources once a service has been completed.

7.2.2.5.3 Inputs and Outputs

Inputs (from):

E2E Service Automation: when a service is admitted for execution and E2E path and domains have been selected, the MS is alerted by the E2E Service Automation to request resource configuration in the selected technology domains. .

Outputs (to):

Resource Configuration: the MS issues resource configuration requests to all the participating management domains.

7.2.2.6 E2E Topology Exposure

7.2.2.6.1 Overview

Service used to expose topology information from the E2E perspective, i.e., abstracting topology information for all technology domains.

7.2.2.6.2 Exposed Functionalities

The E2E Topology Exposure service aggregates the retrieved information by integrated the data collected from different domains by the per-domain Topology Exposure services, so building more complete and complex topological relationships.

Table 7-14. List of functionalities provided by E2E Topology Exposure

Functionalities	Support (M|O|C)	Description
Ingest abstracted topology information	M	Ingest abstracted topology information from the Topology Exposure MS from each technology domain
Expose abstracted topology information	M	Expose abstracted information on endpoints (ingress/egress points) per domain to other E2E MS through a dedicated interface

7.2.2.6.3 Inputs and Outputs

Inputs (from):

- Topology Exposure: the information from each Topology Exposure service will be retrieved and conveniently processed for building the E2E overall view of the network.

Outputs (to):

- E2E DT Predictive Analytics: The E2E Topology Exposure service will feed the DT Predictive Analytics for the purpose of obtaining predictions on the overall behaviour of the network.

7.2.2.7 E2E Service Exposure

7.2.2.7.1 Overview

The E2E Service Exposure is a MS designed to expose service information from the E2E perspective to the user or the operator in the shape of feedback related to service events. The feedback provided by this MS could be helpful for informing the user and the operator about the status and the conditions of a specific service, leading potentially to further actions as service request updates or terminations. To provide such feedback, the E2E Service Exposure interacts with several E2E and technology specific MSs such as the Service Exposure MS of each technology domain or the E2E Service Automation MS.

7.2.2.7.2 Exposed Functionalities

In line with the description of the service, the exposed functionalities for this service are detailed below:

Table 7-15. List of functionalities provided by E2E Service Exposure

Functionalities	Support (M|O)	Description
Ingest abstracted service information	M	Ingest abstracted service information from the Service Exposure MS for each technology domain and information related to service SLAs and other service conditions from the E2E Service Automation MS.
Map abstracted information	M	Adapt the abstracted information received from the different MS and map it in a common data model for exposing it to the user/operator.

Expose abstracted information	M	Expose abstracted information related to the service general status and conditions to the user or the operator through a dedicated interface.

7.2.2.7.3 Inputs and Outputs

Inputs (from):

- Service Exposure MS: E2E Service Exposure MS ingests abstracted information from each Service Exposure MS available in specific technology domains for retrieving basic information such as the service type, the owner of the service, etc.
- E2E Service Automation MS: E2E Service Exposure MS also ingests information from the E2E Service Automation MS for retrieving the status and the conditions of the service running in the different technology domains.

Outputs (to):

- User/Operator: E2E Service Exposure MS exposes abstracted information related to the service status and conditions to the user or the operator through a dedicated interface.

8 PREDICT-6G Management Services vs Technical Domain functions.

8.1 Functional Mapping

The AICP manages E2E deterministic services across multiple technology domains, each having different APIs and capabilities with regards to programming their deterministic capabilities. In order to have a scalable and extendible architecture, the AICP employs technology specific MSs to encapsulate interaction with specific technology domains. The technology specific MSs provide an abstraction of their respective domain towards the E2E MSs that are responsible to create and maintain E2E cross-domain deterministic services.

In order to implement technology specific MSs, the capabilities and North-bound (N-bound) APIs (i.e., which enable network components to interact with higher-level components, e.g., managed entities with a manager entity) of each technology domain should be integrated with the South-bond (S-bound) APIs (i.e., which enable network components to interact with lower-level components) of the technology specific MSs. The approach for PREDICT-6G is to create Open APIs on top of existing technical capabilities (e.g., 3GPP, IETF DetNet) that encapsulate the programmability of the technologies, and leverage these Open APIs in the AICP's technology specific MSs. The responsibility of creating the Open APIs is in WP2 as part of the MDP, and the first delivery that contains the first drop of the Open APIs will be D2.2. The Open APIs are defined as a joint interworking between WP2 and WP3, with WP3 providing requirements towards the APIs (based on the analysis of the MS inputs, ref Section 4-7) and WP2 providing the binding of such input requirements onto existing technological APIs and interfaces. The implementation of the Open APIs will be part of the code of each technology specific MS, which means that on the technology domain level, each MS may have alternative implementations depending on the technology domain they are managing. Practically, this may mean separate MS implementations per technology domain, or an MS implementation with a modular internal structure that can adapt its behaviour towards the right Open APIs. This is a matter or packaging MS functionality and will be decided during AICP implementation and integration (part of WP4 Task 4.1).

<table>
<tr><th rowspan="2">Technology specific MS in AICP</th><th colspan="4">Technology domain</th></tr>
<tr><th>3GPP</th><th>IETF DetNet</th><th>Wi-Fi</th><th>Etc.</th></tr>
<tr><td>Time sync</td><td colspan="4" rowspan="8">1. Defined by cross-WP2-WP3
• WP3 to provide per-MS requirements on input/output/capabilities towards tech. domain
• WP2 to enable those requirements within the respective technology domain
2. Provided by WP2 as MDP Open APIs
3. Consumed by WP3 as tech. spec. MS S-bound</td></tr>
<tr><td>Measurement collection</td></tr>
<tr><td>Service exposure</td></tr>
<tr><td>Resource Config.</td></tr>
<tr><td>Service automation</td></tr>
<tr><td>Topology exposure</td></tr>
<tr><td>Capability exposure</td></tr>
<tr><td>Resource exposure</td></tr>
</table>

Figure 8-1 *Open API matrix between MDP and AICP*

The Open API matrix is shown in **Figure 8-1**, indicating which technology specific MS is foreseen to require integration with an existing technological capability in a targeted technology domain via the MDP Open APIs. In each technology domain, the Open API should bound to the APIs or interfaces of the network elements or functions which provide useful capabilities for the scope of the MSs.

The following list provides an initial understanding of the technology domain specific entities which may play a critical role as entry point into the technology domain's programmability.

8.1.1 3GPP

Time Sensitive communication Time Synchronization function (TSCTSF)

The TSCTSF Control-plane (C-plane) network function enables to interact with a 3GPP network in order to (1) manage its clock synchronization and (2) provide control over deterministic QoS provisioning. For the clock synchronization, the 3GPP network may be configured to act as a Grandmaster (GM) clock and distribute synchronization via gPTP to slave clocks in other adjacent deterministic domains. As for QoS provisioning, the TSCTSF enables an external Application Function (AF) to define deterministic QoS characteristics to the 3GPP network, which will be applicable to selected QoS flows in PDU sessions. The QoS characteristics are provided in a form of Time Sensitive Communication Assisted Information (TSCAI) that describes traffic characteristics and requirements of deterministic data plane flows. The 3GPP network will derive policies based on the TSCAI and additional regular QoS parameters (such as priority and bitrates) and distribute the policies to the access and core networks. Within the interactions through the TSCTSF, the 3GPP technology specific MSs may act as AFs and thus access the 3GPP defined APIs of the TSCTSF (e.g., an AICP MS may be able to provide a TSCAI when PREDICT-6G ingests a new service request and, as part of the E2E deterministic service creation, the 3GPP network segment needs to be provisioned and configured).

Network Exposure Function (NEF)

Interaction with the 3GPP NEF is required if the AF interacting with the 3GPP network is not trusted. The NEF may expose the same APIs and capabilities as if the AF was interacting with the TSCTSF, through additional authentication and authorization mechanisms. In PREDICT-6G, we may assume that 3GPP networks that become part of the E2E domains treat PREDICT-6G AICP MSs as trusted AFs, therefore the additional complexity of traversing the NEF can be omitted. Including the NEF is however technically possible by implementing extra NEF APIs at the AICP side and could be an extension considered for later development of the AICP.

Operation, Administration and Maintenance (OAM)

The OAM of the 3GPP network may serve as a source for Configuration Management (CM), Performance Management (PM) and Fault Management (FM, also known as alarms) data. Collecting CM/PM data is important for abstracting and exposing resources, capabilities and high-level topological information about the 3GPP domain to the E2E MSs.

Additional functions

In order to collect U-plane measurements with the necessary real-time scale and granularity (e.g., per IP flow), it may be necessary to deploy dedicated probe-like measurement functions at the boundaries or within the U-plane of the 3GPP network. Creating such measurement functions are in the scope of WP2.

8.1.2 IETF DetNet

Topology and resource exposure functions

The IETF Interior Gateway Protocol (IGP) may be used collect topology and resource capabilities in the domain when the control plane employs a distributed architecture. These data will serve as input to the path computation function to establish a DetNet flow. In the case of a centralised architecture, topology and resource capability data may be obtained by use of NETCONF (Network Configuration Protocol), which "provides mechanisms to install, manipulate, and delete the configuration of network devices. It uses an Extensible Markup Language (XML)-based data encoding for the configuration data as well as the protocol messages (RFC6241, 2011) and YANG, which is "a data modelling language used to model configuration data, state data, Remote Procedure Calls, and notifications for network management protocols." (RFC7950, 2016)

Resource reservation and configuration functions

Once optimal paths through the domain have been computed, a reserve (RESV) message requesting the reservation of resources is sent to the controller. The latter can then configure resources along the specified paths. To eliminate loss, the control plane will also need to support PREOF (Packet Replication, Elimination, and Ordering Functions): replication of packets, their transit through multiple paths, reordering and the elimination of duplicates.

8.1.3 Wi-Fi

The 802.11 working group has been working with the 802.1 TSN group for several years on adding support for TSN features over wireless media enabled by the 802.11 MAC/PHY. The time synchronization and traffic shaping have been the main focus given their wide applicability to several use cases and markets.

Time synchronization

The IEEE 802.1AS standards define a profile of the IEEE 1588 Precision Time Protocol (IEEE 1588, 2008) that can be used for time synchronization across wired and wireless Local Area Networks (LANs). The 802.1AS-based time synchronization protocol is supported over the 802.11 medium through the Timing Measurement and the Fine Timing Measurement capabilities defined in the 802.11 MAC. FTM has been introduced more recently to increase accuracy of the time stamping capabilities on 802.11 devices. As the 802.1AS protocol (IEEE 802.1AS-2020, 2020) is defined to operate over wired (Ethernet) and wireless (802.11) links, it is possible to achieve E2E time synchronization across heterogeneous wired/wireless links and networks.

Traffic shaping

The standard IEEE 802.1Qbv (IEEE 802.1Qbv-2015, 2015) defines the concepts of priority queues and gates controlled by a scheduler based on a synchronized time reference across the network devices. 802.1Qbv was originally defined for operation over Ethernet links, but it can also be implemented over Wi-Fi and take advantages of the latest capabilities of the 802.11 MAC layer to schedule data transmissions with higher reliability and determinism. The time-aware shaping function must be coordinated by the network to configure each device with appropriate time windows for transmission of packages (managed network approach).

Redundancy

The standard IEEE 802.1CB defines (IEEE 802.1CB-2017, 2017) the redundancy capability of TSNs to provide path diversity and eliminate outages or delays due to events like roaming and interference in a mobile scenario. This capability can be implemented in Wi-Fi through a multi NIC implementation, where the functionality of 802.1CB is implemented on top of the MAC layer. In Wi-Fi 7, the new multi-link operation feature (MLO) enables the possibility to implement redundancy with a single NIC.

8.2 Data and Network Modelling

In this section we discuss the innovation related to data and network modelling used throughout the PREDIC-6G architecture. Being AICP a data driven intelligent network and service management approach, first we concern ourselves with data modelling issues including data collection and distribution as the basic enabler of any other AICP management services: configuration data needs to be collected to be able to manage the E2E deterministic services and performance related data is required to run closed loop automation for service assurance. Furthermore, AICP needs network topology information which can also be acquired through data services.

Second the network modelling issues are discussed using the principles of data collection and modelling. AICP needs to have a complete view of the network topology to provide services that span across multiple technology domains.

8.2.1 Data Modelling and Collection Framework (DMCF)

PREDICT-6G features an efficient, dynamically programmable, extendable multi-technology Data Modelling and Collection Framework (DMCF, **Figure 8-2**) that supports all types of TSC relevant (PM, CM, topology, etc.) data produced by any technology domain and provides a uniform data representation for E2E service management functions.

DMCF is a sub-system of the AICP and as such its architecture is aligned with it.

In this chapter we identify the enablers of such a data collection framework, define its high-level functional architecture and map it to the overall PREDICT-6G architecture.

First, we define the requirements and design considerations of the PREDICT-6G DMCF:

- **Commonality:** applies to the collection of any data (trace, PM, CM, topology, etc., both standardized and custom) from any data source in any technology domain that is relevant for TSC service lifecycle management.
- **Extendibility:** new data types, sources, consumers, and technology domains can be added without architecture impact
- **Programmability:** data consumers can discover available data and can dynamically specify (and modify) when and what data they want to receive.
- **Efficiency:** Data sources should not produce data that is not consumed at all, and they should not need to replicate data due to multiple consumers
- **Feasibility:** Align with existing implementation trends, standards.

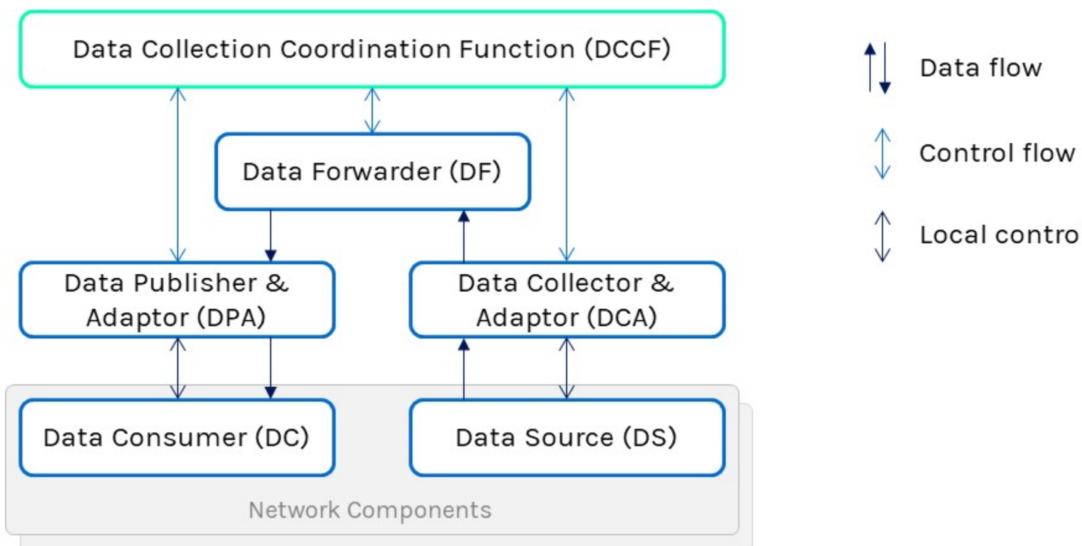

Figure 8-2 Conceptual (technology agnostic) architecture of the DMCF

Figure 8-2 shows the conceptual (technology agnostic) architecture of the DMCF. Note that this is deliberately not yet mapped to the PREDICT-6G architecture to enable the technology agnostic discussion of the framework. The architecture consists of the following functional elements:

- **Data Source (DS)**: producer of data
- **Data Collector & Adaptor (DCA)**: registers data source (at DCCF); triggers data production (on request from DCCF); adapts data source format and i/f to DF.

- **Data Forwarder (DF)**: Data replication, routing and forwarding (to multiple consumers)
- **Data Publisher & Adaptor (DPA)**: registers (at DCCF); receives request for data (from DC); requests data (from DCCF); receives data (from DF); adapts data format and i/f to DC.
- **Data Consumer (DC)**: finds/requests available data (via DCCF); consumes data (via DF/DPA)
- **Data Collection Coordination Function (DCCF)**: facilitates data discovery and routing requests; it does not handle data itself.

The Data Collector & Adaptor (DCA) and Data Source (DS) (**Figure 8-3**) performs the following functions: The DCA oversees one or more DSs and facilitates data discoverability by registering potentially available data with the DCCF. It triggers the generation of data from the DS(s) based on requests from the DCCF, thus the same data is produced only once, regardless of the number of consumers (refer to DF). The DCA ensures that the appropriate metadata is associated with the data. It allows integration of legacy DSs into the framework without affecting them. The interface between the DS and DCA can be either standard or proprietary. It is also possible to combine the DCA and DS into a single entity, depending on the implementation choice. This is applicable if the entity can produce data with correct metadata and handling the DCCF interface.

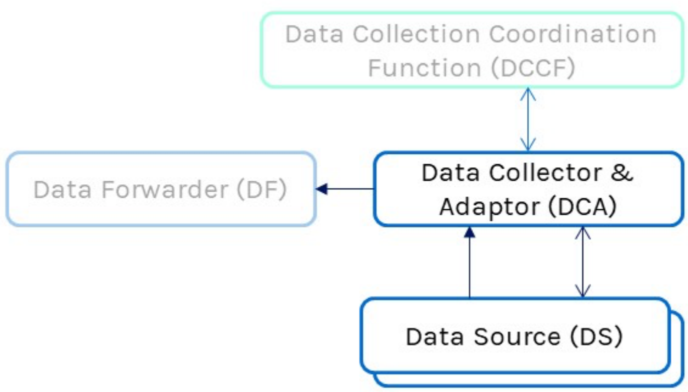

Figure 8-3 *The Data Collector & Adaptor (DCA) and Data Source (DS)*

The **Data Forwarder** (DF, Figure 8-4) is in charge of managing the efficient distribution and duplication of data. It receives data from various DCAs and duplicates the data to fulfil requests from DCs. The technology used for data forwarding is designed to be modular. The implementation of the DF data plane is concealed from DSs and DCs using DCAs and DPAs. DSs and DCs may directly interact with DF through DCA and DPA interfaces. However, in cases where the DS/DCs follow existing standard protocols, a DCA/DPA frontend is required to bridge the standard interface with DF's APIs.

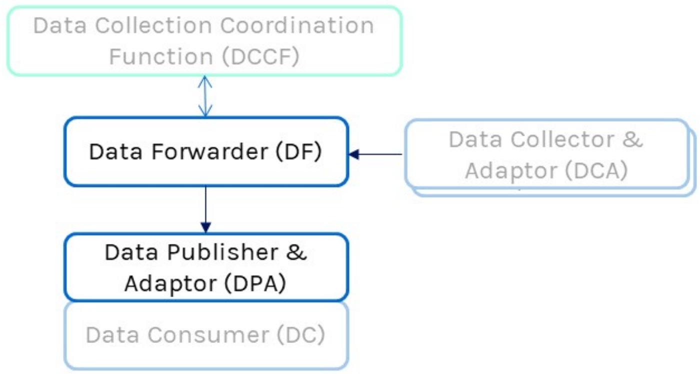

Figure 8-4 The Data Forwarder

The Data Consumer (DC) utilizes the (optional) **Data Publisher & Adaptor** (DPA) (**Figure 8-5**) to request data from the DCCF. The DPA allows the DC to receive data using its existing interface, which may already adhere to standard protocols, regardless of the protocol or mechanism implemented by DF.

The DC, along with the DPA, can inquire about the availability of data from various sources (DCAs) through the DCCF. The DC has the flexibility to dynamically update or modify its data requests as needed.

Data is received by the DC directly from the DF, optionally through the DPA. The received data can take various forms, such as real-time events, or it can be in the form of time series data (e.g., aggregated performance monitoring), traces, streams, and so on, which are received as they are generated.

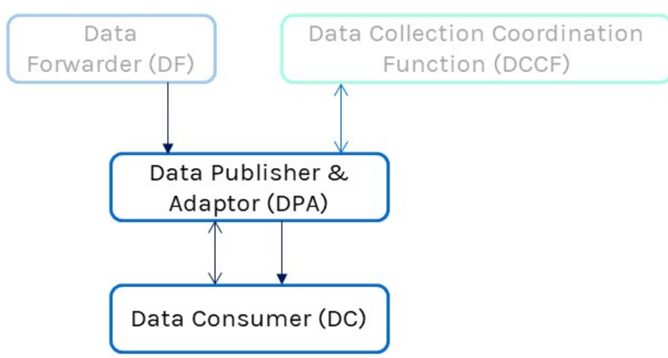

Figure 8-5 The Data Consumer (DC) and the Data Publisher & Adaptor (DPA)

Data Collection Coordination Function (DCCF, Figure 8-6): the responsibility of managing data source registrations lies with the DCCF. Each DCA registers the data available from its associated DS(s) with the DCCF. The DCA also keeps the DCCF updated about any changes in data availability.

The DCCF handles consumer requests either directly or through the DPA. The DC can query the potentially available DSs and access information about data types, formats, and other relevant details. Based on the availability of data, the DC can request specific datasets.

Additionally, the DC can request notifications regarding the availability or unavailability of potentially accessible data. The DCCF ensures that all data, DSs, and DCs are programmatically identifiable through a process of discovery and cataloguing. It manages unique identifiers (IDs) and metadata as part of the registration process for sources and consumers.

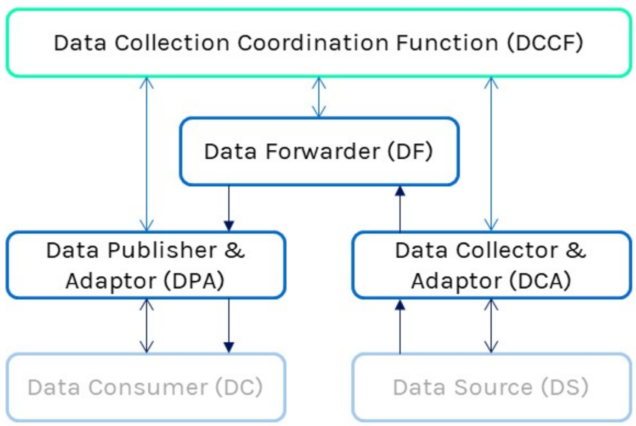

Figure 8-6 The Data Collection Coordination Function

Figure 8-7 shows the Data Source and Data Consumer and Data Forwarder registration as conceptual procedures in the framework outlined above. During registration the components of the system let the DCCF know about their existence, availability, and capabilities. Most importantly the Data Collector & Adaptor describes the data that can be produced by the associate Data Sources. Data Consumer provide delivery information (optionally: authenticate, access control, etc.)

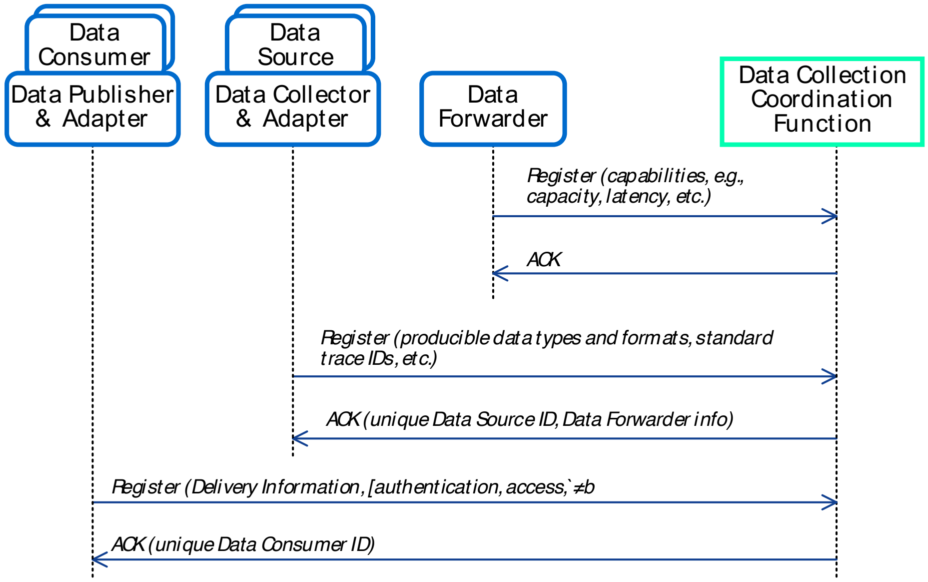

Figure 8-7 Registration procedures

Figure 8-8 shows the conceptual data discovery and request procedure. Data Consumers can initiate data discovery through their associated DPA. They query the DCCF for the list of producible data. The DCCF responds with the list of available, producible data. The DCCF also ensures to notify registered DPAs in case changes in availability of data sources. Based on the list of available data the DPA can make data requests towards the DCCF. The DCCF act as a proxy between the DPA and DCA triggering the start of data production at the corresponding Data Sources (optionally also configure Data Forwarders in between). Finally, data starts to flow between the source and the consumer.

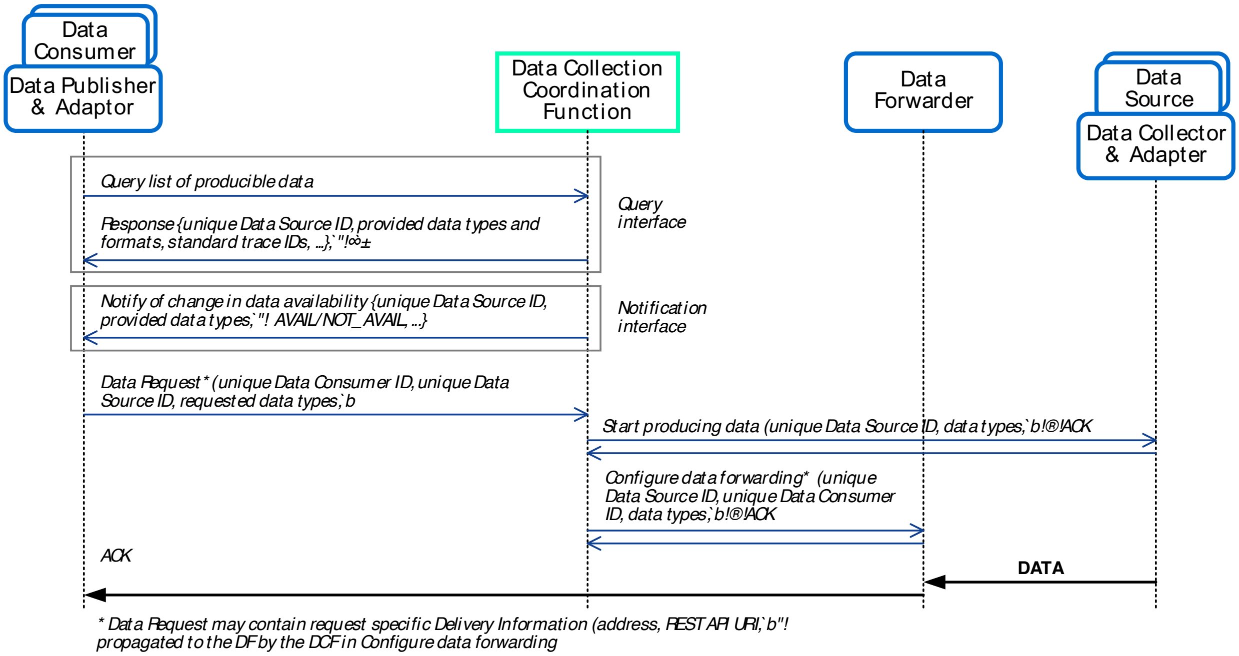

Figure 8-8 Data discovery and request procedure

Chaining of Data Sources and Data Consumers: the connection between Data Sources and Data Consumers, as shown in **Figure 8-9**, involves a process where analytics functions serve as both consumers and producers of data. In addition to consuming multiple input data, these functions generate insights or algorithm outputs. By treating these outputs as Data Sources within the framework, with integrated or separate Data Consumer Applications (DCA), it becomes possible to dynamically link multiple analytics functions together, including the option to store the outputs persistently in Data Lakes (see next section).

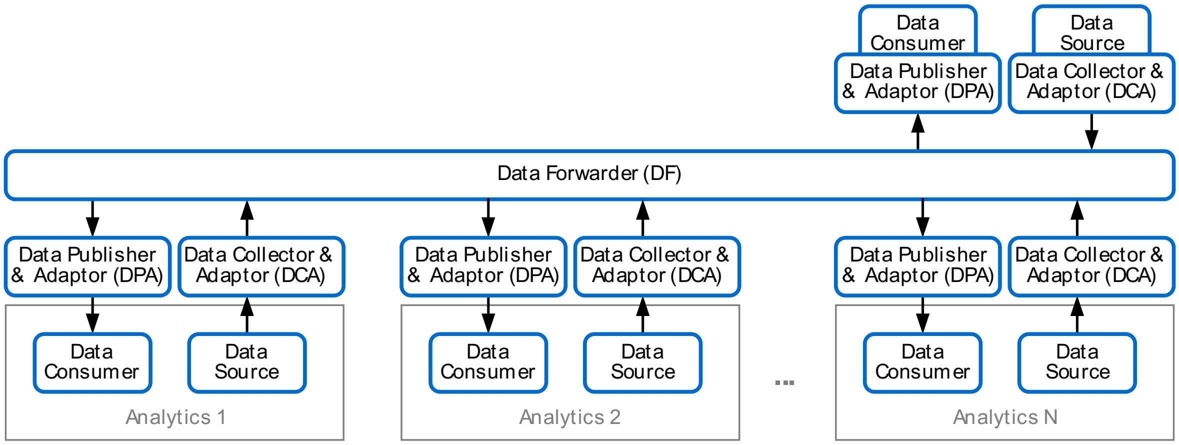

Figure 8-9 *Chaining of Data Sources and Data Consumers*

Integration of Data Lakes (DL): The capability to store data in Data Lakes (DL) should be accessible regardless of requests from other Data Consumers. In certain cases, data may not have an immediate consumer but still needs to be stored for future analysis, such as network and service profiling. DL integration (**Figure 8-10**) enables persistent storage with a single consumer, initiated through a separate request from the DL side. The Data Forwarder has the option to directly access the DL or utilize a Data Publisher & Adaptor frontend implemented at the DL to simplify the complexity of the Data Forwarder. The DL's Data Publisher & Adaptor frontend conceals the underlying storage technology of the DL, such as HDFS, SQL, Cassandra, and so on. Persistent data can be retrieved through a Data Consumer Application (DCA) frontend, which exposes the content of the DL as a Data Source. Direct access to the DL by Data Consumers can enhance the efficiency of big data analytics, especially when extensive processing of historical data is required, as both storage and computation take place on the same cloud platform.

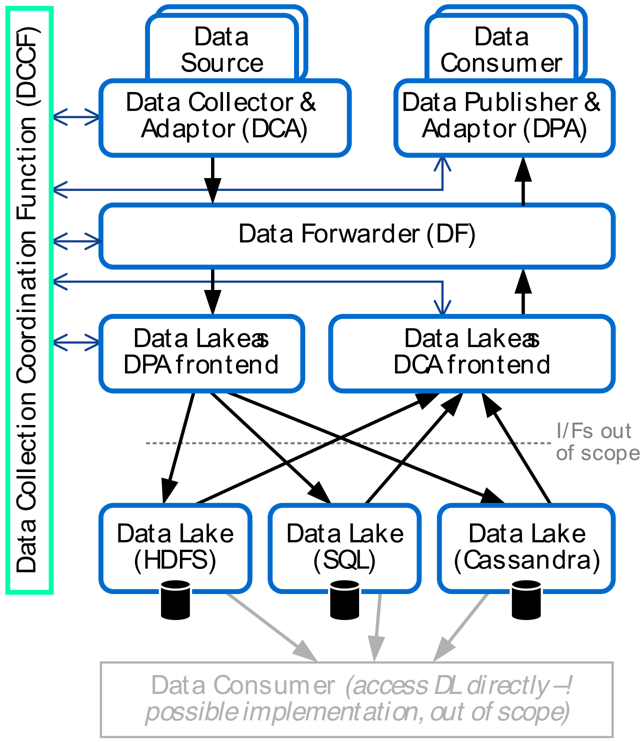

Figure 8-10 Data Lake integration

Figure 8-11 shows one possible mapping of the Data Modelling and Collection Framework (DMCF) to the PREDICT-6G architecture. This layered, hierarchical architecture that, as the AICP itself, has a technology domain specific layer and an E2E layer. Each technology domain has its technology specific Data Sources (optional DCAs are not shown for clarity), which can register itself to the technology specific CDDF and can provide data through its technology specific Data Forwarders. Each technology domain has a special Data Modeler (MD) that consumes data (i.e., it is a Data Consumer, DPA not shown for clarity) that can subscribe to the technology specific Data Sources and convert them to technology agnostic data according to data modelling rules. The MD also acts as a DS towards the E2E layer: it registers itself to the E2E DCCF and provide technology agnostic data towards the E2E Data Consumers via the E2E Data Forwarder.

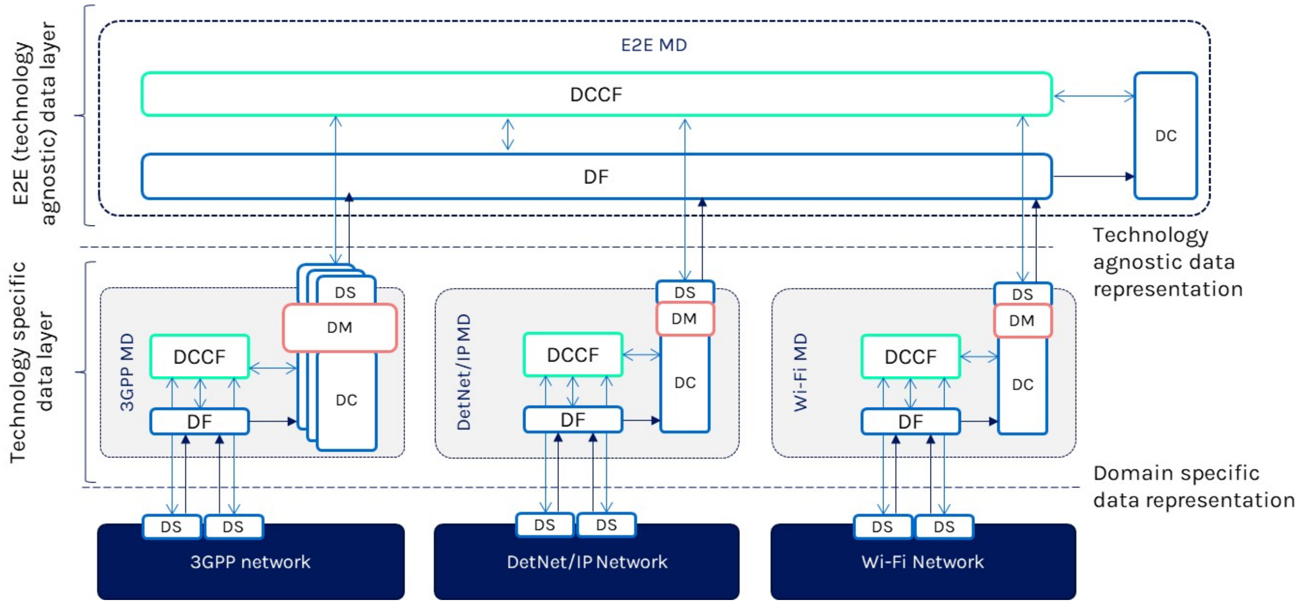

Figure 8-11 Mapping the Data Modelling and Collection Framework (DMCF) to the PREDICT-6G architecture.

Note that each technology domain can have Data Consumers and each domain specific data layer can have more than one DC and multiple technology agnostic DS.

Integration with Management Services

The Data Modelling and Collection Framework (DMCF) described above is a technical enabler for a few MSs identified in the previous chapters (**Table 8-1**):

Table 8-1. Mapping with Management Services

Chapter	Management Function	DMCF enablers
4.2.5	Dataset Registry	DCA, DPA, DCCF
4.2.7	Dataset Repository	Data Lake integration
6.2.1	Measurement Collection	DCA, DPA, DS, DC, DF
6.2.2	E2E Monitoring	DCA, DPA, DS, DC, DF and DCCF

8.2.2 Network Modelling

8.2.2.1 Domain-level network modelling

A domain can be modelled with its topology, i.e., nodes and links between the nodes, and the resource and capability related information related to the nodes and the links. In Predict-6G the topology of a domain contains the domain-specific service endpoints the logical links connecting these entities and the E2E service endpoints connected to the domain. The domain itself and each E2E and domain-specific service endpoint as well as the logical links are associated with a unique identifier. Logical links furthermore have at least three additional attributes: source endpoint identifier, destination endpoint identifier and link direction. In case the link is bidirectional data can flow both directions, i.e., from the source endpoint to the destination endpoint and vice versa. In case the link is unidirectional data can only flow from the source to the destination.

Topology links can be associated with further information: link type (wired, wireless, etc.), capacity, delay.

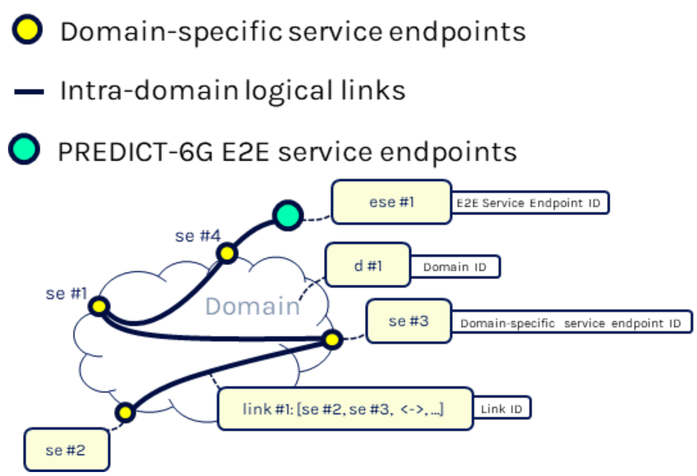

Figure 8-12 Topology model of a single domain

8.2.2.2 E2E network modelling

The E2E network model consists of domains and inter-domain logical links, i.e., the E2E topology. The inter-domain links are associated by their unique identifier and the service endpoints they are connecting as well as the link direction information the same way as in case of intra-domain links.

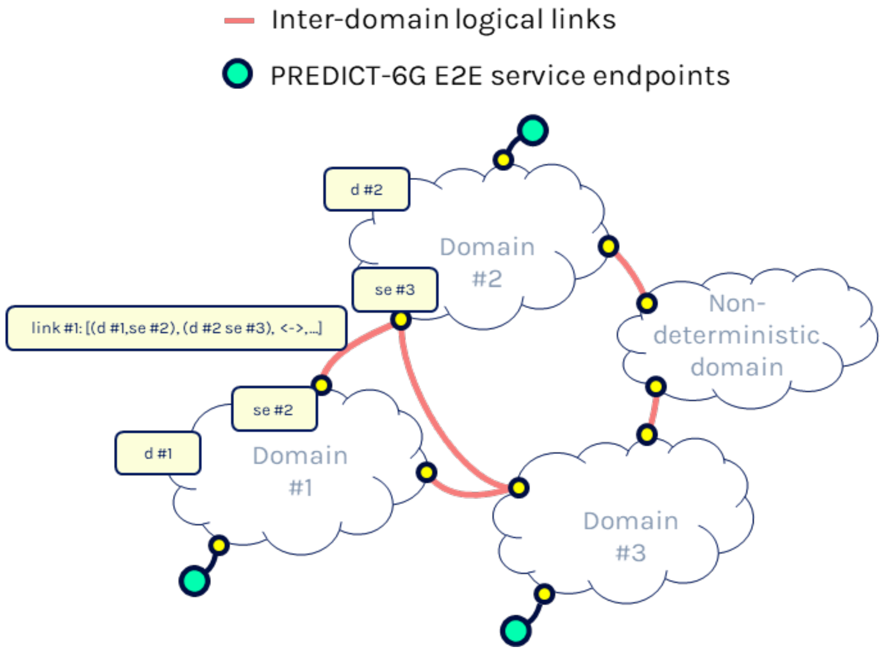

Figure 8-13 Inter-domain topology model

Note that inter domain links are implemented by Domain Border gateways that are service endpoints on at least two domains that they are connecting.

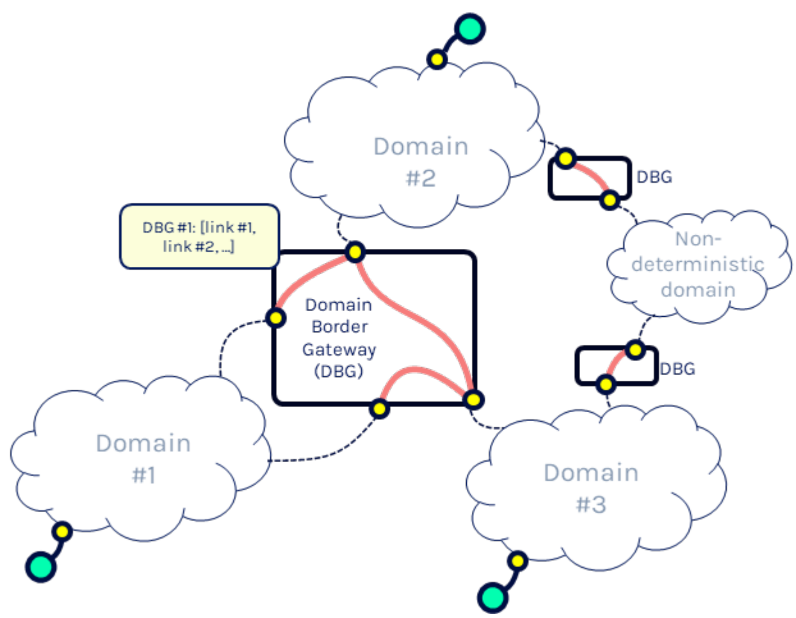

Figure 8-14 Domain Border Gateways implementing inter-domain links.

The Domain Border Gateways are associated with Layer 2 capabilities such as PAREO, PREOF, packet replication and the different schedulers (PREDICT-6G/D2.1/3, 2023) .

Finally, the on the top level a Predict 6G E2E deterministic network can be modelled with a single domain with E2E service endpoints and the E2E links, which connect these endpoints.

8.3 Non-deterministic domains

PREDICT-6G will provide E2E deterministic paths for services spanning through multiple heterogeneous network domains and technologies. Most of the domains targeted by PREDICT-6G (e.g., 3GPP domain, DetNet/IP domain) implement deterministic network capabilities, which are leveraged by the PREDICT-6G control and management plane through the Open APIs (see in D2.1 section 4.3 (PREDICT-6G/D2.1/4.3, 2023)) to compose and assure E2E services. In some cases, however, PREDICT-6G's E2E service needs to extend over domains that lack inherent deterministic capabilities, e.g., by only providing best effort services over a routed IP network. Such network segments may still need to be integrated in PREDICT-6G's E2E service as either they are the only available domains through which two deterministic network segments can be physically connected, or there is a demand to connect to legacy devices with no deterministic capabilities either in the devices themselves or in the last mile connecting them to the first deterministic device.

This section concentrates on the control and management plane aspects of non-deterministic network domains, i.e., how a non-deterministic domain MF interworks with E2E MF and specific domain controllers (SDN controller, device managers, etc.). The requirements for data plane on both the device and network sides are discussed in (PREDICT-6G/D2.1/4.2, 2023).

To provide some level of determinism in non-deterministic network domains, a potential approach is to try to keep the domain in a state where its intrinsic resources and user plane mechanisms cause acceptable level of performance both in terms of reliability (mostly measurable by packet loss) and delay/jitter. Depending on the information available on the domain's internal resources, topology and potential QoS/scheduler mechanisms, these could be considered and even controlled to improve the deterministic goals of low and stable latency as well as no packet loss. Inferring and learning the correlation between the amount of traffic load, observed latencies/losses, throughputs and the configuration of the domain is possible by either active probing (e.g., generating dedicated traffic for the sake of measurement) or passive probing (e.g., measuring the traffic that is traversing the domain) via measurement agents deployed in the user plane (e.g., at the edge of the network domain). Controlling the amount of traffic and pre-scheduling/-shaping the traffic flows at the domain edge to better

Funded by the European Union

align with the domain's internal traffic management mechanisms as well as with the deterministic requirements of the traffic flows is also an option.

Depending on the level of the availability of the above information different scenarios may be considered:

1. **Only self-generated traffic is under control at the edge devices.**

   In this case, the other, unknown existing and upcoming traffic in the network could cause non-controllable congestion, which disables the support of determinism for self-generated traffic. Thus, due to the lack of information this scenario is not suitable to guarantee any determinism.

2. **All the traffic is under control at the edge devices.**

   Although the topology of the network is unknown in this scenario, all the traffic is controllable by admission control mechanisms at the edge devices, which is the only tool to maintain the network in a non-congested state and thereby to keep the level of determinism for the services. The operating points, where the drop of traffic is started, could be discovered by dedicated measurement traffic or collected historical measurements for different QoS classes (traffic with different DSCP or TOS value). Based on the measurements, session groups could be defined, where the services belong to any session in a group are congested at once somewhere in an unknown network device, because the routes of these services contain at least one common node or link. One session, which is defined by two edge devices in its endpoints, may take part in more groups. For these session groups the determinism could be supported by admission control at the edge devices on the way that an upcoming service is rejected if any session group for its session is beyond the operating point of the group, i.e., the network goes to congested state on the common node or link of this group. For example, assuming that A-B and C-D sessions are in a session group, if a service belonging to A-B session is served by the network and the group will achieve its operating point by the acceptance of an upcoming service belonging to C-D session, the new service will be rejected. The admission control of services could be solved as a sequential problem for upcoming services one by one or as an optimization problem for a burst of upcoming services.

3. **All the traffic is under control at the edge devices and network topology is known.**

   In this scenario the network itself is not a black box anymore as in the previous scenarios, because the devices and links of the network are known via a domain controller (e.g., SDN controller). Two sub-scenarios may be differentiated in this case:

A. Network topology is known, but QoS mechanisms are not accessible for the domain MF.

In this case the admission control of services is still the only method to maintain the non-congested state of the network, however based on the information about the resource and QoS capabilities of network entities (e.g., link utilization, WFQ weights, shaper and policer settings, etc.) the session groups could be defined more accurately, than in the previous scenario.

B. Network topology and QoS mechanisms are available for the domain MF.

In addition to the service admission control the domain MF may request the change of network and QoS configuration from domain controllers to maintain the non-congested state in the network. Based on the current state and the pre-calculated congestion points of the network the domain MF could send request to the domain controller to route the upcoming service over a less congested path; to mark the traffic as Strict Priority; to re-route lower priority services from the route of upcoming deterministic traffic; to set WFQ weights, shapers, policers to increase the available bandwidth for traffic in a certain QoS forwarding class.

In Scenario 2 and 3 based on the measured latency, throughput and jitter characteristics the achievable determinism of the network domain could be modelled. Supervised- and unsupervised ML techniques could be good candidates to determine the achievable determinism for a given traffic patterns in Scenario 2, extending with the configuration of the network (scheduler and WFQ weight configuration, traffic classes, policing and shaping parameters) in Scenario 3. Domain MF could promote the information about the highest-level of determinism for E2E MF. This information could be for example the minimum delay for a traffic class per session or the maximum bandwidth of acceptable service per session.

As a summary, although a TSN service could not be requested directly from a non-deterministic network, but implicitly via maintaining the network in a non-congested state by service admission control and configuration of QoS mechanisms (depending on the accessibility to network capabilities as introduced in the above defined scenarios), a certain level of determinism could be provided for the services. This may require dedicated measurement and scheduling agents deployed in the user plane at the edge of the non-deterministic domain to indirectly implement some of the functionality that could improve the determinism of flows traversing the domain. Integration with such additional functionalities could try to follow the MS-based logic that is defined in Section 3.2; however, some changes may be necessary therefore no strict compliance with the management architecture is going to be enforced for controlling non-deterministic domains.

9 Operational workflows

As AICP follows a service-centric architectural paradigm, one MS can theoretically interact with any other MS belonging to the control plane. Nevertheless, in the practice, only a subset of such interactions actually happens during the execution of the control plane workflows. The following subsections describe as examples the interactions between MSs during i) the typical operations for the lifecycle management of an E2E Deterministic Service (Provisioning, Decommissioning) and ii) the AI/ML focused operations.

9.1 E2E Deterministic Service Provisioning

The E2E Service provisioning operational workflow describes the interactions between multiple MSs involved in the E2E and technology-specific MDs. The workflow is first described and illustrated in **Figure 9-1** from the E2E perspective, and then, analysed step by step, including specific workflows for loops in local MDs (**Figure 9-2** and **Figure 9-3**). The AI/ML MSs are here indicated as AI/ML AI-based & Predictive Decision Service (E2E and not E2E) in a single box per domain, in order to ease the readability of the figures.

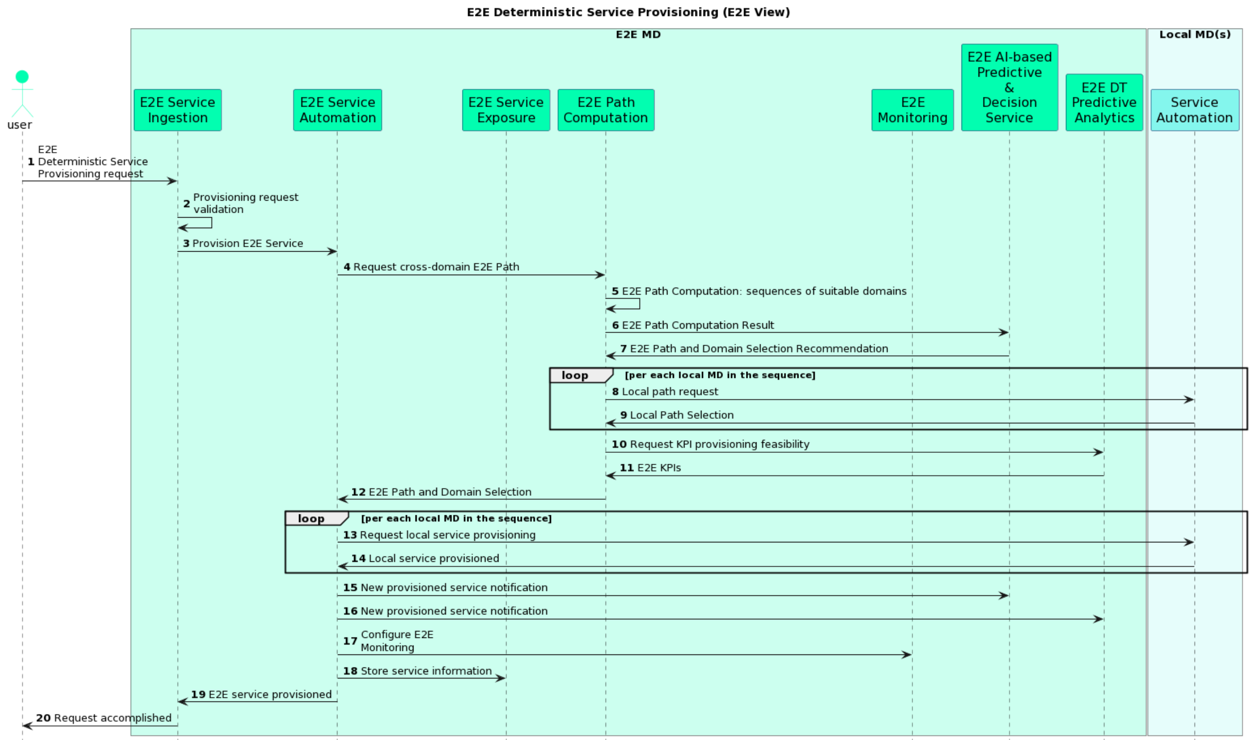

Figure 9-1 E2E Deterministic Service provisioning - E2E Management Domain view

- **Step 1.** *User/Operator* sends a Service Provisioning Request to the *E2E Service Ingestion MS.*
- **Step 2.** *E2E Service Ingestion* validates the format and the syntax of the Service Provisioning Request.
- **Step 3.** *E2E Service Ingestion* forwards the Service Provisioning Request to the *E2E Service Automation* for launching the provisioning process**.**
- **Step 4.** *E2E Service Automation* kicks off the provisioning process by requesting the cross-domain E2E path to the *E2E Path Computation.*
- **Step 5.** *E2E Path Computation* computes gross-grained E2E paths where the nodes in the network graph have the granularity of a domain, interconnected by the existing links between domain's border nodes.
- **Step 6.** *E2E Path Computation* sends the results to the *E2E Learning Orchestrator* for their evaluation.

- **Step 7.** *E2E Learning Orchestrator* recommends the best path and domain selection for the E2E, considering the path computation alternatives, and forwards this information to the *E2E Path Computation*.
- **Step 8** and **9.** After preselecting the domain and the path, the *E2E Path Computation* requests an E2E KPI computation to the *E2E DT Predictive Analytics MS, based in the KPIs computed for each local domain*.
- **Step 8.** *E2E Path Computation* starts an iterative path computation process in each local Management Domain by interacting with the corresponding technology-specific *Service Automation MS*. See details in **Figure 9-2**.
- **Step 9.** *E2E Path Computation* receives the local path selection from the different technology domains.
- **Step 10** and **11.** After preselecting the domain and the path, the *E2E Path Computation* requests an E2E KPI computation to the *E2E DT Predictive Analytics MS, based in the KPIs computed for each local domain*.
- **Step 12.** *E2E Path Computation* forwards the *E2E Path Computation* result to the *E2E Service Automation*.
- **Step 13.** *E2E Service Automation* triggers an iterative process for requesting the service provisioning in the corresponding domain through an interaction with the *Service Automation MS*.
- **Step 14.** *E2E Service Automation* receives feedback about the local service provisioning from the *Service Automation MS*.
- **Step 15.** *E2E Service Automation* sends a notification about the E2E service provisioning to the *E2E Learning Orchestrator.*
- **Step 16.** *E2E Service Automation* sends a notification about the E2E service provisioning to the *E2E DT Predictive Analytics MS***.**
- **Step 17.** *E2E Service Automation* configures *E2E Monitoring* to collect from *Measurement Collection MSs* of each domain.
- **Step 18.** *E2E Service Automation* stores the E2E Service Information in the *E2E Service Exposure.*
- **Step 19.** *E2E Service Automation* informs about the *E2E Service Provisioning* to the *E2E Service Ingestion.*
- **Step 20.** *E2E Service Ingestion* informs the ***user or the operator*** about the success of the Service Provisioning Request.

In the following, we detail the first loop, i.e., the operations happening in local domains when the E2E Path Computation requests a local deterministic path.

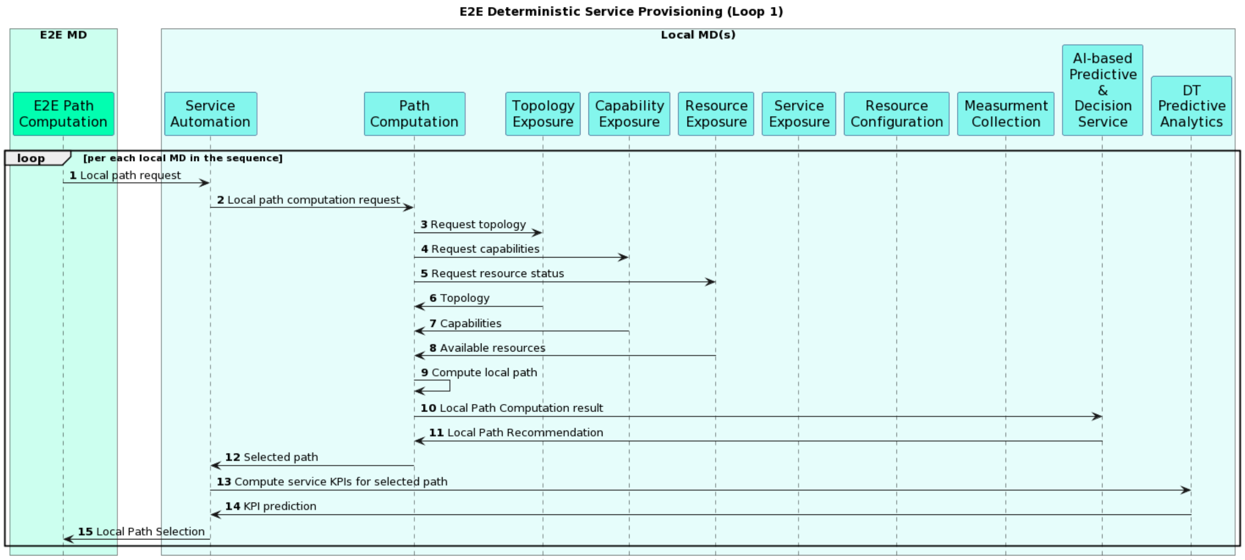

Figure 9-2 E2E Deterministic Service provisioning – Loop 1, Local Management Domain view

- **Steps 1** and **2.** *Service Automation MS* forwards the local path computation request to the local *Path Computation MS.*
- **Steps 3, 4** and **5.** *Path Computation MS* in each domain asks *Topology Exposure*, *Capability Exposure*, and *Resource Exposure* to retrieve information from domains.
- **Steps 6, 7,** and **8.** *Topology Exposure, Capability Exposure,* and *Resource Exposure* get the corresponding information from the data plane, abstract it, and send it back to *Path Computation.*
- **Steps 9, 10,** and **11.** *Path Computation* performs the path calculation for each domain and forwards this calculation to the *Learning Orchestrator* to get a recommendation for the best path alternative to provision the service in the specific local domain.
- **Step 12.** *Path Computation* forwards the selected path to the *Service Automation*
- **Steps 13** and **14.** Under request from *Service Automation,* the *DT Predictive Analytics* MS composes a dedicated scenario with the sub-topology defined for the route to evaluate, selects the already provisioned services that are supported by these resources and runs simulations to estimate the KPIs based on the definition of traffic for the new service and the traffic models for the current services. The estimation results for the local domain are returned.
- **Step 14.** *Service Automation* forwards the local path selection to the E2E Path Computation.

Below is detailed the second loop characterizing the provisioning of an E2E Deterministic service, i.e., the set of operations required to provisioning a local deterministic sub-service.

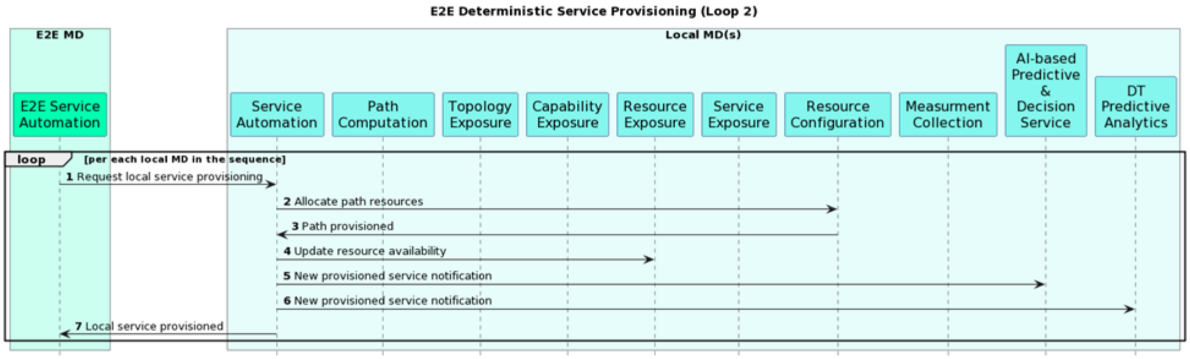

Figure 9-3 *E2E Deterministic Service provisioning - Loop 2, Local Management Domain view*

- **Step 1.** *E2E Service Automation* requests the service provisioning in the selected local domain through an interaction with the *Service Automation MS.*
- **Step 2.** *Service Automation MS* performs a request for resource configuration to the *Resource Configuration MS.*
- **Step 3.** *Resource Configuration MS* receives the request and configures resources directly on the data plane and provisions the path to the *Service Automation MS.*
- **Step 4.** *Service Automation MS* updates resource availability at the *Resource Exposure MS.*
- **Steps 5** and **6.** *Service Automation MS* informs the *Learning Orchestrator* about the service provisioning.
- **Step 7.** *Service Automation MS* informs the *E2E Service Automation* about the service provisioning in the local domain.

9.2 E2E Deterministic Service Decommissioning

The termination of a deterministic service can happen under two potential situations: (1) explicit termination request from the user/operator, or (2) due to the expiration of the service lifetime, which is a subset of the steps performed by (1). As in previous workflow the AI/ML MSs are generically indicated as AI/ML AI-based & Predictive Decision Service (E2E and not E2E) in a single box per domain, in order to ease the readability of the figures.

(1) Explicit Termination Request

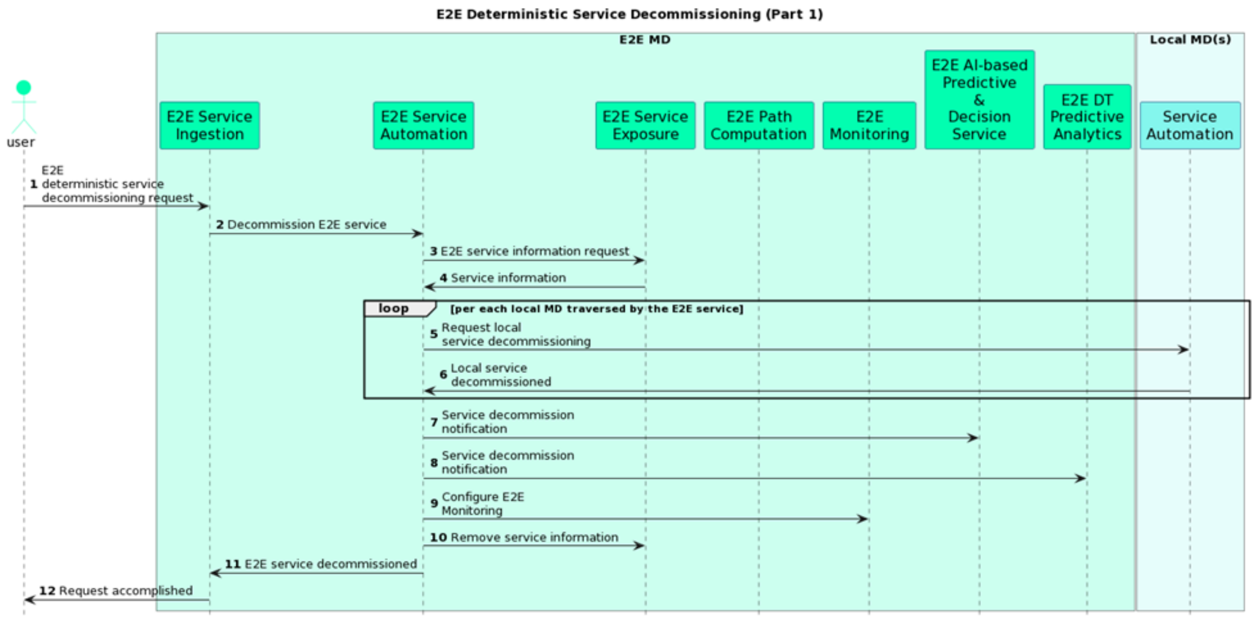

Figure 9-4 E2E Deterministic Service decommissioning - Part 1, E2E Management Domain view

- **Step 1.** *User s*ends Service Decommissioning Request to *E2E Service Ingestion.*
- **Step 2.** *E2E Service Ingestion* forwards the request to *E2E Service Automation* where E2E service decommissioning is initiated.
- **Step 3.** *E2E Service Automation* retrieves additional service information from the *E2E Service Exposure* to perform the decommissioning.
- **Step 4.** *E2E Service Exposure* returns additional service information to the *E2E Service Automation.*
- **Step 5.** *E2E Service Automation* informs the *Service Automation MS* of the specific technology-domain about the service decommissioning and starts an iteration in the local MD.
- **Step 6.** *E2E Service Automation* is informed about the local service decommissioning and turns it into an E2E service decommissioning.
- **Step 7.** *E2E Service Automation* forwards the E2E Service Decommissioning notification to the *E2E Learning Orchestrator.*
- **Step 8.** *E2E Service Automation* forwards the E2E Service Decommissioning notification to the *DT Predictive Analytics.*
- **Step 9.** *E2E Service Automation* configures E2E monitoring for stopping the monitoring of the parameters related to the decommissioned service.

- **Step 10.** *E2E Service Automation* removes the E2E service information from the *E2E Service Exposure MS.*
- **Step 11.** *E2E Service Automation* informs the *E2E Service Ingestion* about the E2E Service Decommissioning.
- **Step 12.** *E2E Service Ingestion* notifies the user or operator about the success of the E2E Service Decommissioning request.

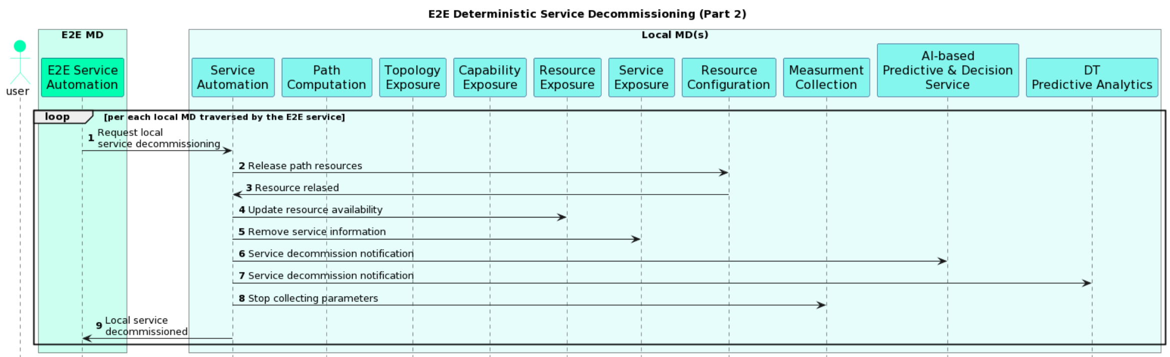

Figure 9-5 E2E Deterministic Service decommissioning - Part 2, Local Management Domain view

- **Step 1.** *Service Automation MS* receives the local service decommissioning request.
- **Step 2.** *Service Automation MS* requests to the *Resource Configuration MS* the release of the resources allocated to the specific service in the local domain and the specific path.
- **Step 3.** *Resource Configuration MS* releases the corresponding resources.
- **Step 4.** *Service Automation MS* informs the *Resource Exposure MS* about the update on the resource availability.
- **Step 5.** *Service Automation MS* removes service information from the *Service Exposure*.
- **Steps 6 and 7. Service Automation MS** informs about the service decommissioning to the *Learning Orchestrator* and the *DT Predictive Analytics MSs*.
- **Step 8.** *Service Automation MS* requests to stop collecting monitoring parameters to the *Measurement Collection MS*.
- **Step 9.** *Service Automation MS* informs the *E2E Service Automation MS* about the local service decommissioning.

<u>(2) Expiration of the Service Lifetime</u>

In this case, the E2E Service Automation automatically detects the service expiration and trigger its termination. The workflow coincides with (1) starting from step 3.

9.3 AI/ML Service Setup

The workflow in **Figure 9-6** illustrates the interaction that takes place between the system entities whenever an AI/ML service, e.g., an ML-based decision-making process, is requested by the network orchestrator to configure an E2E network service.

The E2E Learning Orchestrator receiving the request identifies the per-domain Learning Orchestrator(s) to be involved and relays the request to such entity(ies) (**Step 1**). It is thus responsible for the local setup of the AI/ML Management Service. It first queries the ML Model Registry (**Steps 2-3**) to retrieve the models metadata and select the right model for the AI/ML operation to execute, and, similarly, the Dataset Registry (**Steps 4-5**) whenever a dataset is necessary for model (pre-)training. It then gets information from the ML Resource Orchestrator (**Steps 6-7**) about the available network and computing resources offered by the nodes in the domain.

Using the above pieces of information, the Learning Orchestrator can run the appropriate algorithm (**Step 8**) and determine the most efficient and effective configuration for executing the AI/ML task (e.g., inference task, or ML model training and inference). Such configuration is then notified to the ML Resource Orchestrator (**Step 9**), which, through the help of the Learning

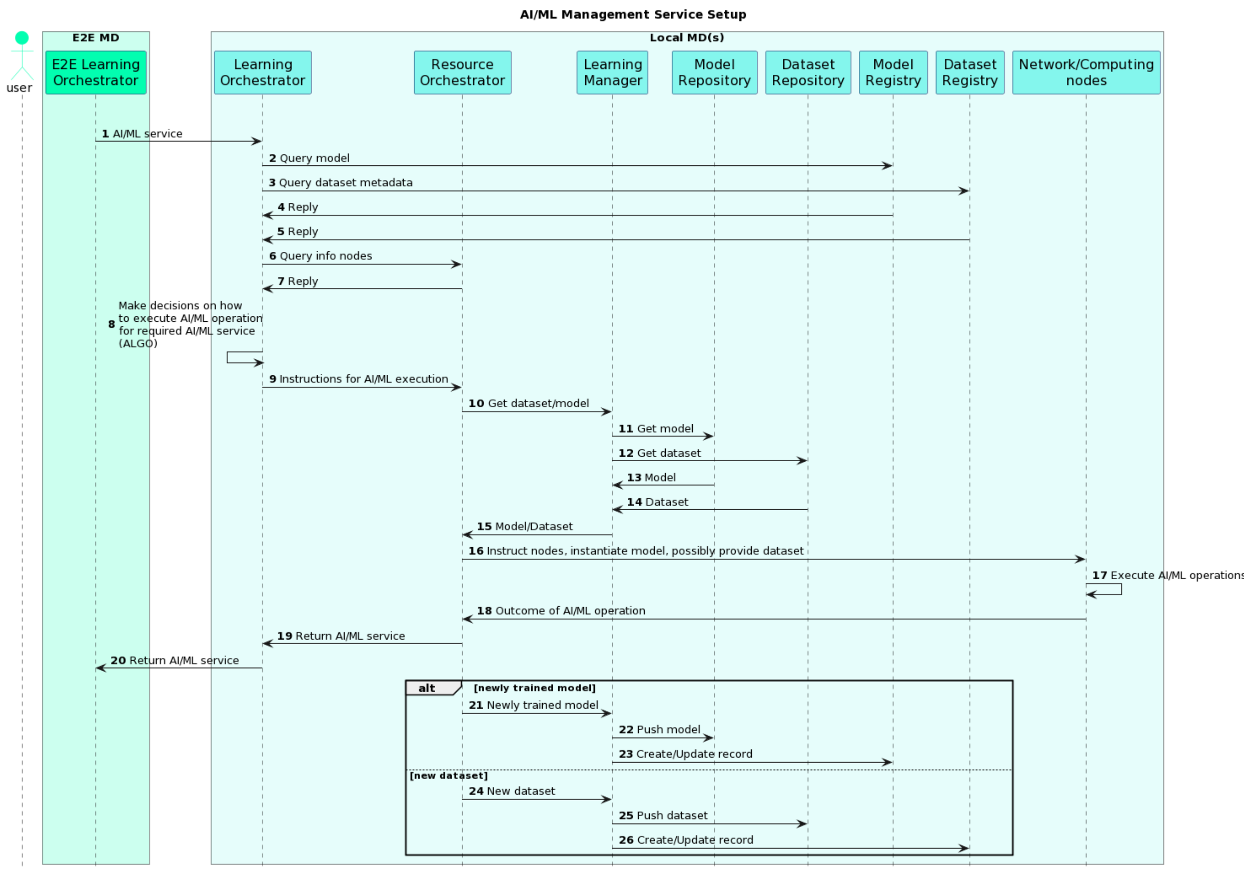

Figure 9-6 AI/ML Management Service setup

Manager retrieves the selected model (and dataset) from the Model (and the Dataset) Repository (**Steps 10-15**). The ML Resource Orchestrator then instructs the nodes to be involved in the AI/ML operational tasks (**Step 16**), providing them with the model (and dataset). Once the nodes execute the task (**Step 17**), they return the outcome to the ML Resource Orchestrator (**Step 18**), which further relays it to the Learning Orchestrator (**Step 19**). On its turn, the latter forwards the outcome to the E2E Learning Orchestrator (**Step 20**), for the subsequent delivery to the Network Orchestrator.

Additionally, whenever the AI/ML task outcome consists or implies the training/update of a model, or the creation/update of a dataset, the ML Resource Orchestrator also sends the new model/dataset to the Learning Manager (**Steps 21** and **24**), which accordingly pushes the model/dataset into the corresponding Repository (**Steps 22** and **25**) and the associated metadata into the corresponding Registry (**Steps 23** and **26**).

10 Conclusions

This document discussed the architecture of AICP for PREDICT-6G, a control plane, AI-driven, in charge of implement E2E determinism across heterogenous domains (administrative and technological), supporting different degrees of service time-sensitiveness. Design decisions have been widely described, especially the followed service-centric pattern, with the concept of MSs, sitting in different managed domains, technical and E2E. The different MSs have been grouped per logical categories, widely analysed in the document and representing the key enablers of AICP: i) AI/ML for smart resource allocation, ii) DTs for system and (deterministic) service' KPI predictably, iii) Data Collection and Management for continuous monitoring of the services and iv) Inter-domain orchestration for the management of E2E deterministic service lifecycle.

A description with functionalities has been provided per each MS, along with a set of possible interactions with other MSs in both hierarchical and local manner. A meaningful part of the document has been also dedicated to the mapping of the AICP functionalities on the control features provided by the different technological domains targeted, which also includes the description on how to abstract such technologies (network modelling) and related data (data modelling). Finally, the descriptions of the various MSs interactions have also been reinforced with additional graphical and textual examples of operation workflows.

The work presented in this document should be considered preliminary, an initial outcome of WP3 effort, aiming to fix base cornerstones for next implementations of the different services (an initial implementation will be reported in D3.2 due by project M12). This means that while the service-centric approach is not expected to change in the future, the number and the features described so far for the MSs might change, considering the feedback received during the initial prototyping, the test and integration activities (WP4) and interaction with PREDICT-6G data plane (MDP, WP2).

11 References

Wen Wu et al. (2022) “AI-Native Network Slicing for 6G Networks” IEEE Wireless Communications, Available at: AI-Native Network Slicing for 6G Networks | IEEE Journals & Magazine | IEEE Xplore

(RFC6241, 2011) IETF RFC6241, "Network Configuration Protocol (NETCONF)", 2011, Available at: https://datatracker.ietf.org/doc/html/rfc6241

(RFC7950, 2016) IETF RFC7950, "The YANG 1.1 Data Modelling Language", 2016, Available at: https://datatracker.ietf.org/doc/html/rfc7950

(IEEE 1588, 2008) IEEE 1588-2008, "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", Available at: https://standards.ieee.org/ieee/1588/4355/

(IEEE 802.1AS-2020, 2020) IEEE 802.1AS-2020,"IEEE Standard for Local and Metropolitan Area Networks--Timing and Synchronization for Time-Sensitive Applications", Available at: https://standards.ieee.org/ieee/802.1AS/7121/

(IEEE 802.1Qbv-2015, 2015) IEEE 802.1Qbv-2015, "IEEE Standard for Local and metropolitan area networks -- Bridges and Bridged Networks - Amendment 25: Enhancements for Scheduled Traffic", Available at: https://standards.ieee.org/ieee/802.1Qbv/6068/

(IEEE 802.1CB-2017, 2017) IEEE 802.1CB-2017, "IEEE Standard for Local and metropolitan area networks--Frame Replication and Elimination for Reliability", Available at: https://standards.ieee.org/ieee/802.1CB/5703/

(PREDICT-6G/D1.1/9, 2023) PREDICT-6G D1.1 "Analysis of use cases and system requirements.”, Section 9, Jun. 2023.

(PREDICT-6G/D2.1/3, 2023) PREDICT-6G D2.1 "Release 1 of PREDICT-6G MDP innovations", Section 3, Aug. 2023

(PREDICT-6G/D2.1/4.3, 2023) PREDICT-6G D2.1 "Release 1 of PREDICT-6G MDP innovations", Section 4.3, Aug. 2023

(PREDICT-6G/D2.1/4.2, 2023) PREDICT-6G D2.1 "Release 1 of PREDICT-6G MDP innovations", Section 4.2, Aug. 2023

(PREDICT-6G/D1.1/A, 2023) PREDICT-6G D1.1 "Analysis of use cases and system requirements.”, Sec. 15.1 Appendix A, Jun. 2023.u

Funded by the European Union