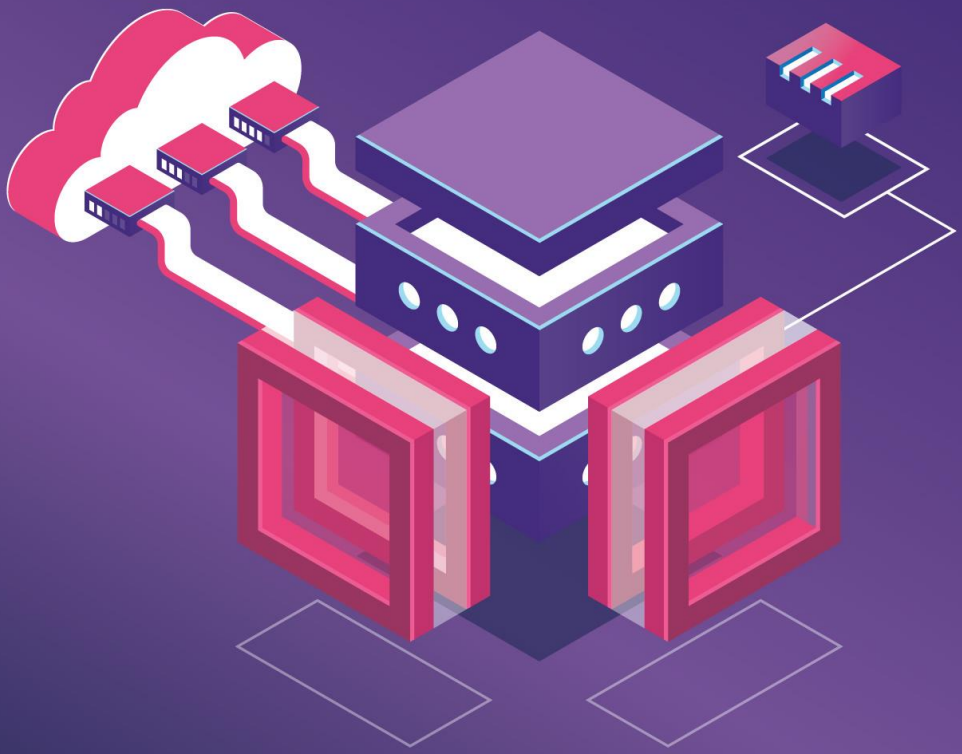
EUCloudEdgeIoT.eu
Open Continuum

Functional view of the Continuum Reference Architecture

Minimum set of expected functionalities

Authors

This work was performed by Task Force 3:
Architecture leader, part of the OpenContinuum project consortia, within the EUCloudEdge initiative.



Grant Agreement No.: 101070030 **Topic:** HORIZON-CL4-2021-DATA-01-07
Call: HORIZON-CL4-2021-DATA-01 **Type of action:** HORIZON-CSA

CONTENTS

Funded by Horizon Europe Framework Programme of the European Union

1. INTRODUCTION

This is the second document of a series of three defining a reference architecture for the continuum. In the previous one, different building blocks were identified setting the path for the different aspects to be taken into account while defining the architecture. The next step is, then, to develop a functional overview of this reference architecture from this big picture to a more complete one providing the basic information to understand the interconnections between the abovementioned building blocks. Functional architectures are needed to properly understand how a system acts and interacts. This is even more relevant within the continuum, as the gap between IoT and edge/cloud must be covered. In this sense, the work already performed aims to go one step further of other available architectures tackling one or two pillars of the continuum, by integrating the three of them totally agnostic of the application domain, that is, not influenced by any vertical or potential use case but flexible enough to address most of their needs.
This paper focuses on the minimum set of functionalities needed for the continuum lifecycle management as well as the relationships between them, providing different diagrams representing each of the initially identified building blocks and their interrelationships, with the main goal of developing the homogenised version of the Reference Architecture for the continuum.

ABOUT THE AUTHORS

This work was performed by Task Force 3: Architecture leader, part of the OpenContinuum project consortia, within the EUCloudEdge initiative.
Task Force 3 aims to provide a common vision on Continuum computing, providing a homogenised language and a reference architecture to set the basis for a European standard, positioning Europe ahead of the competition. The work presented here has been further validated with 30 ongoing research projects in the fields of edge, cloud and IoT (mainly ICT-40-2020, ICT-50-2020, ICT-56-2020, HORIZON-CL4-2021-DATA-01-05, HORIZON-CL4-2022-DATA-01-02, HORIZON-CL4-2022-DATA-01-03, HORIZON-CL4-2022-DIGITAL-EMERGING-01-26 and HORIZON-CL4-2023-DATA-01-04) and will be extended in the coming months until the reference architecture is fully developed and mapped with all projects' solution architectures.

Funded by Horizon Europe Framework Programme of the European Union

2. BRIDGING THE GAP FROM IOT TO EDGE AND CLOUD

After a quick overview at the current state of the art of reference architectures, it can be highlighted that there are many standardized (officially or de facto) ones mainly applying to cloud or Internet of Things (IoT). However, there is no homogenization while integrating cloud, edge and IoT into a single continuum. Thus, research projects focused on identifying those common functionalities that are needed in this environment. Considering, at the same time, different perspectives according to their knowledge, as, e.g., Function as a Service (FaaS), cognitive or swarm computing requirements.
The work presented in the following subsections integrates these requirements into one single functional view. However, additional ones can be added on top of them, as the main goal of the architecture is to be flexible and usable according to each implementation needs.
In order to deeply analyze the requirements and identify the minimum set of functionalities, research projects clustered into working groups (WG) according to their experience.

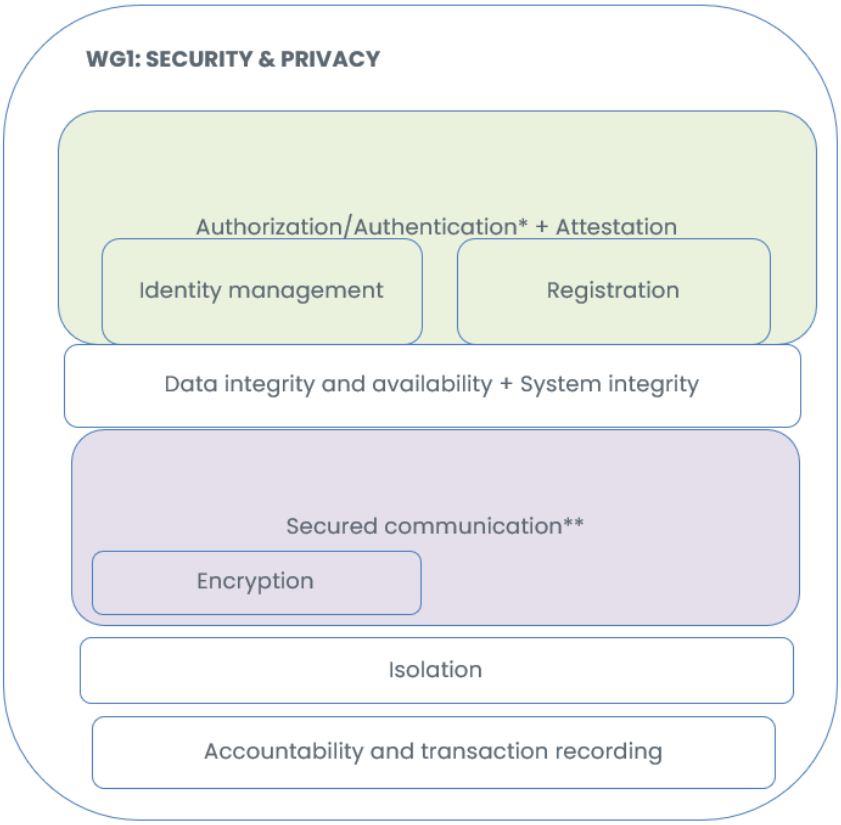
ABBREVIATIONS

AI	Artificial Intelligence
FaaS	Function as a Service
IoT	Internet of Things
LM	Lifecycle Management
ML	Machine Learning
QoE	Quality of Experience
QoS	Quality of Service
SLA	Service Level Agreement
SLO	Service Level Object
TEE	Trusted Execution Environment
UC	Use Case
WG	Working Group

Funded by Horizon Europe Framework Programme of the European Union

3. SECURITY AND PRIVACY

Security is a transversal challenge to be addressed while operating data across the continuum. The main goal is to properly identify all security considerations that must be considered. At the same time, privacy is a mandatory requirement to address needs and requirements from the legislative framework. So, complementing the main goal, a set of needed functionalities has been also identified to address privacy requirements.



** Needed for all building blocks*

*** Mandatory to ensure security between components*

The figure shows the functionalities needed to ensure proper security and privacy management within the architecture, including those that must be included in the definition of functionalities from other building blocks.

Based on this, the following list defines each of the identified functionalities:

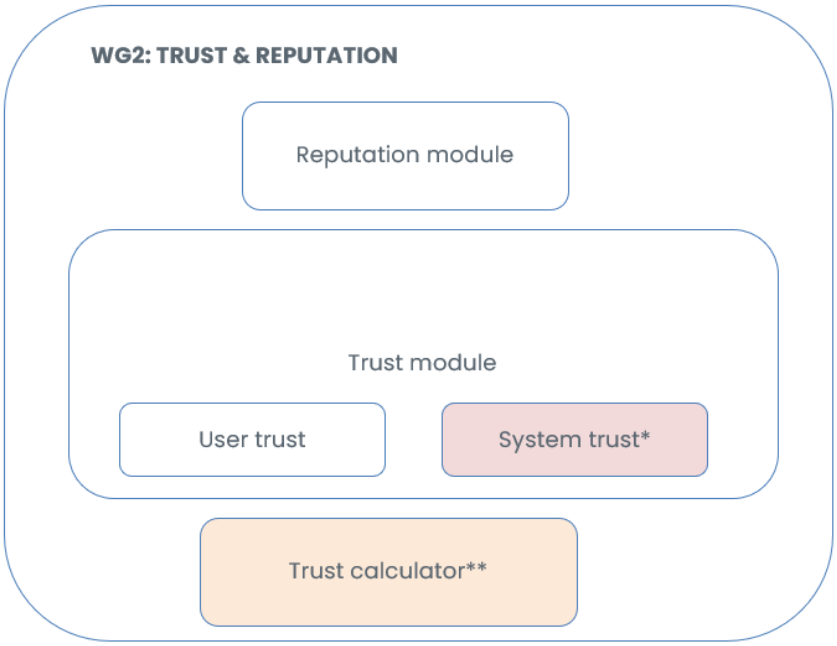
- **Authorization/Authentication + Attestation:** in charge of allowing the access to the system to all users according to a predefined role, plus ensuring all agents are allowed to be part of the system. This information must be spread to all building blocks:
    - **Registry:** catalogue of users and resources and their role within the system.

Funded by Horizon Europe Framework Programme of the European Union

    - **Identity (and access) management:** to ensure only allowed users can access the system.
- **Data integrity and availability + System integrity:** ensuring all data is trustable, accessible and coherent. As well as for the system. This includes a dedicated chapter for safety (to be implemented under demand) for specific use cases.
- **Secured communication:** Needed to ensure the security and safeness of all communications between the different components (needed for all building blocks).
    - **Encryption:** of data transmitted during communication.
- **Isolation:** to minimize threats based on, e.g., Trusted Execution Environments (TEEs) on the different hardware architectures (Intel SGX, ARM Trustzone, etc.).
- **Accountability and transaction recording:** only for those implementations that involve any economic transaction (blockchain-related).

4. TRUST AND REPUTATION

Beyond security, it is needed to implement the necessary mechanisms to measure the trust and reputation levels of the users, providers and any additional source.



**Needed for building block 1 (Security & Privacy)*
***Distributed among nodes*

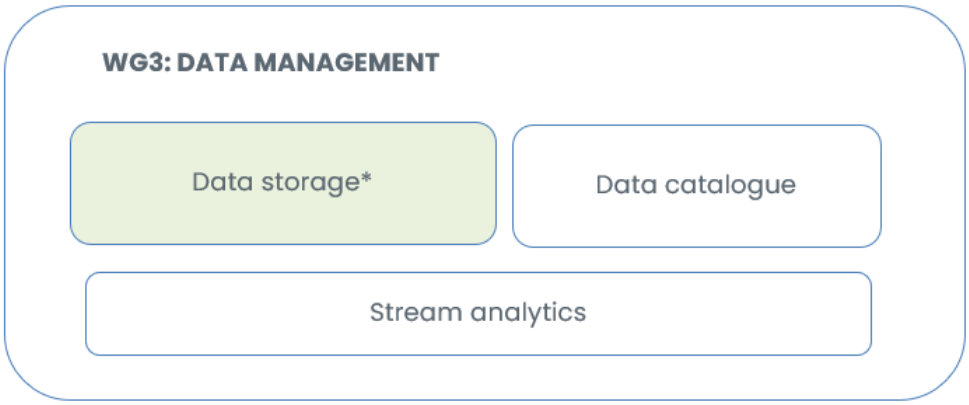
- **Trust calculator:** distributed among nodes to collect specific information of a given node. This information will be shared with the Trust module or exposed to the user if needed.

Funded by Horizon Europe Framework Programme of the European Union

- **Trust module:** in charge of calculating the trust level of the system or the users and sharing this information with the user.
    - **User trust:** measures the trust level of a user based on his/her actions.
    - **System trust:** measures the trust level of the system. Initially based on the available information from the provider, to be updated later on with the results from the trust calculator. This information is also shared with building block 1 (Security & Privacy) with regards to the system trustworthiness to be included as an additional security consideration within the system integrity evaluation.
- **Reputation:** measures the reputation level of a given provider and makes suggestions to the user in order to select one or another for his/her deployment. More related to Quality of Experience (QoE).

5. DATA MANAGEMENT

Data management is basic, not only for application analytics, but also for taking informed decisions, e.g., the best location to perform an action. Thus, it is important to properly define the data management at system and application level.



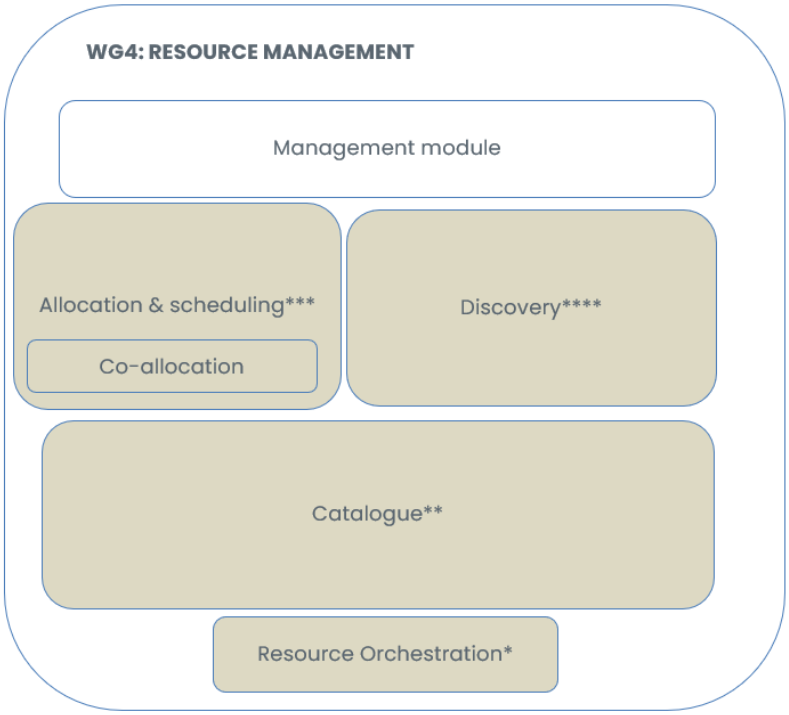
**Needed for all building blocks*

- **Data storage:** common to all building blocks and in different formats.
- **Data catalogue:** registers all data stored within the system so it is available for different components.
- **Stream analytics:** performed after request.

Funded by Horizon Europe Framework Programme of the European Union

6. RESOURCE MANAGEMENT

Infrastructure management is important in any cloud/edge architecture, however, when incorporating IoT the number of devices to be managed grows exponentially. So, the main goal is identifying the functionalities needed to properly manage all of them.



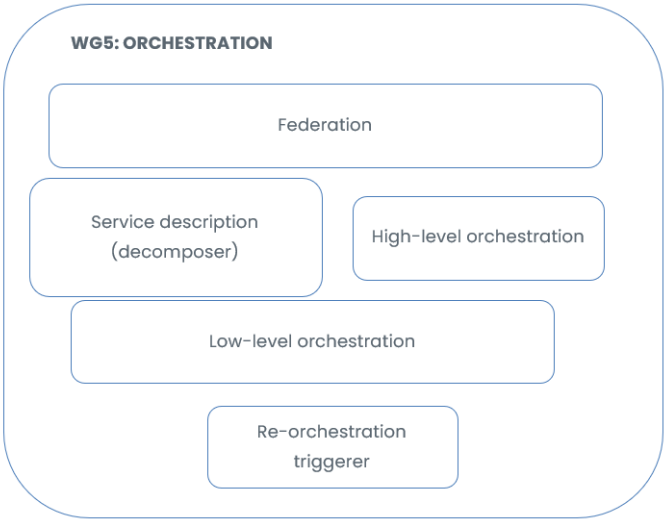
**Linked to low level orchestration (building block 5 - Orchestration)*
***Linked to storage and analytics (building block 3 – Data management)*
****Linked to scheduling (building block 8 – Artificial Intelligence) and federation (building block 5 - Orchestration)*
*****Linked to scheduling (building block 8 – Artificial Intelligence) and monitoring (building block 7 – Monitoring & Observability)*

- **Management module:** smart decision maker for resource management.
- **Allocation & scheduling:** collects information about resource availability and sets up the cluster of needed resources.
    - **Co-allocation:** allocate resources from different providers.
- **Discovery:** looks for those resources which fulfil the requirements established by the user or the application.
- **Catalogue:** contains the information about all the resources registered within the system.
- **Resource orchestration:** provides the information about the resource deployment scheme.

Funded by Horizon Europe Framework Programme of the European Union

7. ORCHESTRATION

The core of the architecture relies on the orchestration of services, applications or even functions, as well as network resources or devices. Thus, it is important to identify how to properly perform the orchestration of services and applications, and the integration with other related blocks in charge or orchestrating different resources (devices, network or artificial intelligence models, among others).



- **Federation:** manages cloud/edge, combined with the resources' information, to ensure the availability of all of them for a given deployment.
- **Service description:** contains the requirements of a given application about deployment needs. This includes day-0 deployment Service Level Agreement (SLA), requirements, and other specific policies as well as service workload specification. This global flow should be understood two-fold: (i) autonomous way, (ii) exposing the information so that it could be leveraged by other components. In the autonomous way, this can be conceived as well in two ways: self-contained, expressed by the user, and the other one is that the application generates dynamically its own requirements. The service description (composer) shall also include the "topology" concerns (e.g., network, ports exposure, relation to other microservices, resources needed - in terms of spot in the continuum, etc.).
- **High-level orchestration:** collects the scheduling recommendations to develop the deployment scheme and sends this information to the low-level orchestrator. In case of rescheduling, based on predictive maintenance, it capitalizes the action again.
- **Low-level orchestration:** orchestrates network, resources and services as needed. In case something is not properly working and needs immediate action, it takes the lead.
- **Re-orchestration triggerer.** A variety of reasons (e.g., overload of resources, predictive maintenance alerts, certain events) that will trigger a re-orchestration request. Also, the reorchestration can happen from a logical point of view, not necessarily coming from specific events.

Funded by Horizon Europe Framework Programme of the European Union

NOTE: FaaS special case not as same level as Low-level orchestration

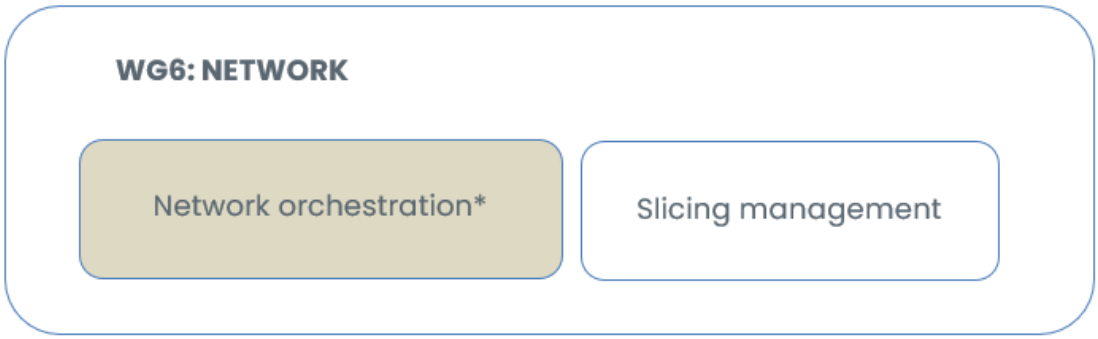
- **FaaS scheduling:** Part of the low-level orchestration only for specific UCs where functions are involved.
- It might be omitted as an extra box

Scheduling FaaS*

**Specific for FaaS use cases (UCs)*

8. NETWORK

In the transition from cloud/edge to IoT, network constraints are even more relevant.



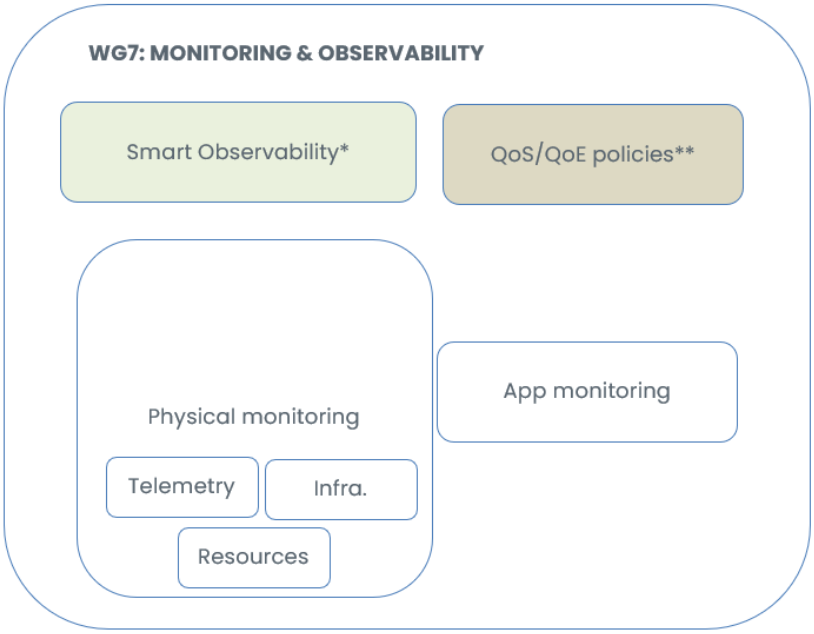
**Linked to low level orchestration (building block 5 - Orchestration)*

- **Network orchestration:** whenever it is possible (relies on the infrastructure setup).
- **Network slicing:** additional to the orchestration and performed under request.

MONITORING & OBSERVABILITY

QoS/QoE are beyond the proper functioning of the system, so it is necessary to monitor the system health and provide the needed information to other blocks.

Funded by Horizon Europe Framework Programme of the European Union



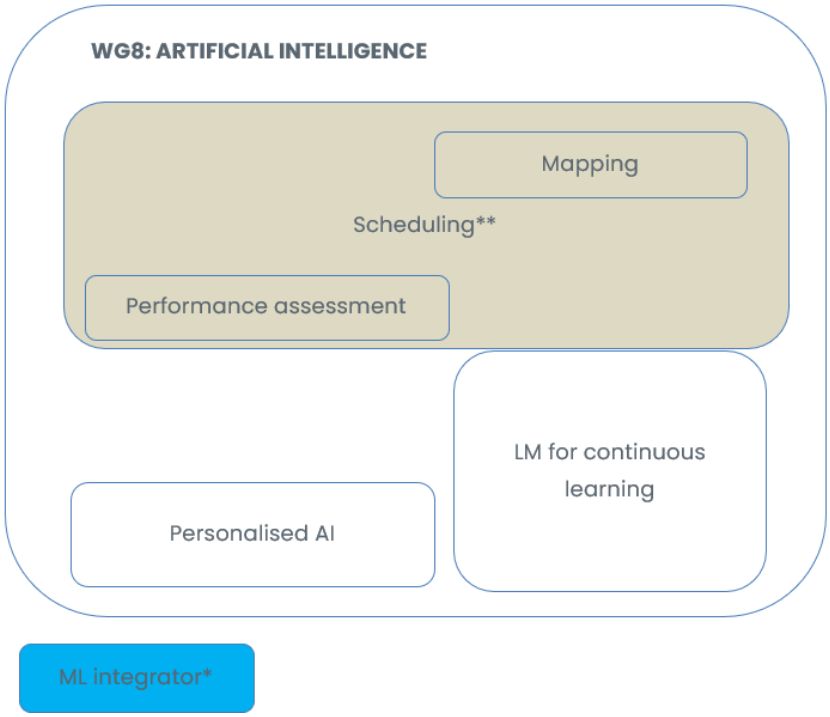
**Linked to storage and analytics (building block 3 – Data management). Data exposure to all WGs. And scheduling (building block 8 – Artificial Intelligence) for predictive maintenance.*
***Linked to storage (building block 3 – Data management)*

- **Smart Observability:** collects monitoring information and provides the first analysis together with the Service Level Objects (SLOs) previously defined to inform about any potential event. It establishes different levels of alerts to make the distinction between predictive maintenance and immediate actions.
- **Quality of Service (QoS)/QoE policies:** catalogue of SLOs.
- **Physical monitoring:** in charge of collecting data from the physical layer.
  - **Telemetry monitoring:** monitors the network health.
  - **Infra monitoring:** monitors cloud/edge infrastructure.
  - **Resource monitoring:** monitors IoT devices.
- **App monitoring:** in charge of collecting data about the application status.

9. ARTIFICIAL INTELLIGENCE

Artificial Intelligence is a transversal topic to all blocks, as it can be incorporated as a functionality in any of them. Additionally, it also focuses on the execution and deployment of different models.

Funded by Horizon Europe Framework Programme of the European Union



**Challenge for future research*
***Linked to high level orchestration and federation (building block 5 - Orchestration) and resource management (building block 4)*

- **Scheduling:** analyses the available information and provides recommendations to support high-level orchestration decision making.
  - **Mapping:** maps the service, or the Artificial Intelligence (AI) algorithm, requirements with the most appropriate physical device for deployment.
  - **Performance assessment:** of an already deployed service or AI algorithm. This information is used for predictive maintenance or directly communicated to the user.
- **Personalised AI:** for specific cases where the user demands dedicated models to be used.
- **Lifecycle Management (LM) for continuous learning:** lifecycle management to allow the involved AI/Machine Learning (ML) to learn during a non-defined time period.
- **ML integrator:** nice to have but not currently supported, in order to allow the combination of different ML algorithms to develop new ones.

10. CONCLUSION AND NEXT STEPS

As already presented in previous document of the series, there are three major steps for developing a reference architecture:

1. Identify those terms that are common in the main pillars of the continuum and provide a homogenized definition to ensure all relevant stakeholders are using the same concepts.

Funded by Horizon Europe Framework Programme of the European Union

2. Identify the common building blocks and the minimum set of functionalities to provide a simplified and easily understandable version of the architecture.
3. Design a reference architecture covering all aspects of the continuum.

This document focused on the second step, identifying the minimum set of functionalities and the relationship between the different building blocks to develop a common view of the reference architecture.
This work set the basis for the final step in order to develop a useful architecture that can be reused and further extended in the future, as long as technology evolves. Furthermore, all projects participating in its design are also working in developing a set of research challenges with those aspects that cannot be covered right now but must be addressed in the coming future to keep European research ahead of competition.
Additional documents are under development, providing information about specific results that can be of interest for the implementation of a Reference Architecture for the continuum.

11. ABOUT EUCLOUDEDGEIOT INITIATIVE

The **EUCloudEdgeIoT.eu** initiative aims to realise a pathway for the understanding and development of the Cloud, Edge and IoT (CEI) Continuum by promoting cooperation between a wide range of research projects, developers and suppliers, business users and potential adopters of this new technological paradigm.
The EUCloudEdgeIoT initiative acts as an enabling force, to reach key outcomes:

- Support the definition of the large-scale pilots envisaged by the European Commission in line with the EU Data Strategy.
- Baseline common open architecture for computing continuum research projects.
- Reinforce the collaboration between European public and private initiatives from cloud to edge to IoT.
- Increase the awareness of the importance of open source and standards for EU digital autonomy.

Within this initiative, TF3 Architecture main goals are as follows:

- Enable the architectural discussions among projects in the area of IoT/Edge and Cloud to create a continuum.
- Identification of the thematic areas and building blocks.

Understanding the contribution of each project to the thematic areas, allowing the identification of cross-project synergies.

Funded by Horizon Europe Framework Programme of the European Union

EUCloudEdgeIoT.eu
Open Continuum

www.eucloudedgeiot.eu

@EU_CloudEdgeIoT

eucloudedgeiot

Grant Agreement No.: 101070030 **Topic:** HORIZON-CL4-2021-DATA-01-07
Call: HORIZON-CL4-2021-DATA-01 **Type of action:** HORIZON-CSA