HEXA-X-II

A holistic flagship towards the 6G network platform and system, to inspire digital transformation, for the world to act together in meeting needs in society and ecosystems with novel 6G services

Deliverable D2.6
Final end-to-end system evaluation results of the overall 6G system design

Co-funded by the European Union

6GSNS

Hexa-X-II project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101095759.

Date of delivery:	30/06/2025	Version:	1.0
Project reference:	101095759	Call:	HORIZON-JU-SNS-2022
Start date of project:	01/01/2023	Duration:	30 months

Document properties:

Document Number:	D2.6
Document Title:	Final end-to-end system evaluation results of the overall 6G system design.
Editor(s):	Sokratis Barmpounakis (WIN), Vasiliki Lamprousi (WIN), Vasilis Tsekenis (WIN), Panagiotis Demestichas (WIN)
Authors:	Sylvaine Kerboeuf (NFR), Akshay Jain (NFI), Carl Collmann (TUD), Ricard Vilalta, Pol Alemany, Raul Muñoz, Daniel Adanza, Lluís Gifre, Behnam Ojaghi (CTTC), Riccardo Nicolicchia (TID), Pawani Porambage, Rafael Pires (VTT), Pietro G. Giardina (NXW), Erin Elizabeth Seder (NXW), Pietro Piscione (NXW), Ioannis Tzanettis (ICC), Grigoris Kakkavas (ICC), Anastasios Zafeiropoulos (ICC), Selim Ickin (EAB), Luís Santos (UBW), Yann Lebrun (QLC), Taesang Yoo (QLC), Mohammad Hossein Moghaddam (QRT), Dani Korpi (NFI), Ahmad Nimr (TUD)
Contractual Date of Delivery:	30/06/2025
Dissemination level:	PU
Status:	GA approval phase document
Version:	1.0
File Name:	Hexa-X-II_D2_6_FINAL.docx

Revision History

Revision	Date	Issued by	Description
0.1	01.10.2024	Hexa-X-II WP2	Complete first draft towards internal review phase
0.2	28.03.2025	Hexa-X-II WP2	Internal review phase document version
0.3	30.04.2025	Hexa-X-II WP2	External review phase document version
1.0	30.05.2025	Hexa-X-II WP2	GA approval phase document version
1.0	30.06.2025	Hexa-X-II WP2	Submission to EC

Abstract

This deliverable presents the final evaluation results of the Hexa-X-II end-to-end (E2E) 6G system design, building upon the foundational concepts, enablers, and intermediate validations established in previous deliverables. It focuses on the experimental validation and assessment of the third and final system and component proof-of-concept (PoC #C) framework, comprising advanced 6G radio and device components, AI-driven orchestration mechanisms, flexible topologies, and intent-based management across the compute-communication continuum.

The document provides a comprehensive mapping between system enablers and the final 6G system blueprint, followed by detailed validation results for system-level performance and sustainability. Key innovations include the integration of energy-harvesting proximity sensors, AI-native air interfaces, zero-energy crowd-detectable devices, and advanced joint communication and sensing capabilities. The evaluation also covers latency-sensitive and collaborative robotics scenarios, highlighting system

intelligence, resilience, and dynamic adaptability through cognitive closed-loop automation and trust-aware resource allocation.

Extended validation results are provided through real-world testbeds and simulation-based studies, demonstrating the feasibility and maturity of the Hexa-X-II 6G architecture. The outcomes contribute critical insights towards the standardization and deployment of 6G systems, addressing key performance indicators (KPIs), key value indicators (KVIs), and reinforcing the project's objectives of enabling sustainable, trustworthy, and intelligent ecosystems.

Keywords

6G, end-to-end system, system PoC, component PoC, collaborative robotics, intent-based networking, closed-loop automation, AI-native air interface, exposure APIs, zero-energy devices, energy harvesting, flexible topologies, integration fabric, trust management, multi-stakeholder orchestration, joint communication and sensing, application enablement platform, cloud continuum, network programmability, sustainability, resilience, KPI - KVI evaluation

Executive Summary

This deliverable is the final outcome of Work Package 2 (WP2) of the Hexa-X-II project, presenting a comprehensive evaluation of the full end-to-end (E2E) 6G system architecture developed throughout the project. It builds upon the design principles, 6G E2E system blueprint, and enabling technologies detailed in earlier WP2 deliverables (D2.1–D2.5) and focuses on validating their integration and performance within a unified platform, through System Proof-of-Concept (PoC) #C and associated Component PoCs.

The document provides a mapping between the evaluated components and the Hexa-X-II 6G E2E system blueprint. It showcases how key architectural innovations—such as AI-native orchestration, energy-harvesting and zero-energy devices, joint communication and sensing (JCAS), trust-aware resource allocation, and intent-based service management—have been practically implemented and tested. Two representative use cases are used for this evaluation: Warehouse Inventory Management and Video Surveillance with Cobots, both reflecting real-time, latency-sensitive, and resource-constrained service requirements.

System-PoC #C demonstrates advanced orchestration across multiple domains, combining closed-loop automation, trust assessment, service migration based on energy- or trust-related parameters, and exposure of 6G capabilities via CAMARA-compliant APIs. These scenarios are supported by the Management Capability Exposure (MCE) Framework, enabling dynamic, policy-driven adaptation and collaborative management across cloud, edge, and device layers.

The deliverable includes both real-world testbed validations and evaluations, addressing Key Performance Indicators (KPIs) such as latency, reliability, energy consumption, orchestration efficiency, and system responsiveness. It also aligns outcomes with Key Value Indicators (KVIs) such as trustworthiness, and inclusion. Results confirm the system's ability to operate intelligently and securely across diverse stakeholder environments while supporting future 6G services like XR, federated learning, and robotic collaboration.

In conclusion, this deliverable completes WP2 by transitioning from architectural design to validated implementation. It confirms the feasibility of Hexa-X-II’s 6G vision and provides a solid foundation for standardization, future research, and industrial adoption. The outcomes strongly contribute to the SNS JU objectives, reinforcing Europe’s leadership in the development of a sustainable, and intelligent 6G ecosystem.

Table of Contents

List of Tables

List of Figures

Acronyms and abbreviations

Term	Description
AI	Artificial Intelligence
AIaaS	AI-as-a-Service
AMR	Autonomous Mobile Robot
CaaS	Compute-as-a-Service
CIR	Channel Impulse Response
CN	Core Network
CNN	Convolutional neural Networks
Cobot	Collaborative Robot
CP	Control Plane
CU	Central Unit
DLT	Distributed Ledger Technology
DT	Digital Twin
DU	Distributed Unit
E2E	End-to-End
HLS	Higher Layer Split
IBE	Intent-based Entity
IBM	Intent-based Management
IBN	Intent-based Network
IME	Intent Management Entity
JCAS	Joint Communication and Sensing
kNN	k-Nearest Neighbours
KPI	Key Performance Indicator
KVI	Key Value Indicator
LLS	Lower Layer Split
LoS	Line of Sight
LoTAF	Level of Trust Assessment Function
MAC	Medium Access Control
MCE	Management Capabilities Exposure
MTP	Mobile Test Platform

NBI	North-bound Interface
NDT	Network Digital Twin
NLP	Natural Language Processing
NTN	Non-Terrestrial Networks
OAI	Open Air Interface
PCA	Principal Component Analysis
PDCP	Packet Data Convergence Protocol
PHY	Physical
PoC	Proof-of-Concept
PT	Physical Twin
QoD	Quality-on-Demand
RAN	Radio Access Network
RLC	Radio Link Control
RPC	Remote Procedure Call
RRC	Radio Resource Control
RU	Radio Unit
SBI	South-bound Interface
SDR	Software-Defined Radio
SLA	Service level Agreement
SP	Service Provider
SPR	Security, privacy and Resilience
SU	Sensing Unit
SM	Support Vector Machine
TEF	Trust Evaluation Function
TMS	Trust Management System
TX	Transmitter
UAV	Unmanned Aerial Vehicle
UE	User Equipment
UP	User Plane
WebUI	Web User Interface
Wi-Fi	Wireless Fidelity

1 Introduction

Hexa-X-II is the 6G Flagship project under the European Union Horizon Europe research and innovation program Smart Network and Services Joint Undertaking (SNS JU). This document is the sixth public deliverable of Work Package 2 (WP2), i.e. D2.6. The outcomes presented in this deliverable have a strong relationship with the works done in the previous deliverables: The foundation for this validation framework was first introduced in Hexa-X-II Deliverables D2.1 [HEX223-D21] and D2.2 [HEX223-D22], which laid out the initial concept and scope of System Proof-of-Concepts (PoCs). D2.3 [HEX224-D23] provided further details on System-PoC A, accompanied by simulation results, and a preliminary description of System-PoC B. The subsequent deliverable, D2.4 [HEX224-D24], offered a detailed breakdown of System-PoC B, including various validation results. D2.5 [HEX225-D25] further expanded on the final overall 6G system design, integrating insights from the PoC realisations and outlining the architecture's readiness for real-world deployment.

The validation of enablers of the 6G system is a crucial phase that ensures the feasibility, efficiency, and robustness of the proposed end-to-end (E2E) system architecture. This process is driven by the design and implementation of PoCs, which serve as practical testbeds for evaluating key 6G enablers across multiple dimensions, including management and orchestration over the 6G continuum, beyond communication enablers, network transformation, network control programmability and telemetry, 6G devices, and radio protocols.

The validation process that will be presented in the document focuses on:

- Assessing PoC-driven validation results to verify the effectiveness of key enablers in real-world and simulated environments.
- Evaluating system integration by mapping PoC components to the final E2E system blueprint and ensuring seamless interoperability.
- Measuring system performance against project-defined Key Performance Indicators (KPIs) and Key Value Indicators (KVIs).
- Demonstrating practical implementations of advanced Artificial Intelligence (AI)-powered automation and service orchestration, 6G radio protocols, novel devices, intent-based orchestration, and beyond communication aspects such as flexible topologies.
- Addressing migration challenges from 5G to 6G, ensuring that the proposed solutions align with future deployment scenarios.

1.1 Objective of the document

The main objectives of D2.6 can be identified with the following three elements:

- To introduce the final 6G E2E system design developed within the Hexa-X-II project.
- To provide a detailed evaluation of the E2E 6G system through System-PoC #C and its associated Component-PoCs. This involves mapping the tested components to the final system blueprint and assessing their performance and value in line with project-defined KPIs and KVIs.
- To conclude with an assessment of how the evaluated E2E system addresses the overarching project objectives of the Hexa-X-II system architecture for future deployment and standardisation efforts.

1.2 Structure of the document

This document is structured as follows:

- Chapter 1 introduces the document, offering an overview of its content and structure. It outlines the primary objectives of the deliverable and highlights its role and key contributions to the Hexa-X-II project.
- Chapter 2 provides a brief introduction on the design of the E2E system, including its architecture and alignment with the final 6G system blueprint. It lays the foundation for understanding the system's components and their integration.

- Chapter 3 presents the validation framework and technical realisation of System-PoC #C and associated Component PoCs. It covers AI-native orchestration, novel radio and device enablers, intent-based networking, trust mechanisms, closed-loop control, and system-level monitoring.
- Chapter 4 provides the validation and evaluation results, including real-world and simulation-based assessments. Results are analysed for KPIs and KVIs, covering aspects such as latency, energy consumption, trust index propagation, orchestration efficiency, and system sustainability. Additionally, this chapter discusses the impact of the evaluated 6G system. It elaborates on the list of open APIs that have been used in the context of the PoCs, on the open-source tools used, as well as on how the system addresses the overarching objectives of Hexa-X-II.

2 The E2E 6G System

This chapter presents the final E2E 6G system, as described in Hexa-X-II D2.5 [HEX224-D25]. While the prior phase of the project established the system architecture, key enablers, and validation framework, this deliverable focuses on assessing the real-world feasibility and performance of the proposed design through experimental validation. The E2E system blueprint, formulated with an emphasis on modularity, scalability, security, and automation, has now been tested in PoC implementations, allowing for an in-depth evaluation of advanced pervasive functionalities (intent-based management, closed loop controls, multi-platform orchestration, security, privacy and resilience mechanisms, AI orchestration, etc.), novel flexible topology-based networks, beyond communication functions such as localisation/sensing, their exposure via the application enablement platform layer, as well as advanced application layer components in the context of the Collaborative Robots use case. This chapter recaps the E2E system architecture in light of the validation results that will be presented later in the document, highlighting the effectiveness of key enablers of the system, such as novel devices and flexible topologies, service exposure of new 6G services, multi-stakeholder support mechanisms, and pervasive functionalities.

2.1 E2E system design overview

This section presents the final version of the 6G E2E system blueprint defined by Hexa-X-II (presented in detail in [HEX225-D25]) (Figure 2-1). The iterative design methodology started with Hexa-X-II D2.1 [HEX223-D21], and through the work developed by the technical work packages, the original system blueprint was refined, focusing on the specific enablers of the 6G system; during these multiple iterations, the enablers were analysed and reworked in order to fit the Hexa-X-II vision. Their integration in the E2E 6G system design considered the system design principles, the system requirements defined from the 6G use-cases, including the performances objectives and objectives related to the sustainability aspects. This final result presents a mapping of these selected enablers, representing the dependence between each other across the different layers of the 6G network, as well as the pervasive functionalities that support all system functions and interactions.

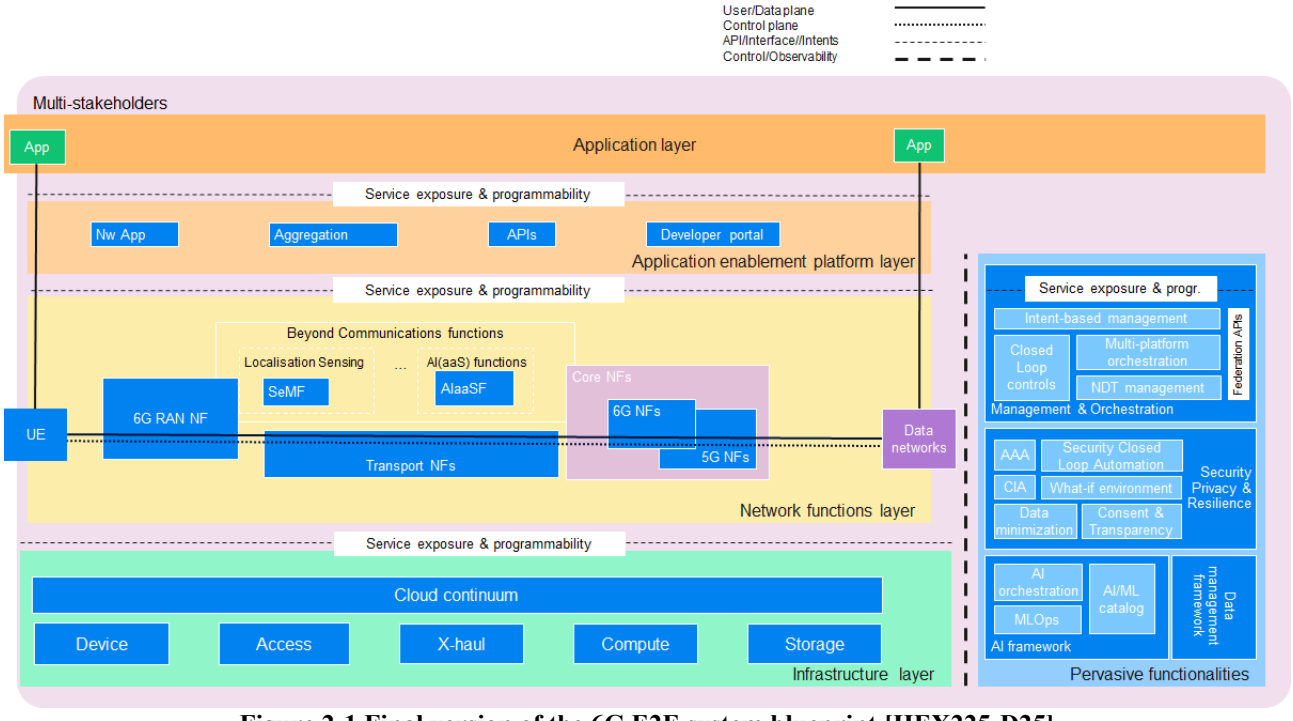


Figure 2-1 Final version of the 6G E2E system blueprint [HEX225-D25]

Infrastructure layer

The infrastructure layer, introduced in [HEX223-D21], sits at the lowest level of the system blueprint, encompassing the physical components that handle most 6G network transmission such as Device hardware like end-user equipment and IoT devices; Access technologies that connect these user devices to the core

network such as wireless access technologies (mobile networks, Fixed Wireless Access, Wi-Fi, etc.), wired access technologies (DSL, cable broadband, fibre access, etc.), but also other infrastructure dealing with the integration of Non-Terrestrial Networks (NTN) and Terrestrial Networks (TN); X-haul infrastructure supporting the access infrastructures to the core network through fronthaul and backhaul links; Compute hardware and resources (e.g. CPU, GPU) used to process data, host applications and network services; Storage hardware and resources used to store and retrieve data (e.g. HDD and SDD) hosted in data centres and data lakes, as well as distributed storage services, through edge and cloud storage resources. The Cloud continuum domain, stretching across the infrastructure layer, is abstracting and aggregating resources across public and private cloud services, on-premises and edge infrastructures. The interconnected resources in the infrastructure layer enable the E2E data transmission, as well as providing network and compute resources over which the various functionalities of the 6G system are executed and /or orchestrated.

Network functions layer

The Network functions layer, introduced early on in [HEX223-D21], encompasses the User Equipment (UE) box, that host the logical network functions residing in the devices used directly by end-users to communicate; 6G RAN functions that cover the different RAN deployment scenarios for various stakeholder needs, through centralized, distributed, cloud-based, for both physical network functions (PNFs) and virtualised network functions (VNFs); Core Network functions refined in [HEX225-D25] to host both 6G and 5G Core Network functions; in this design, the 6G CN shares selected NFs with 5GC, while it allows for introducing new NFs and non-backward compatible changes to existing NFs [HEX225-D35]. Beyond Communication Functions enhanced in [HEX224D23] to include functions that use sensing data collected from the Infrastructure layer and enhanced by AI through pre-built AI models and tools, providing the 6G network with new functionalities and services, such as the Sensing Management Function (SeMF) and AI as a Service Function (AIaaSF); Transport Network functions based on software defined networking (SDN) principles and virtualised transport network functions to enable the programmability of system elements in the network infrastructure; and Data Network, reflecting the functionalities related to services provided by the operator, such as Internet access or 3rd party services that support the increased usage of cloud infrastructures for current and future network deployments that are relevant for the scope of this 6G system blueprint.

Application enablement platform layer

The Application enablement platform layer, introduced in [HEX223-D21] as Network-centric application layer and renamed later to better reflect its purpose, provides applications access to the full range of 6G services for the development of new use cases and support to the Network functions layer for providing the enhanced capabilities for these applications. In [HEX225-D2.5], the application enablement platform layer has been extended to sit also on top of the UE side, so that an application client at the UE can built upon the 6G services exposed by the 6G platform. The Nw App box collects services that enable intelligent operations and optimisations across the network, for end-user and stakeholder usage, exposed via APIs. The APIs box gathers all APIs in a standardized manner (e.g. CAMARA), that abstract network capabilities and Management & Orchestration (M&O) services for developer usage in the Application layer. The Aggregation box represents the multiple services that can be collected from the multiple 6G service providers across multiple networks, connected through federation APIs. Finally, the Developer portal box introduced in [HEX224-D23] provides resources like specifications, documentation, SDKs, and test environments to facilitate development of these new applications.

Application layer

The Application layer, also part of the original blueprint in [HEX223-D21], is the higher layer of the 6G system blueprint, representing the applications available on the 6G system blueprint, interfacing with the Application enablement platform layer and Network functions layer to request the necessary service expectations from the network and exposing requirements such as bandwidth, latency, security, scalability, etc. Relevant network

information and control APIs are exposed through the Service exposure & programmability layer towards the Application layer, allowing for adjustments according to current network conditions and capabilities.

Pervasive functionalities

The pervasive functionalities were envisioned from the start [HEX223-D21] to enable and enhance all four layers of the 6G system blueprint, reaching full potential of the 6G platform. This facilitation is achieved independently or through joint operation, as these functionalities may permeate the entire system blueprint.

The data management framework is designed to support the complex data requirements from 6G system, by gathering diverse data from various network domains and layers. This framework should support cloud-native observability, enabling the collection and processing of various signals to generate valuable insights, improve monitoring capability and reduce energy consumption. This framework adheres to DataOps practices, automating the lifecycle management of distributed data across the cloud continuum and enabling the availability of high-quality, robust and secure data.

The AI/ML framework enables a data-driven architecture with distributed intelligence for dynamic resource allocation and optimisation necessary to 6G applications and services. Both domain-specific and cross-domain AI services are covered, ranging from specialized AI for specific tasks to AI spanning multiple layers and functionalities, enabling AI mitigation for hardware impairments [LSL+22], zero-touch management operations [KLM+22], resource orchestration across the computing continuum, among other AI embedded network functions [ZFF+24]. This framework was enhanced in [HEX225-D25] to include an AI orchestrator, an AI/ML catalogue and an MLOps Orchestrator, to manage the lifecycle of AI services, ensuring efficient deployment, optimisation, and continuous improvement.

The security, privacy and resilience pervasive functionality addresses the new security and privacy challenges brought on from 6G network capabilities, requiring a value-oriented, E2E system design prioritizing trustworthiness. The implementation of this framework was improved in [HEX225-D25] through outlining security, privacy, and resilience controls such as Confidentiality, Integrity, and Availability (CIA) solutions, through cryptography, redundancy, and quantum-safe solutions; Authentication, Authorization, and Accounting (AAA) processes that ensure secure access and identity management; Data minimisation, consent, and transparency safeguarding data privacy and confidentiality; enforcement mechanisms through Network Digital Twins (NDTs) that deal with "What-if" scenarios, simulating network conditions to optimise network resilience; and through the implementation of Security Closed Loops, that ensure the 6G network is autonomously secure and reliable.

The management and orchestration framework addresses the complex 6G ecosystem, through simplified operations, intuitive interfaces, automated processes, and intelligent decision-making. Intent-based management was defined early in [HEX223-D22] as a key pillar toward simplification, enabling user-friendly service requests and supporting diverse vocabulary for service deployment. Also, the development of different M&O enablers raised the necessity in [HEX224-D23] of including specific closed loop controls, powered by AI/ML, that achieve automation for 6G network operations. Multi-platform orchestration provides unified management and orchestration of network services and applications across this cloud continuum, ensuring seamless management across diverse network environments. In [HEX225-D25], through effective monitoring and telemetry interfaces that gather real-time data, NDTs can play an important role in for sustainable and efficient networking, which is the solution that the NDT management box offers for safe AI/ML-based orchestration and control.

Service exposure and programmability framework

The Service exposure and programmability framework, introduced in [HEX224-D23], is implemented at the different layers of the system blueprint and is tasked with leveraging the exposure of data, resource, service and control capabilities to both internal and external entities, as well as providing APIs for service discovery, access control, and enforcement, encompassing data, resources, and control across various layers. It was extended in [HEX225-D25] to include the programmability of the network over 6G capabilities, allowing for tasks like configuring quality of service for specific UEs. From the M&O perspective, it also enables the

exposure of management services through APIs or intent-based APIs, augmented with services from network applications, from different stakeholders for end-user applications.

Multistakeholder support

The 6G ecosystem will be characterized by increased interactions between diverse stakeholders, shifting from a linear value chain toward a multi-sided one. This evolution necessitates the definition of new roles as outlined in [HEX223-D22] including Digital Service Providers (DSPs), Capability Operators, and Resource Providers, alongside various digital service customers (B2B, Application service providers, aggregators). The multi-stakeholder dimension, introduced in [HEX224-D23], is particularly relevant in the 6G cloud continuum where interactions between multiple DSPs offering digital services to customers, and collaboration between capability operators and diverse resource providers (network, cloud, AI, applications) for the management and orchestration of E2E services are crucial [HEX225-D65]. Furthermore, integrating resources provided by vertical industries and extreme-edge resources belonging to end-users is essential for aggregating heterogenous resource and realising the network cloud continuum [HX225-D35], through a secure federation model and federation APIs.

Integration of enablers into the system blueprint

To address the complexity of 6G system design, a knowledge graph approach was developed to identify the optimal set of enablers for specific use cases. This approach, detailed in [HEX225-D25], was applied to the industrial cobots use case. The chosen enablers, well-suited for this use case, and detailed in Annex C in [HEX225-D25] are mapped into the blueprint architecture in Figure 2-2. For clarity, the figure represents enabler categories rather than individual enablers.

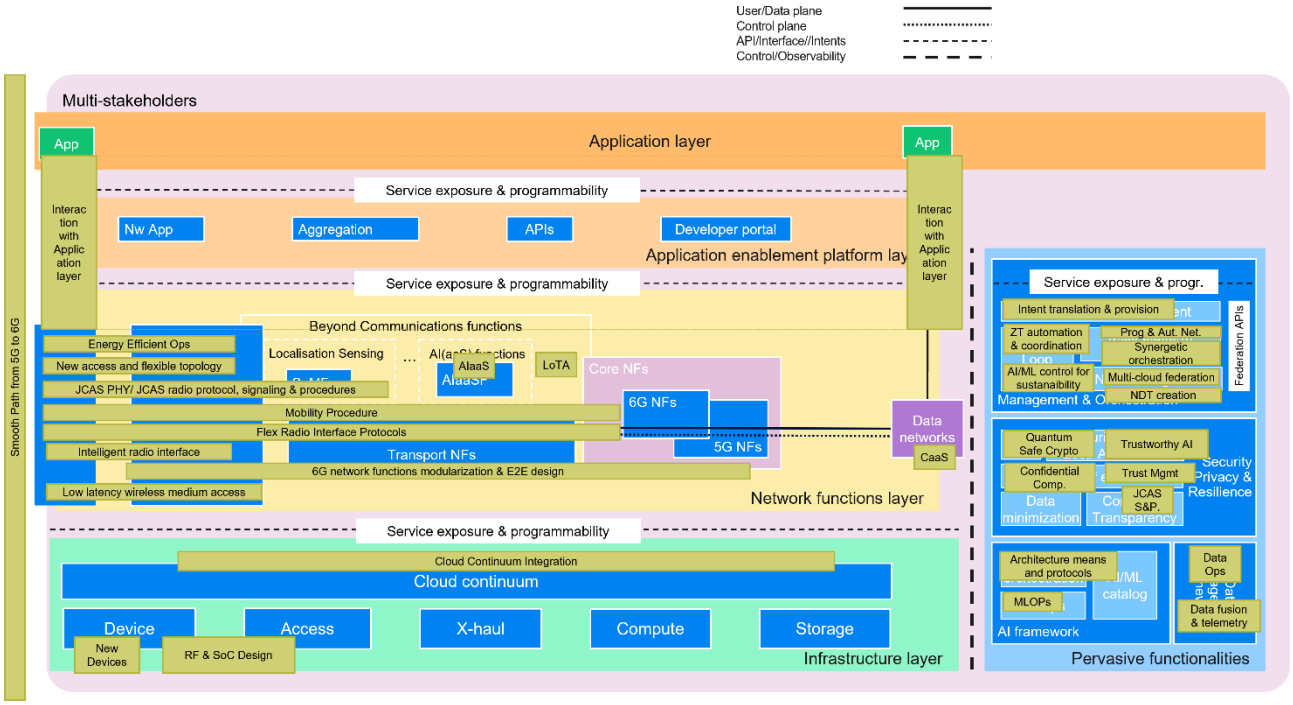


Figure 2-2: System blueprint with categorized enablers suited to the industrial cobot use case

The following section details which of the selected enablers have been implemented in the system-PoC.

2.2 System-PoC E2E architecture and alignment with the final 6G system blueprint

Figure 2-3 illustrates the components of System PoC-C to the E2E 6G system blueprint (Figure 2-1), showcasing their alignment with the project's key design principles.

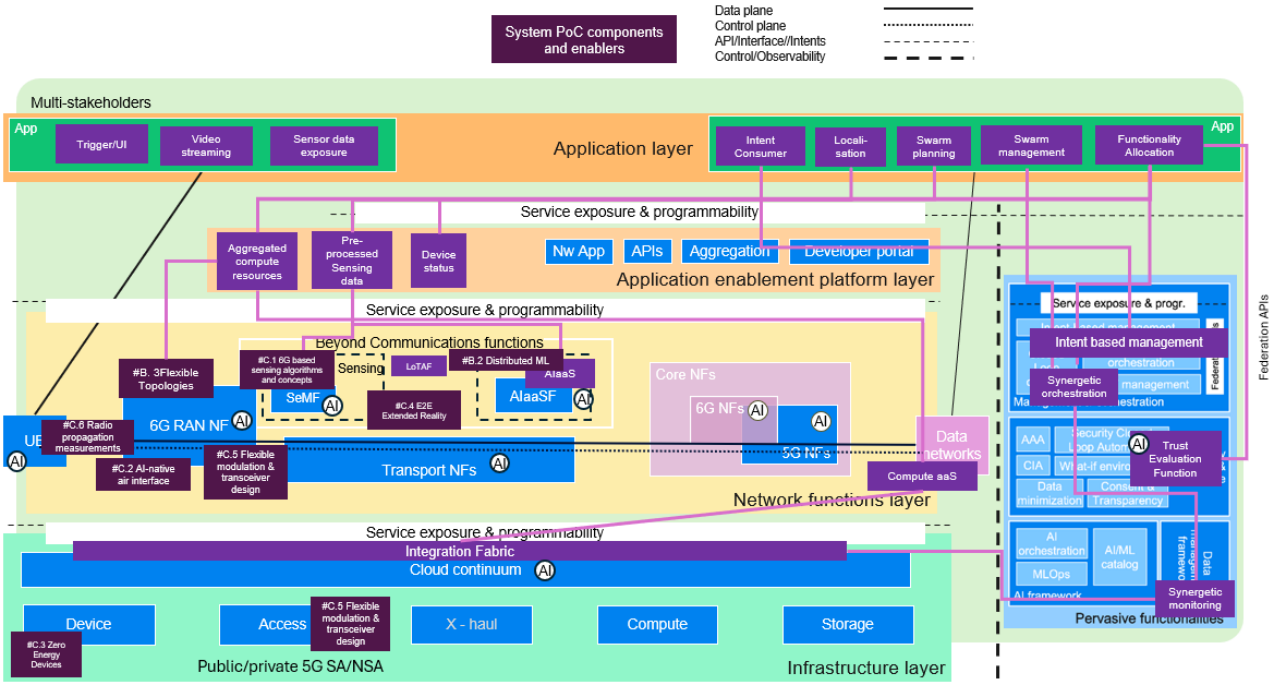


Figure 2-3 System-PoC #C components mapping to 6G system blueprint

The Hexa-X-II System PoC architecture illustrates various components (highlighted in purple), which collectively enable intelligent automation, AI-driven decision-making, flexible and sustainable radio performance, as well as real-time orchestration across various 6G applications such as collaborative robotics (cobots), immersive environments, and multi-stakeholder service coordination.

At the application layer, System-PoC #C elements enable advanced user interaction with the exposed infrastructure and network services, data and capabilities. The Trigger/UI module enhances user interaction by providing an intuitive interface, while video streaming, sensor data exposure, and device status monitoring facilitate real-time situational awareness. These elements enable the seamless retrieval and processing of visual feeds, sensor data, and device telemetry, which are crucial for applications like collaborative robotics and smart logistics. Additionally, aggregated compute resources and pre-processed sensing data contribute to optimising workload distribution, ensuring that AI-driven applications receive the necessary computational power for real-time decision-making.

Moving to the application enablement platform layer, the PoC components provide the necessary programmability and service exposure capabilities, allowing for the seamless integration of AI-powered applications. New application modules and API exposure mechanisms facilitate dynamic service composition, while aggregation mechanisms and developer portals create a programmable environment for service orchestration and multi-stakeholder collaboration. These functionalities contribute to making 6G networks more adaptable and responsive to evolving application demands.

At the network functions layer, the PoC components validate advanced communication enablers and AI-driven network intelligence. The 6G RAN Network Function (NF) supports flexible network topologies, ensuring efficient spectrum utilization and adaptive radio control. AI-native air interfaces and optimised transceiver designs further enhance spectral efficiency, power consumption, and latency-sensitive applications. The beyond communication function capabilities comprise the AI-as-a-Service (AIaaS) framework that enables real-time predictive analytics and intelligent decision-making, advanced sensing capabilities leveraging fusion of multi-modal sensing sources, while the Level of Trust Assessment Function (LoTAF) enables quantified assessment of trust levels of involved entities. These advancements are complemented by innovations such as flexible modulation techniques, precise radio propagation models, and AI-driven sensing mechanisms, all of which contribute to a more adaptive and intelligent 6G network.

The integration fabric serves as the backbone of System PoC-C by interconnecting diverse compute, storage, and networking resources. It provides the necessary infrastructure for distributed computing, enabling real-

time AI execution across the cloud continuum. The cloud continuum architecture ensures that computational workloads are dynamically allocated between edge, fog, and cloud layers, optimising processing efficiency while maintaining low-latency data exchange. Through advanced resource orchestration, the integration fabric enhances service exposure, ensuring that 6G applications can seamlessly leverage distributed computing capabilities.

The pervasive functionalities, comprising the management, orchestration, and security domains, showcased by the PoC-C components introduce AI-driven automation and security mechanisms that ensure autonomous system control and trustworthiness. Intent-based management frameworks enable dynamic decision-making by translating high-level service intents into optimised resource allocation strategies. AI-powered closed-loop orchestration ensures that the system remains adaptive and self-optimising, allowing for automated fault detection and recovery. To reinforce system security and resilience, trust evaluation functions and synergetic monitoring mechanisms continuously assess network integrity, providing real-time threat detection and adaptive risk mitigation. AI/ML frameworks further enhance security by leveraging MLOps-based monitoring, ensuring that system vulnerabilities are proactively identified and addressed.

3 E2E system-level validation design

3.1 Overview of the E2E system validation activities

The goal of the Hexa-X-II project is to move from the early stages of 6G technology development to defining a fully integrated 6G system, with a strong emphasis on social, environmental, and economic sustainability. For achieving this, a holistic approach was introduced including the development of innovative devices, infrastructure, radio and network capabilities, E2E management and orchestration, as well as enhanced security and system-level resilience. To support this, three system prototypes (PoCs) were developed progressively throughout the project's lifecycle.

The first iteration, System-PoC #A, detailed in previous reports [HEX223-D22], [HEX224-D23], focused on smart network management, demonstrating management mechanisms essential for 6G. It established a foundation for further development in E2E system evaluations within a single-domain configuration, utilizing a testbed with autonomous mobile robots (AMRs) and unmanned aerial vehicles (UAVs) to support an automated warehouse inventory management scenario. System-PoC A focused on key aspects such as:

- Intent-Based Orchestration: Facilitating seamless interaction across various domains, including application, AI, and service management resources.
- Energy-Efficient Functionality Allocation (FA): Implementing a metaheuristic algorithm to optimise workload placement across compute nodes, ensuring energy efficiency and system trustworthiness.
- Trust Evaluation Functions: Real-time trust evaluations that informed workload placement, enhancing system reliability.
- Advanced Resource Allocation: Dynamically assigning tasks such as object detection and quality inspection, considering energy availability, proximity, and hardware capabilities.

The second iteration, System-PoC #B, presented in earlier deliverables [HEX224-D23] [HEX224-D24], expanded on System-PoC #A by incorporating new enablers such as pervasive technologies and network functions. This iteration also assumed that not all network and cloud domain resources, application components, and devices were located within the same domain or geographical area. As such, it required meticulous orchestration to ensure coherent and synchronized operation of all components across different domains. The integration of new enablers and refinements to existing ones extended the sustainability considerations from System-PoC #A, including exploring data exposure security for social sustainability and dynamic re-planning of device roles for environmental sustainability. System-PoC #B expanded on the results from PoC A, incorporating more advanced network architecture elements such as flexible topologies, network-of-networks enablers, and beyond-communication network aspects. This iteration also introduced refinements to the management mechanisms introduced in System-PoC #A, focusing on multi-domain configurations and extending the functionality to support latency-sensitive services. Key enhancements included:

- Flexible Topologies: Enabling dynamic exposure and exploitation of temporary communication and compute resources.

- Beyond Communication Features: Integration of Compute-as-a-Service (CaaS) and AI-as-a-Service (AIaaS), allowing for real-time resource exposure and utilization.
- Enhanced Applications: Supporting latency-sensitive services, such as video streaming and object detection, ensuring robust performance in dynamic environments.
- Advanced Orchestration and Management: Utilizing intent-driven resource allocation and multi-platform orchestration for seamless operation across domains.
- Security and Privacy Enhancements: Reinforcing system resilience through trust evaluation functions.

The third iteration, System-PoC #C, further evolves the 6G architecture design and smart network management concepts introduced in earlier PoCs, with a focus on radio and device aspects. This iteration was first described in [HEX225-D25] but a more complete description together with the final validation outcomes and evaluation results are reported in detail in this deliverable. This iteration places special emphasis on radio and device aspects, advancing the technological components necessary for a comprehensive 6G system.

3.2 Elements of the System-PoC #C and Components-PoC #C

System-PoC C builds upon the advancements introduced in System-PoC #B, further expanding the capabilities of the 6G system by integrating new enablers derived from 6G radio technologies and 6G devices, developed under WP4 and WP5, respectively. These innovations are incorporated both within the System-PoC itself and through complementary Component PoC Cs that run in parallel.

The Component PoC elements comprise 6G radio and 6G device components, namely:

- 6G-based sensing algorithms and concepts with real-time performance
- AI-native air interface
- ML-based channel state feedback compression in multi-vendor scenarios
- Flexible modulation and transceiver design
- Radio propagation measurements for radio channel modelling
- Crowd-detectable zero-energy devices
- End-to-end extended reality

Overall, the System-PoC #C extends its scope and features via the following directions:

- New APIs leveraging exposure of Application Enablement Platform capabilities (Figure 3-1), such as Quality-on-Demand (QoD), device status/density, as well as integrating System PoC-B capabilities, such as advanced flexible topologies functionality and compute offloading e features
- Integration Fabric adoption as a core enabler in multi-domain resource orchestration scenarios
- Multi-agent reinforcement learning for multi-cluster orchestration
- New devices with energy harvesting capabilities as part of the warehouse management scenario
- Advancement from reactive to cognitive Closed-Loop Automation with AI-driven predictive analytics for proactive resource planning
- Intent-driven orchestration leveraging intent hierarchy, i.e., business intents, as well as intent-based DSM capabilities

Two driving scenarios are leveraged to showcase system-level advancements, both under the Collaborative Robots UC, namely the Cobot-powered Warehouse Inventory Management, which is the driving System PoC scenario as an evolution from System-PoC #A, and the Cobot-powered Video Surveillance scenario, via the evolution of the Component-PoC #B.1 system-level features.

Regarding new devices, System-PoC #C introduces proximity-sensing devices with energy-harvesting capabilities into the warehouse inventory management scenario. These devices, designed to accurately monitor shelf occupancy, incorporate solar-powered energy harvesting, enabling sustainable and autonomous operation. Real-time updates of digital twins representing warehouse storage areas are facilitated through these devices, which provide precise shelf occupancy data. This data is then utilised for enhancing the FA algorithm, as previously introduced in [HEX224-D24]. Additionally, these devices feature an intelligent wake-up/sleep cycle adjustment mechanism, powered by an integrated TinyML model, ensuring optimised energy efficiency.

In terms of APIs exposed via the Application Enablement Platform, the Warehouse Inventory Management platform has been enhanced with new capabilities that leverage infrastructure and network-level data such as localization, compute-as-a-service, QoD, and device status insights as reported in [HEX225-D25] and shown in Figure 3-1. Notably, some of these APIs, including device status/density and QoD, are designed to be CAMARA-compliant [CAMARA], ensuring seamless interoperability within broader 6G service ecosystems.

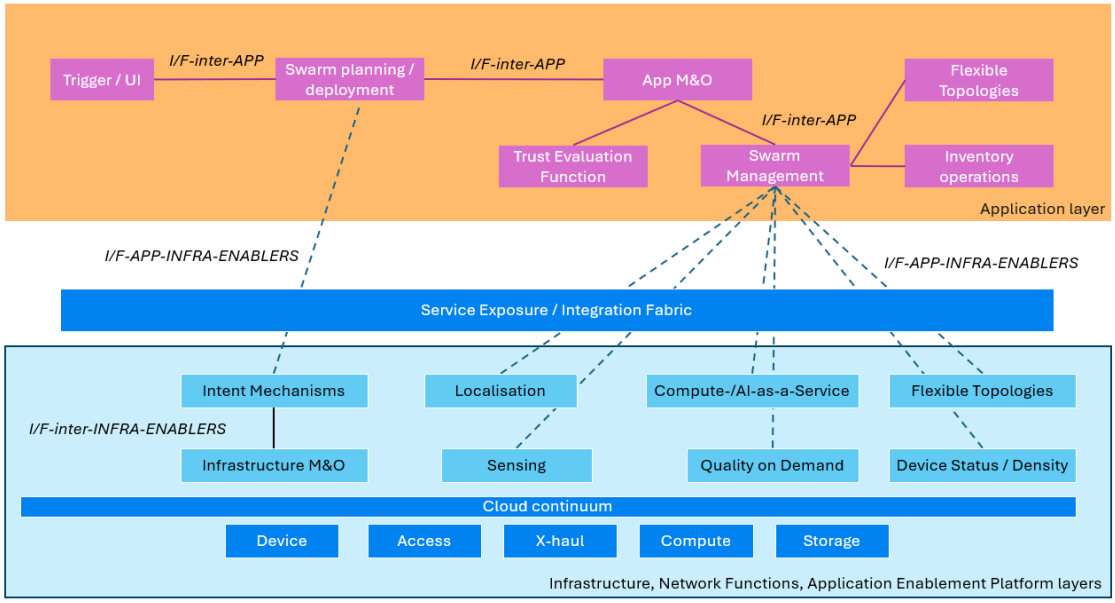


Figure 3-1 Simplified view of the Warehouse inventory management application components interfaces with E2E system's Application Enablement Platform exposed capabilities

3.3 System-PoC #C elements as evolution of System-PoC #A and #B components

3.3.1 Sustainability-oriented orchestration in 6G

As outlined in [HEX223-D22, HEX224-D24], one of the system demonstration configurations involved a warehouse inventory management scenario powered by cobots. This setup utilises advanced computer vision and sensor data fusion to accurately detect, count, and localise items in real time. Additionally, a dynamic translation system allows seamless conversion between symbolic warehouse locations and precise 3D geometric coordinates, improving the navigation of drones and AMRs for efficient inventory tracking.

Several critical components contribute to the functionality of this system. The inventory realisation component encompassed all essential functionalities related to identifying and tracking warehouse objects, including specialised computer vision modules. The video streaming service handles the transmission of data streams from AMRs and UAVs for further processing. Sensor data exposure mechanisms and interfaces ensured that AMRs, UAVs, and infrastructure sensor information remained accessible to the respective data consumers.

To support user interaction, an intent-based tool allows warehouse managers and application end-users to input high-level commands and requests concerning inventory operations, performance expectations, task duration, and resource allocation. The localisation service provided the necessary functionality to supply accurate location data to relevant consumer modules, such as those responsible for swarm planning and management. The swarm planning/deployment service is responsible for reserving AMR and UAV resources to execute inventory operations, while it monitors runtime performance, detected operational issues, and handles events such as battery depletion or hardware failures.

The functionality allocation (FA) algorithm focuses on optimising the placement of inventory management workloads and computationally intensive tasks. The system also incorporates a Trust Evaluation Function (TEF) for analysing real-time and historical data from compute nodes to assess trust indexes. These trust indexes are used by the FA component to allocate workloads to compute nodes with higher trust scores, thereby

improving the overall trustworthiness of the system. The component responsible for assessing the trust indexes/levels on the elements of interest has been enhanced, as described above.

Additionally, a synergetic monitoring service is utilised to facilitate the exposure of infrastructure, device, and network monitoring metrics across multiple domains in a unified manner, supporting the joint optimisation of available resources.

Several enhancements and additions have been introduced to the cobot-powered warehouse inventory management scenario. One of the key additions is the integration of solar energy harvesting devices with proximity sensors. These devices have been prototyped to improve efficiency, reduce operational time and lower AMR/UAV energy consumption. They are placed on warehouse shelves to indicate shelf occupancy. If only a single shelf is occupied, an alert is sent for inspection. The detailed functionality of these devices is presented in Section 3.5.2.

The orchestration-related components that have been restructured and enhanced with this addition include the FA, and the swarm planning/deployment service. These components along with the associated data models, have been extended to support and integrate the capabilities of the newly prototyped devices.

Additionally, the FA mechanism evolved to enable device-specific functionality allocation optimisation. This enhancement considered various physical tasks/roles with their requirements, such as robotic-specific features, cameras, wheels, propellers, the physical distance for task handling and the various physical nodes, with their capabilities (e.g., available memory, CPU), power consumption, battery level, robotic-specific features, and physical location. The objective was to minimise total energy consumption, reduce overall travel distance, and balance workload among robots by minimising the longest individual route. The FA physical task planning problem statement and formulation is described in [Section 2.1.4.2, HEX225-D65]. As presented there, it was addressed through the development of a metaheuristic algorithm based on the ant colony optimisation approach (ACO) [DSS04]. Simulation results of this algorithm can be found in Section 4.4.14 of this deliverable and in [HEX225-D65].

Another significant enhancement is related to the trust management system, which has been incorporated to assess the trust indexes of the compute nodes in the system. The trust management system integrates TEF and the LoTAF, resulting in an improved trustworthiness estimation. TEF has been introduced in [Section 3.4.3, HEX224-D63] and some simulation results of its integration in the cobot-powered inventory management PoC were reported in [HEX224-D24].

When it comes to the LoTAF, such a mechanism introduces an add-on over the FA algorithm since an additional set of trust features are evaluated to determine trust indexes. Concretely, LoTAF extends trustworthiness computation by adding real-time quantitative features regarding CPU, memory, network, processes, and disk. These features are continuously gathered while LoTAF is running on the compute nodes/robotic units, and such information is directly collected from the low-level operative system. Unlike TEF, LoTAF is capable on its own of collecting the necessary monitoring parameters to calculate its confidence index through its internal monitoring agent. In addition, a set of pre-defined rules has been declared to continuously verify whether health scores associate with the CPU, memory, processes, network, and disk have the expected behaviour.

Data models generated by LoTAF are continuously forwarded to the Management Capability Exposure (MCE), as it is the link between TEF and LoTAF to calculate a unified trust index. In this vein, multiple communication buses via the MCE have been designed to enable integration with the TEF, LoTAF, and Trust Management System.

Figure 3-2 shows the message sequence chart of the trust management system. Different system components are involved in retrieving, processing, and publishing compute nodes' trust indexes. The message sequence chart visually illustrates how these components communicate and work in parallel to fulfill a request.

The process begins from the API Server incorporated in the swarm planning/deployment service, which initiates a request to the Trust Management for computing trust indexes for specific compute node IDs. This request is processed by the Trust Management and is then published to the MCE - Integration Fabric, which serves as an intermediary responsible for handling and distributing the request. Once published, the request is consumed by both components, the TEF and LoTAF.

At this stage, the system enters a parallel execution phase. The TEF component sends a request to Infrastructure Monitoring in order to obtain compute node information, and once the data is gathered, it processes them and returns the calculated trust indexes. Meanwhile, the LoTAF component internally gathers the health metrics and generates trust indexes based on the compute nodes' data. This process involves generating compute node information for specific IDs and subsequently publishing the computed trust indexes.

As the trust indexes are generated, they are published back to the MCE - Integration Fabric, ensuring that the relevant component like the Trust Management can consume the published trust indexes. At this point, the Trust Management carries out a simple moving average to combine both trust indexes with the same significance. Finally, after processing, the trust indexes are sent back to the API Server, completing the process. Time-related results of this integration are reported in Section 4.4.14.

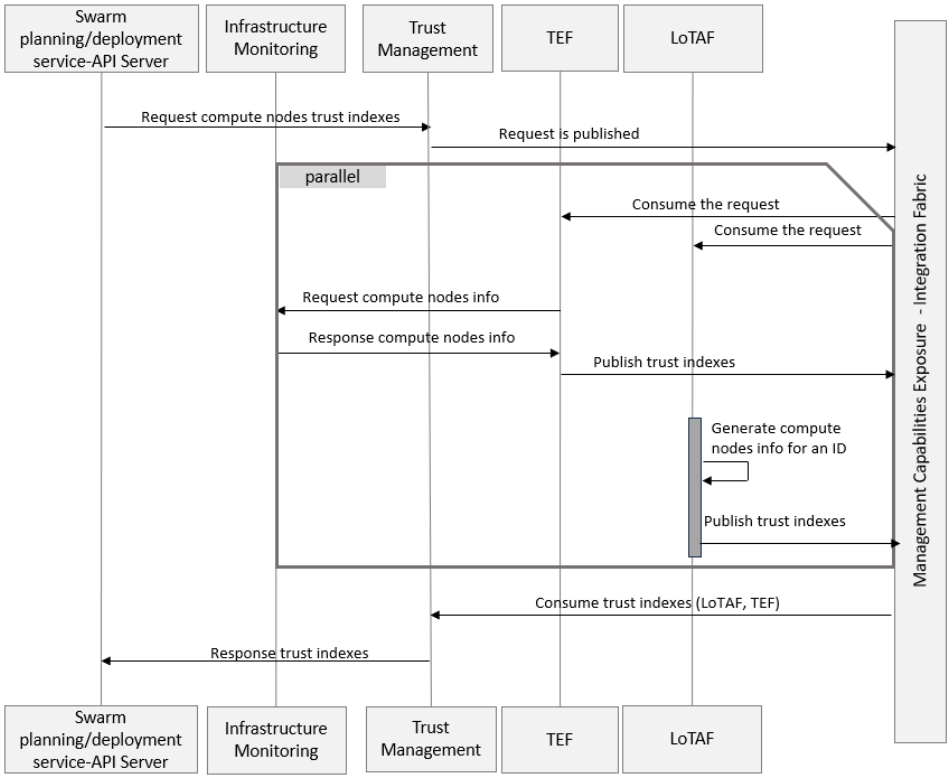


Figure 3-2 Message Sequence Diagram for Trust Management Processing: Receiving TEF and LoTAF Indexes and Feeding the swarm planning/deployment service.

3.3.2 AI-assisted E2E lifecycle management of a 6G latency-sensitive service across the compute continuum

A latency-sensitive smart manufacturing microservice-based application was developed for managing the operation of a stationary robotic arm, composed of two service chains: one regarding the teleoperation of a robotic vehicle surveilling the workspace of the arm and a second regarding the generation of alerts to safely pause and start the arm's operation based on an ML-based object detection component analyzing the video stream of the vehicle. Specifically, the first service chain comprises three essential components: the *encoder*, responsible for encoding frames before forwarding the stream; the *media server*, handling the received stream for distribution and management; and the *front end*, serving as the user interface for accessing and interacting with the vehicle. The object detection service chain also comprises three main components: the frame sampler, which samples the frames from the stream with a configurable sampling rate; the rescaler, which resizes the sampled frames to an appropriate resolution, ensuring compatibility with the object detection process; and the object detector, which analyzes the frames and generates alerts in an emergency.

The two service chains define the workloads that are going to stress the network and the computing infrastructure as follows:

- Object detection – Arm control: Object detection through the robot surveillance camera is continuous with a low sampling rate (normal operation). Once a person is identified in the camera footage, the object detector tries to validate if there is indeed a human in the location by increasing the sampling rate (i.e., we move to warning operation). If there is no human, we return to normal operation. If the module continues to identify a human in the incoming footage, an alert is sent to stop the robotic arm and the sampling rate is decreased (i.e., we move to danger operation). Once the human is out of the camera's scope, the object detector re-initiates the arm's operation, and we return to normal operation.
- Live video streaming and teleoperation: This workflow is a high-bandwidth (5-30Mbps depending on frame rate and resolution) , high-availability scenario that runs continuously and poses a communication and computational challenge for maintaining its uninterrupted operation.

In multi-cluster setups, a single operator may be responsible for managing service deployment across multiple clusters. However, individual cluster operators may also deploy their own orchestration agents to manage local resources. Depending on agreements between operators, such layered environments can give rise to various inter-agent dynamics. This Component PoC evaluates a series of different agent networks that collaboratively or independently try to tackle concurrent scaling and migration of services. Specifically, in the scenarios considered, the agents need to select a suitable number of replicas (with regard to performance/cost trade-offs) to handle incoming workloads. At the same time, the multi-cluster setup introduces a complexity regarding the services' proximity to the source of the requests. The agents interact with the multi-cluster environment to obtain state information (e.g., latency, CPU usage, memory, request rate) and to initiate scaling and migration actions. During training, the agents learn to take actions using state transitions, reward feedback, and communication messages from other agents, while during execution, they take optimal actions based on their models. Responsibility distribution is key here, since central control increases optimality but also complexity, and vice versa for decentralized approaches.

Initially, a centralized Reinforcement Learning (RL)-based scheme was employed. In the evolution of the PoC, a hierarchical approach using a decentralized Multi-Agent RL (MARL) methodology is explored, enabling distributed agent coordination. In particular, each agent tackles a different task: agent 1 migrates the service across clusters, and agent 2 handles the scaling factor of the service. The multi-agent mechanism is compared against a centralized, RL-based, joint autoscaling-placement mechanism managing the application. The orchestration actions of both setups are decided based on the evaluation of real-time data regarding computation and communication delays collected at specific monitoring points. The autoscaling and placement mechanisms evaluate the collected data, and for every interval, the agents select where the optimal placement is and how many replicas are needed for guaranteeing Service Level Objectives satisfaction in terms of end-to-end latency requirements. Detailed results for the aforementioned evaluation process can be found in D6.5 [HEX225-D65].

3.3.3 Training and inference of collaborative distributed machine learning model on a dynamically changing heterogeneous 6G architecture

This PoC enables collaborative model training across application and networks using split learning, a technique for vertical federated learning. This approach allows joint neural network (NN) models to be trained for tasks such as video streaming quality estimation, without sharing raw data between entities. By dynamically offloading NN model layers based on resource availability (compute, memory, energy), the PoC ensures efficient use of resources while maintaining model accuracy. The neural network architecture includes distributed input nodes processing domain-specific data and a generalization node at the network. It also includes output nodes that can be deployed at the external applications. The application integrates outputs for multiple tasks like delay and throughput predictions. Key innovations with split learning include energy-aware model layer offloading to balance performance and energy consumption. These features showcase the potential of split learning for secure, efficient, and adaptive AI-driven applications in 6G networks. Please see 4.3.2.2 for the results.

3.3.4 Trustworthy flexible topologies in 6G, leveraging on "beyond communication" aspects

The evolution towards 6G wireless networks has necessitated the development of network architectures that are both dynamic and inherently trustworthy. This section presents a novel approach based on unstructured, flexible topologies that can adapt in real time to varying service requirements and environmental conditions. In contrast to traditional architectures, the proposed approach integrates beyond communication services (BCS) by embedding advanced data processing and insight extraction capabilities directly into the network fabric. Trustworthiness in this context extends well beyond conventional security measures to encompass capacity, energy efficiency, throughput, and cost, as well as the minimisation of unnecessary data exposure. Central to the architecture is an Adhoc Network Controller that orchestrates the selection and configuration of network nodes. This controller continuously monitors the network environment through robust node discovery mechanisms and employs a Trust Manager coupled with a Trust Evaluation Function. These components assess critical parameters such as computational capacity, energy resources, latency, throughput, and deployment costs to generate a composite trust metric. This multidimensional metric is used to determine the optimal selection and placement of nodes and allocate processing functions in an efficient manner. The emphasis on trust management ensures that only the most reliable and efficient nodes are selected, thereby enhancing overall network resilience and performance.

The system employs a dual approach to topology instantiation, combining systematic methods with heuristic optimisation techniques to select and deploy the most efficient nodes. Following the selection procedure, advanced node discovery protocols and efficient algorithms, such as Minimum Spanning Tree formation, are employed to optimise UAV placement and interconnectivity. Using advanced techniques, the node discovery process efficiently approximates optimal UAV positions, reducing computational overhead and enhancing scalability. The subsequent Minimun Spanning Tree (MST) formation, as illustrated in Figure 3-3, minimises connection costs and ensures robust network connectivity even in highly dynamic environments. Energy efficiency is further addressed through AI/ML algorithms that prioritize energy conservation by optimising UAV flight paths and minimising redundant movements.

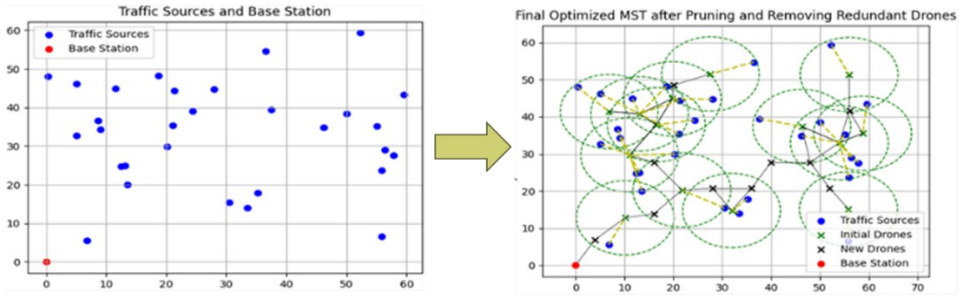


Figure 3-3 Flexible UAV placement process transforming an initial distribution of traffic sources (a) into an optimised, low-cost network topology (b)

Furthermore, the integration of CAMARA-compliant APIs facilitates standardized exposure of network metrics and inter-domain communication. These APIs enable seamless interoperability between various network functions and operators, thereby supporting real-time coordination of connectivity, resource allocation, and application placement. Indicatively, Figure 3-4 presents a high-level message sequence chart (MSC) that outlines the Flexible Topology Network (FTN) orchestrating key operations such as authentication, resource allocation, and mobility management across various network functions. In this workflow, the FTN initiates authentication and registration procedures via the NEF and AMF, each step validated by corresponding responses. Once registration is confirmed, resource requests traverse the NEF to the SMF and PCF, ensuring QoS adherence before the AI/ML Resource Optimisation Component adjusts allocations in real time. Meanwhile, the AMF and UPF handle node discovery and data flows to maintain seamless connectivity. Throughout these interactions, metrics such as latency, throughput, and energy consumption guide decision-making, reinforcing both reliability and performance within the network.

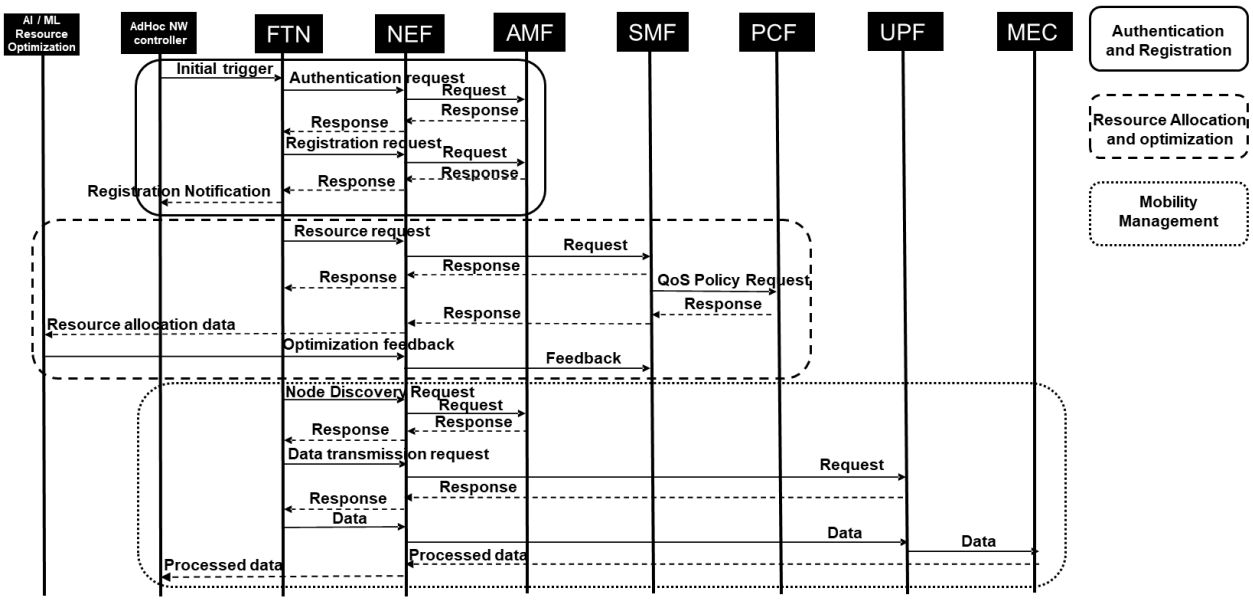


Figure 3-4 High-Level Message Flow for Flexible Topology Network Orchestration Using Exposed Network Functions

This optimisation ensures that selected UAVs enhance trustworthiness while adhering to capacity and energy constraints, achieving a balanced and cost-effective network deployment. Following this procedure, in practical applications such as disaster recovery or large-scale outdoor events, UAV-assisted topologies exhibit exceptional flexibility and resilience. For example, in disaster response operations, UAVs are dynamically deployed to establish temporary communication links in areas where terrestrial infrastructure has been compromised. The trustworthiness metric guarantees that UAVs deployed in such critical situations achieve an optimal balance between reliability and performance, facilitating uninterrupted communication and coordination among emergency responders.

3.3.5 Synergetic monitoring and orchestration

The multi-domain, synergetic monitoring and orchestration scenario demonstrates dynamic management of services across multiple sites, ensuring efficient resource allocation considering strict service-level requirements. It highlights how workflows are orchestrated and adjusted in real time based on network conditions, resource availability, and performance metrics. The scenario emphasizes seamless migration of workloads and effective orchestration between sites to maintain optimal system performance. The general description of the scenario is as follows:

A Service Provider (SP) wants to deploy a service that must meet a strict Service Level Agreement (SLA) while minimising costs. The SP has two deployment options:

1. Hosting the service on local infrastructure, where they have full control, but incurs maintenance costs and must manage the corresponding resources to satisfy the given SLA
2. Offloading the service to a Cloud Provider (CP), which charges based on the chosen service tier that introduces specific guarantees. Since services may be deployed across different locations, connectivity between co-dependent services is facilitated by Network Providers (NPs). The NP offers network guarantees, but at a corresponding cost.

The SP manages the service deployment by deciding: 1) Where to deploy the service, 2) How to manage on-premises resources, 3) What network guarantees to request from the NP, 4) What processing guarantees to request from the CP. A high-level schematical representation of the scenario is shown in Figure 3-5.

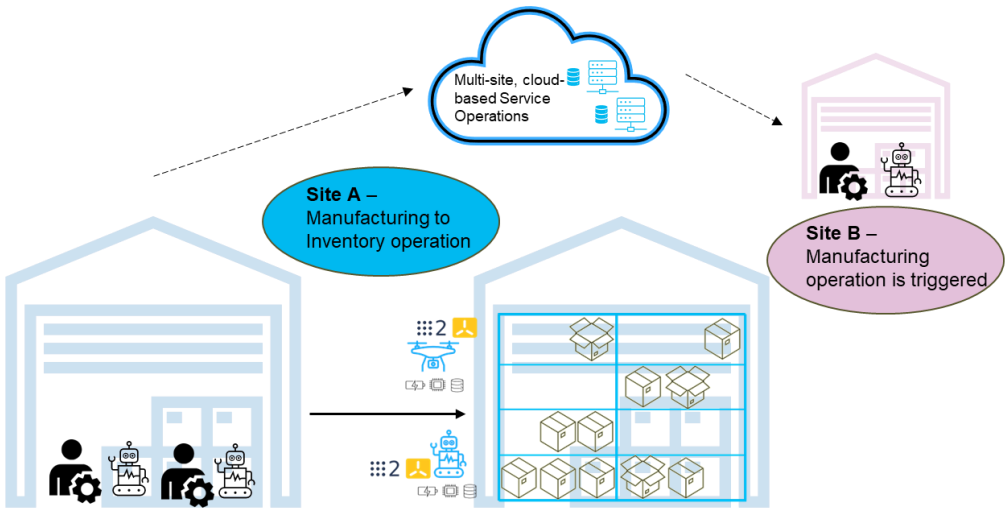


Figure 3-5 Multi-domain synergetic monitoring and orchestration scenario.

The scenario implementation involves two key services. The Service A, which is a manufacturing-related latency-sensitive service, and the Service B, which is a warehouse inventory management service. The Service A consists of two workflows, a video streaming workflow (Workflow A.1) and an object detection workflow (Workflow A.2).

The E2E scenario begins at time t_0, when an intent is received to start Workflow A.1 at Site B (Considered as the Service Provider – SP) and Workflow A.2 at Site A (considered as the Cloud Provider - CP), with a strict latency requirement (e.g., 40ms). Software-Defined Network (SDN – considered as the Network Provider - NP) ensures the necessary network performance, with metrics such as frame per second, throughput, E2E latency, packet loss being continuously monitored. If latency or throughput violations occur, SDN dynamically adjusts the network bandwidth to maintain performance.

At time t_1, a new intent for Service B (inventory management) is received, requiring high-priority execution at Site A. To accommodate this, Workflow A.2 at Site A is interrupted due to insufficient resources. Site B is notified of the need to migrate Workflow A.2, and the site responds based on available resources. By time t_2, Workflow A.2 has successfully migrated to Site B, and its performance metrics are monitored and compared to the initial setup.

KPIs for this scenario include the downtime during the migration of Workflow A.2 from Site A to Site B (from the time the intent is received until the service is fully operational at Site B), as well as a comparison of throughput, E2E latency, and packet loss at both sites relative to the allocated compute and network resources. Additionally, power consumption is evaluated to assess the efficiency of running Service A.2 at both sites.

The set of interactions that compose the workflow to carry out the synergetic scenario have been illustrated in Figure 3-6. The chart represents three main cases: sub-scenario A, where a latency requirement is included, sub-scenario B where a dynamic regulation of bandwidth is requested and sub-scenario C: where the workflow stops due to insufficient resources.

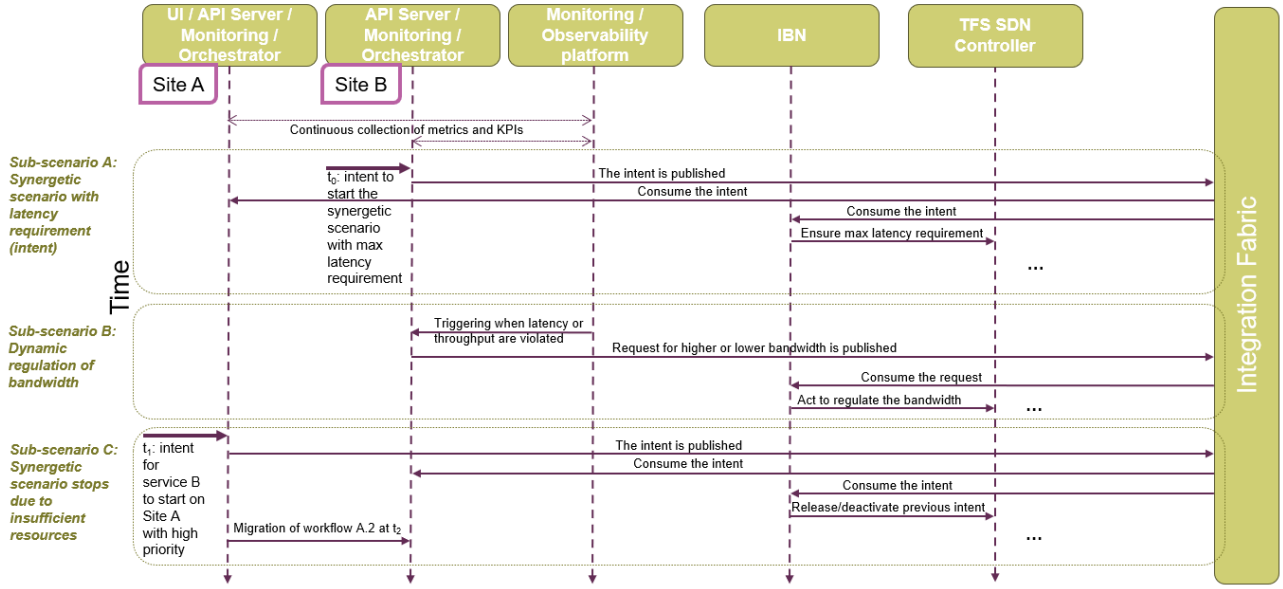


Figure 3-6 Message sequence chart of the synergetic scenario.

The process is demonstrated in Figure 3-6, where it is possible to appreciate that the Scenario 1 has been broken down into three main cases:

- Sub-scenario A - Latency requirement: The presented case starts when "Site B" generates the intent object including the desired requirements in terms of latency. This intent is forwarded directly with the usage of the Kafka real-time streaming data pipeline acting as the Integration Fabric (IF), remaining as an always on system which receives the intent asynchronously. To manage the incoming intent request, the Intent-Based Management (IBM) solution implements the following actions:
    1. Intent interpretation: The algorithm takes as input the received intent object and it implements natural language processing (NLP) techniques to extract the most relevant attributes from the received message.
    2. Database storage: The second action to carry out is to storage the processed information into a dataset to keep a record of all the historical requests.
    3. Feasibility check: After the main information of the intent has been extracted and stored, the algorithm then proceeds to ensure that the requirements included in the intent can be indeed accomplished. For that purpose, the algorithm performs a simplistic feasibility check by looking at the requested and the available resources in the network.
    4. Integration with TeraflowSDN (TFS): The next step consists of, first, composing the hierarchical object to perform the TFS request and to finally trigger the Layer 3 VPN request.
    5. Updating status through Kafka: To keep the synchronization with the other components, the IF distributes the status of the request to the other involved components in the same data-pipeline where the intent has been received.

  The previously mentioned set of actions have two main outcomes: First of all, the UI component refereed as "Site A" receives the update confirming the status of the sent intent and, second of all, the intent request gets consumed by the IBM solution and the intended changes in terms of latency requirements are forwarded to the SDN controller.
- Sub-scenario B - Dynamic regulation of bandwidth: A different case but involving the same functionality and components is presented in Sub-scenario B. The workflow starts now when the observability platform identifies that the latency or throughput requirements have been violated. This information is forwarded to Site B which composes an intent object to modify the latency and throughput requirements. Site B then composes the intent and forwards the information though the IF using a Kafka message. The IF the receives the intent asynchronously and performs the same set of actions described in the previous sub-scenario A: intent interpretation (i.e., NLP), database storage,

feasibility check, integration with TFS and updating status though Kafka. The difference between this sub-scenario B and the sub-scenario A relies on the content of the intent request and the TFS request which includes now dynamic changes in terms of latency and bandwidth.

- Sub-scenario C - Stopping due to insufficient resources: The third case is triggered by Site A, where some high priority changes are being requested. As a main result the intent data structure is being created and forwarded to the IF (i.e., Kafka) in the same manner than the previous sub-scenarios. The IF performs the same set of actions. Firstly, it implements NLP techniques to extract and process the most important information out of the intent. Secondly, it stores the received information. Third, a feasibility check is performed in order to identify that there are no available resources to comply with the desired request. As a main result, a TFS request is being generated requesting the available resources in the network, which are lower than the requested in the intent. Finally, the status gets updated through the IF with a Kafka message. Hence, the intent gets consumed, and the IBM solution forwards this information to the SDN controller. Additionally, Site B receives the corresponding status of the intent request asynchronously.

The implementation of the service migration process includes three main components: infrastructure, service and observability. The infrastructure includes two different sites, Site A and Site B as described above. Two main service components are considered for this scenario, 'Frame Sampler' which is deployed in Site B close to the robot that captures the frames and 'Object Detector' which is moved across the two sites depending on the availability of Site A's resources. Heterogeneous data is gathered from both components using a distributed monitoring data collection mechanism of the Monitoring and Telemetry Framework and stored at the deployed observability server.

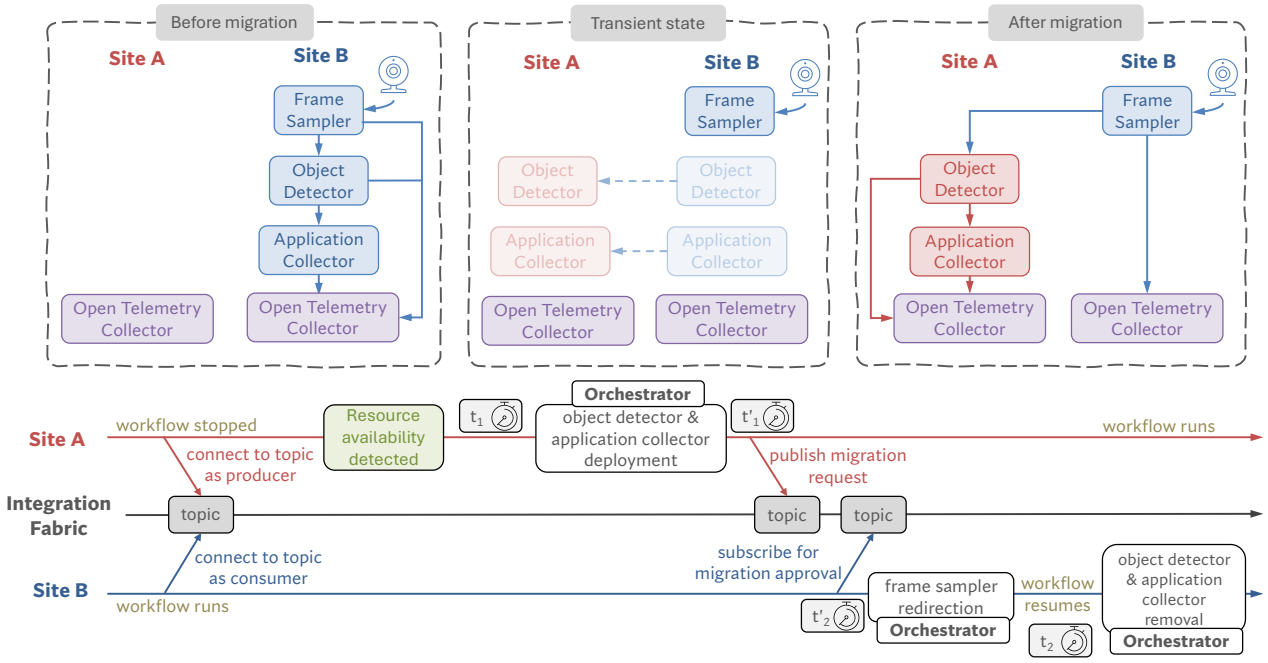


Figure 3-7 Sub-scenario A - service migration to Site A.

The intents coming from higher layers are translated into migration directives from Site A to Site B and vice versa. At Figure 3-7, the migration process of the 'Object Detector' component from Site B to Site A is described, taking into consideration the timestamps used as measurement points. After evaluation of the two active intents on Site A, the migration directive instructs the orchestrator of Site A to start the service and the orchestrator of Site B to drop the current deployment.

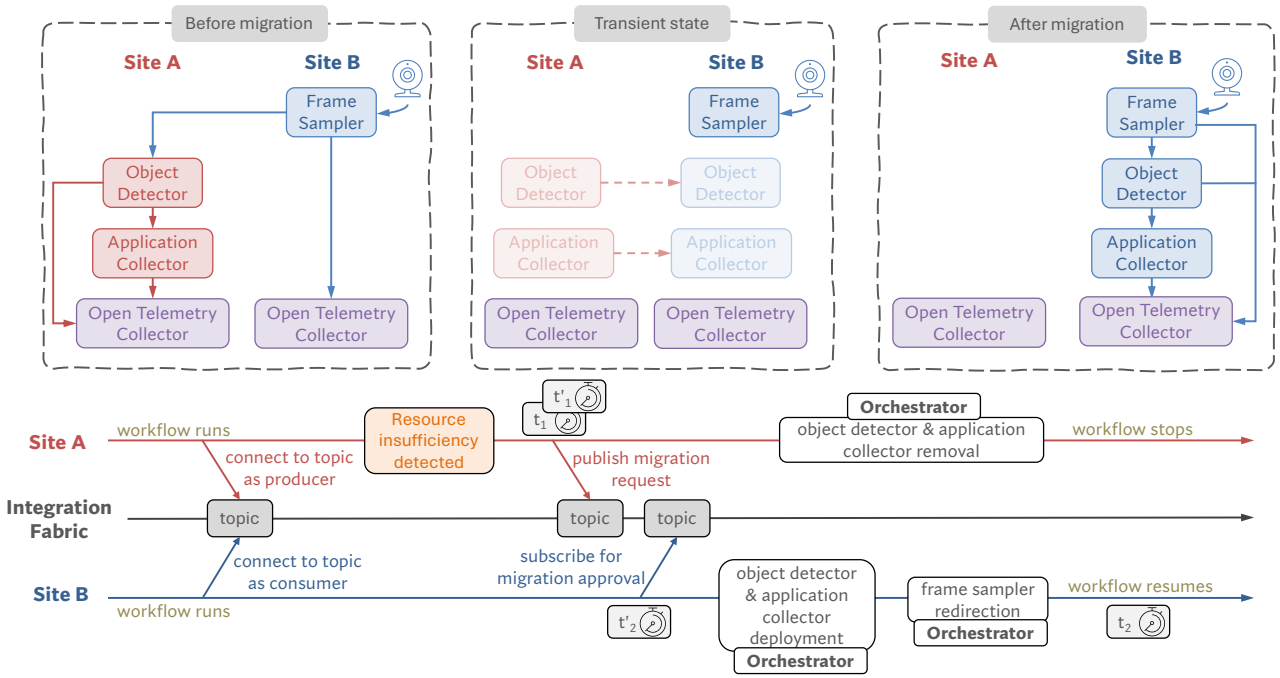


Figure 3-8 Sub-scenario C - service migration to Site B.

The opposite process is described in Figure 3-8, where the resources are assigned to another application running on Site A and thus, the directive to move the service from Site A to Site B is initiated.

3.3.6 Intent-based Cobot Service Provisioning and Migration with Integrated Closed Loop

This section describes the expansion of the Intent-Based service provisioning in a Cobot platform use case. On the one hand, several improvements related to the cobot platform are introduced. Including the use of a digital twin that enhances the surveillance capabilities of the platform and allows humans to interact with the platform and control the cobots. On the other hand, the Predictive Closed-Loop (CL) is presented, targeting significant performance improvements with respect to the earlier Reactive CL.

The VR digital twin system leverages 5G connectivity to enable real-time transmission of SLAM (Simultaneous Localization and Mapping) point cloud data, creating an interactive virtual representation of a physical environment. As seen in Figure 3-9, the architecture consists of a 5G Standalone (SA) network powered by Open5GS for core functionalities and a Nokia ASIR RAN for radio access. Two autonomous robots, Cobot 1 and Cobot 2, are equipped with LiDAR sensors for SLAM, video cameras for visual data capture, and Telewell 5G modems for high-speed data transmission. These robots act as mobile sensor nodes, continuously mapping their surroundings and transmitting data over 5G.

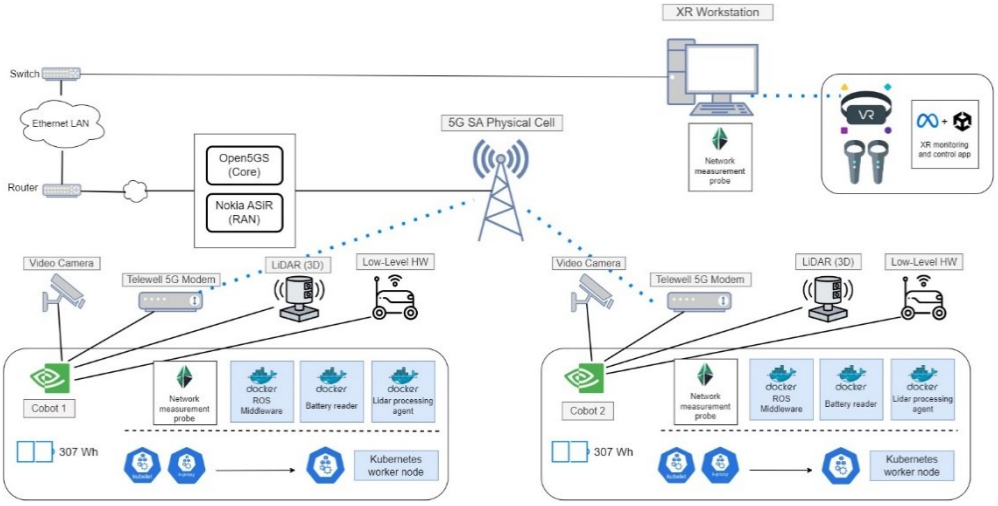


Figure 3-9 Updated measurement setup for the evaluation of the XR surveillance service.

Each robot features an onboard edge computing unit, NVIDIA Jetson-based, the Digital twin queries the cobots' ROS bridge to receive their LiDAR point cloud. The Kubernetes nodes manage multiple computational tasks, battery monitoring, and SLAM processing. The 5G SA network provides connectivity to the cobot workers. The system includes a network measurement probe that continuously monitors performance metrics like latency and packet loss to maintain a stable connection between the physical and virtual environments.

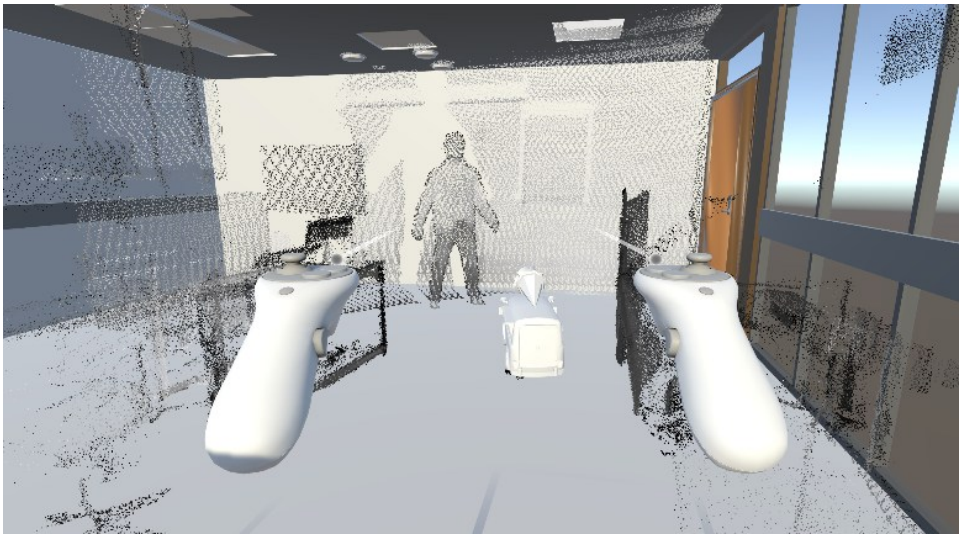


Figure 3-10 Digital twin model of the testing premises. LiDAR point cloud is overlayed.

On the visualization side, Figure 3-10 an XR workstation receives the SLAM-generated point cloud data and renders it into a digital twin, accessible via an XR headset. The digital twin allows operators to interact with the remote environment using XR controllers, providing immersive real-time feedback. This setup enables users to monitor robot movements, inspect mapped areas, and even control robotic actions remotely with minimal delay, enhancing situational awareness in industrial and research applications. The same view in the physical world is shown in Figure 3-11.

Figure 3-11 Picture of the measurement premises taken from the same perspective as Figure 3-10.

This 5G-powered digital twin system has numerous applications, including remote teleoperation, smart factory monitoring, and autonomous navigation. By utilizing 5G's high-speed connectivity and SLAM-based point cloud mapping, the system can dynamically update digital representations of real-world environments, making it ideal for industries requiring precise, real-time spatial awareness. The integration of XR technology further enhances user interaction, allowing for advanced monitoring and decision-making without physical presence in the operational area. In Section 4.4.11 evaluation results show the network performance of the platform.
As mentioned early in this subsection, different improvements and enhancement on the CL are later described in the following paragraphs, concerning the predictive CL used for the automatic cobot service migration demonstration. Briefly, from the original reactive CL logic, the enhancements consist on mainly adopting a predictive CL logic, with the support of AI reasoning. In particular, these improvements include an advanced monitoring platform as well as an AI-enabled analysis leveraging work done in WP6 [HEX225-D65]. A description of the advancements and the final integration are given in the following section.

A closed-loop, following guidelines from ETSI GS ZSM 009-1, consists of different functional components, each of them responsible for a specific role within the workflow of the cobot service migration. In brief, within the closed-loop platform, the monitoring, analysis, decision, and execution closed-loop functions are available for realising the logic within the real-time monitoring and control workflow. The sequence diagram for the runtime operations of the closed-loop functions is shown in Figure 3-12.

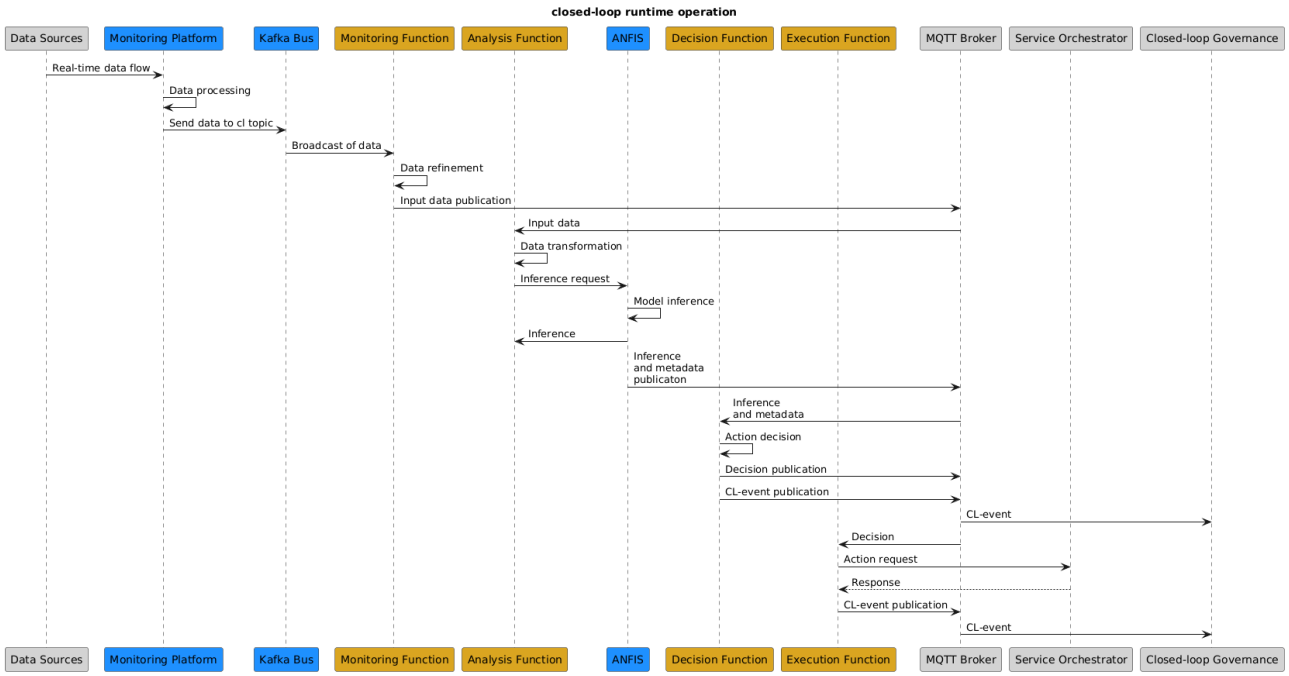


Figure 3-12 Closed-loop workflow for the cobot proof-of-concept. Gold boxes represent the closed-loop functions, blue boxes are extensions from the previous PoC, and grey boxes are supporting elements.

On top of the closed-loop service, the closed-loop governance coordinates the lifecycle of the service and composite functions, realising the overall logic for the automatically designed actions. Interactions between the functions and the governance take place through an MQTT messaging bus, which in this deployment includes start/stop, reconfiguration, and event tracking.

The demonstration is intended to address the management of vertical services on a given set of cloud resources provided by different providers across the cloud-continuum system. To achieve this, two dedicated orchestrators for the services and the resources that fall outside the scope of Hexa-X-II, are leveraged as described in [HEX224-D24]. The closed-loop governance block receives requests from the service orchestrator to instantiate and deploy a closed-loop attached to a vertical service. It then interacts with the resource manager to provision the necessary resources and deploy the composite closed-loop functions.
The closed-loop functions leverage AI to monitor service cobots and trigger when there is the need for a service application migration due to the variable nature of battery-operated cobots.

In terms of implementation, the cobots' data monitoring is realised by the monitoring platform, whose technical description has been reported in Deliverable D2.4 [HEX224-D24].
An enhanced AI-enabled closed-loop analysis function has been implemented, exploiting the Adaptive Neuro-Fuzzy Inference System (ANFIS) developed under WP6 and reported in D6.5 for analysing the input data provided by the monitoring function. The ANFIS uses application-level input data including the cobots battery levels, positions, and states to infer the optimal time to migrate a surveillance application from the active cobot to the non-active cobot. The ANFIS was trained using historical data from the cobots during normal operations. The run-time data is received by the analysis function via MQTT, transformed into an input tensor, and sent for inference to the ANFIS. The inference and metadata about measurements (timestamp, cobotID) are then published on MQTT to be consumed by the decision function. The closed-loop decision function calculates the inference weight and acts according to policies configured by the closed-loop governance. When the chosen decision involves an action to the system, the decision function publishes a message to be consumed by an execution function. As in the previous implementation, the execution function receives the message, and a request is made to a Service Orchestrator.

In the first stage of the integration between the closed-loop platform and the cobots service migration, the data monitoring, analysis, decision and eventually the execution were based on a reactive model, i.e. triggered by an event. Moreover, the analysis/decision process was based on a battery threshold only. To overcome this limit, some enhancements have been made to the closed-loop monitoring (i.e. the monitoring platform) and

closed-loop analysis/decision functions. Section 4.4.11 reports the results of the described enhancements made for the monitoring platform and closed-loop analysis/decision functions.

3.3.6.1 Components of the IBN-IME architecture

In order to achieve the accomplishment of having IBM within the System PoC, and based on the work done regarding the functional architecture of the Digital Service Manager-IME (DSM-IME) in [HEX225-D25], the components and their respective relationships that encompass the implemented IBM solution, called Intent-Based Network-Intent Management Entity (IBN-IME), are illustrated in Figure 3-13 and their functionalities are described below:

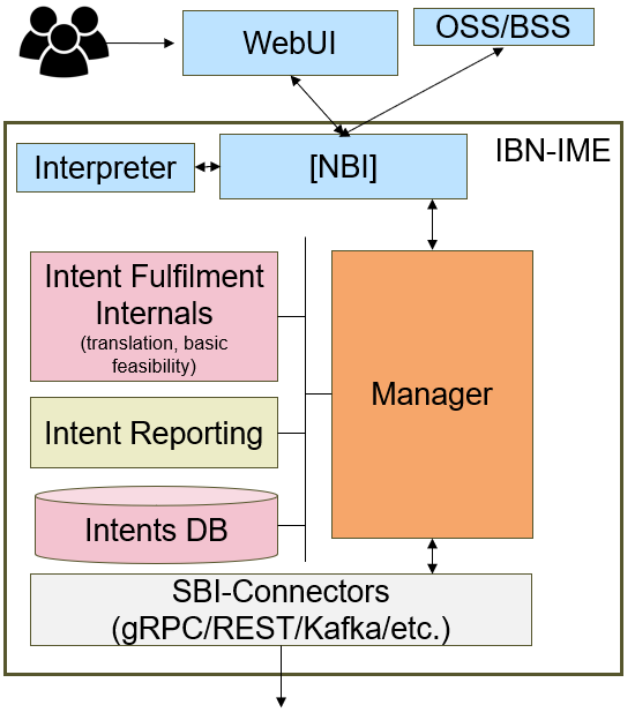


Figure 3-13 IBN-IME components of internal architecture.

- WebUI: The main goal of the component is to allow communication between the system and the user by providing a web interface where to request the desired intent.
- North-Bound Interface (NBI): It receives as inputs all the requests and its main role consists of filtering these requests towards its initial set of actions: to the Manager if the incoming request data already has the right format (i.e., 3GPP Information Model [28.312]) or to the Interpreter if the request data has a human format that needs to be processed.
- Interpreter: It plays the role of creating a 3GPP intent data model from a given request which is in a different format structure. To achieve this, some Natural Language Processing (NLP) techniques are implemented. As a main result, the complete hierarchical data format including all the expectations and contexts are included
- Manager: It is the entity that controls the phases of the internal workflow for each intent request to ensure that its lifecycle is indeed being accomplished.
- Intent Fulfilment: It performs a set of actions that are required to achieve complete intent management. Some examples of these actions are to perform a feasibility check or the identification of executable actions to achieve the intent.
- Intent reporting: Its main role consists of updating the status of the request. To do so, an object with a specific data structure [28.312] is created and stored in the database.
- Intent DB: The dataset that stores the intent data objects.
- South-Bound Interface (SBI)-Connectors: This module identifies and adapts the incoming information to the right data model expected by the selected destination.

Is it key to remark that the implemented IBN-IME solution has been used in two of the System PoC scenarios: the "*E2E implementation scenario: Zero-touch Cobot-based Video Surveillance*" and the "*Synergetic Monitoring and Orchestration*", proving its adaptability to different scenarios. This is proven based on the fact that the presented IBN-IME solution performs a similar functionality to the PoC in subsection 3.3.5 and the results in subsection 4.4.12.2. Hence, the results in terms of execution time could be extrapolated to the presented use case. Further details about the solution can be found in [HEX224-D23, HEX224-D24, HEX225-D25].

3.3.6.2 Intent-based service provisioning with Integration fabric mediation

An external IBN-IME can interact with the Service orchestrator though the MCE (named Integration fabric) and an external wrapper component named Integration fabric Adapter. It handles the requests coming from the IBN-IME through the Integration fabric and maps these requests to the service orchestrator.

Figure 3-14 shows a high-level workflow related to the intent-based service provisioning with the Integration fabric mediation.

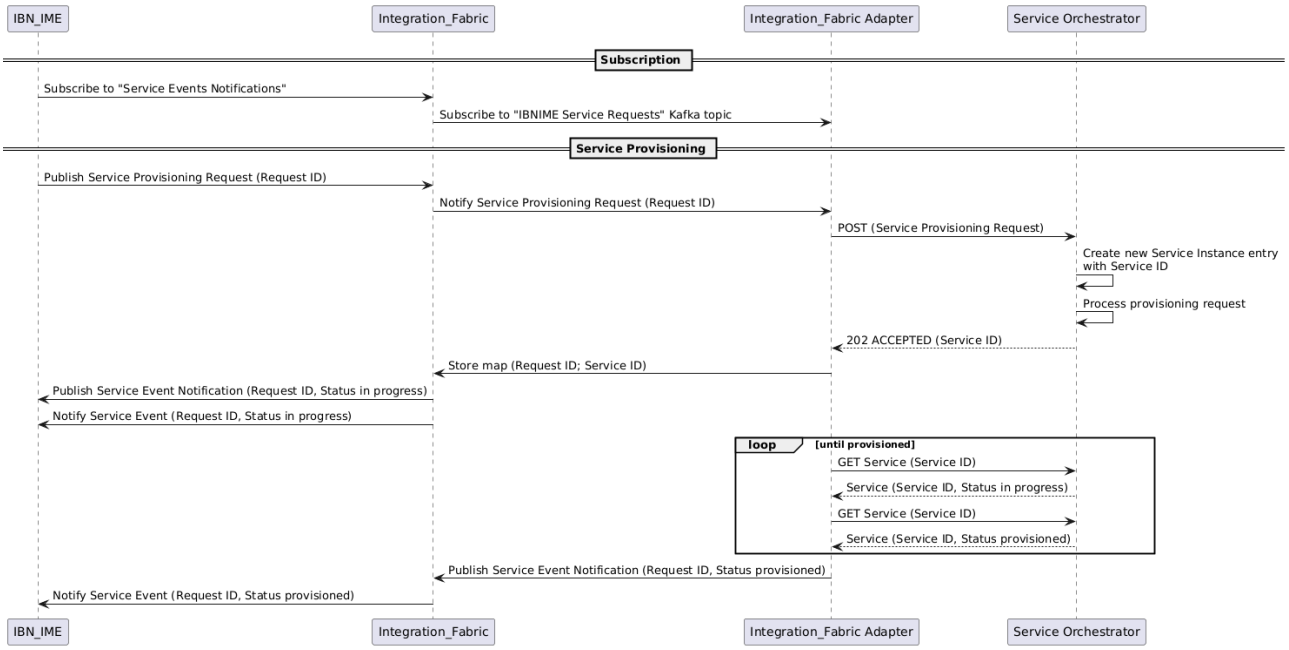


Figure 3-14 High-level workflow of intent-based service provisioning with Integration fabric mediation.

The workflow is composed of two stages: subscription and service provisioning. The subscription is executed once, while the service provisioning stage is executed as many times as the IBN-IME requests one or more services to be provisioned.

In the subscription stage, the Integration fabric Adapter and the IBN-IME communicate by subscribing to two different topics, relying on the Integration fabric. The first topic the Integration fabric adapter subscribes to is for receiving notifications related to the service provisioning request, while the second topic the IBN-IME subscribes to is about the service event notifications.

In the Service provisioning stage, the IBN-IME requests to provision a service, publishing a message to the Integration fabric Adapter topic is subscribed to. Indeed, the Integration fabric adapter receives the corresponding notification and starts the workflow for provisioning the service. During the workflow execution, the different statuses of the service provisioning (status in progress and status provisioned) are notified to the IBN-IME through the Integration fabric. Eventually, the IBN-IME will be aware of the statuses of the different requests. Moreover, the Integration fabric Adapter can store the mapping between the external requests, i.e. from the IBN-IME, and the requests to the Service Orchestrator. Section 4.4.13provides the results about the Integration fabric Adapter implementation, integration and validation against the integration

fabric. Moreover, in terms of results it is also provided an estimate about the overhead introduced by the interaction between the Integration fabric Adapter and the Integration Fabric.

3.4 6G Radio Components in PoCs

This chapter provides a brief description of the 6G Radio Components PoCs, presented in detail in [HEX225-D45], in order then to associate the objectives/focus of each one of them with the E2E system and/or the System-PoC's scope.

3.4.1 6G based sensing algorithms and concepts with real-time performance

A joint communication and sensing testbed has been implemented using Sivers 60 GHz radio modules and a Xilinx FPGA board to explore the feasibility of leveraging telecom signals for simultaneous communication and sensing. Operating in a bi-static setup, the system utilises 5G-NR adjusted waveforms—developed by Qamcom—for both transmission and reception, enabling robust sensing capabilities. Extensive FPGA development, along with advanced software and signal processing algorithm design, has been carried out to enhance target detection accuracy while minimising processing latency. Key techniques include background subtraction to improve target discrimination and reduce false detections. The testbed successfully detects and tracks human movement in an office environment, projecting positions onto a 2D map in real time. Through optimised processing, it achieves a detection rate exceeding 90% at 10 frames per second.

The PoC is a bi-static joint communication and sensing testbed with these specifications:

- Radio transmitter and receivers from Sivers in 60 GHz and 800 MHz of bandwidth
- 50 different transmit beams and 56 different receive beams are used in the radios
- Xilinx FPGA board for signal processing
- Python codes for post processing and data representation
- Adjusted 5G-NR waveform using CP-OFDM signals, 64-QAM modulation, and subframes including SSS, PSS and PDSCH formats

3.4.1.1 Alignment with the E2E system

The PoC is aligned with the Collaborative robots UC in the direction of:

- Enhancing the accuracy of target detection
- Assisting the collaborative robots by providing sensing and detection of not-connected targets (people in the industrial environments)

3.4.1.2 Challenges identified (system-level) and next steps

These challenges are identified during the project:

- OTA synchronization is very challenging for sensing applications. The sync algorithm development for sensing should be much more accurate in comparison with sync for telecom applications. New methods and algorithms are needed to be developed specifically when sync should be done OTA and without access to GPS
- Background subtraction in a dense office environment with lots of reflections can be challenging and requires more advanced methods. As future works, AI/ML tools can be applied to investigate the performance of background subtraction in comparison with model-based signal processing methods we used in this project.

3.4.2 AI-Native Air Interface

In this PoC, the target is to demonstrate the practical feasibility and performance gain of and end-to-end learned transmission link. The setup was described in detail in [HEX224-D43], and essentially it consists of an OFDM-transmitter utilizing a machine learned constellation, which is transmitting data to an ML-based receiver (DeepRx). The constellation and DeepRx are learned jointly with simulated data to communicate without any DMRS pilots, which results in increased spectral efficiency. The target of the PoC is to validate that this results in higher throughput also with real equipment and with real channels.

3.4.2.1 Alignment with the E2E system

The PoC system was measured in an indoor scenario, where a throughput gain of 19.2% was achieved. This demonstrates that the developed system can increase the spectral efficiency in a real-world scenario. Moreover, since the considered measurement location was an indoor lab with multiple sources of reflection, the observed throughput gain can likely be achieved also in many industrial environments. This means that this technology can be applied in many of the key 6G use cases to enhance the spectral efficiency.

3.4.2.2 Challenges identified (system-level) and next steps

One aspect to note in this particular PoC is that the measurements were carried out for an individual link. An important future topic is to evaluate the gains in a more realistic setting, where adjacent cells produce interference.

3.4.3 ML-based channel state feedback compression in a multi-vendor scenario

The target of this PoC is to demonstrate that cooperative AI-based techniques can improve the spectral efficiency and accuracy of channel state information feedback compared to legacy CSI schemes. This PoC focuses on sequential and separate AI-based techniques that do not require sharing proprietary AI models. The setup, detailed in [Section 6.2.1, HEX224-D43] involves an AI/ML algorithm running on the device to encode/compress the CSI for transmission. At the gNB, a reciprocal AI-based technique decompresses the CSI. Together, those two AI algorithms can improve CSI resolution for a given number of bits, or equivalently, reduce overhead for a fixed CSI resolution.

3.4.3.1 Alignment with the E2E system

In the context of cooperating autonomous robots, especially in heterogeneous, multi-vendor environments, the use of ML-based channel state information (CSI) feedback becomes particularly valuable. These robots often operate in dynamic, interference-prone industrial settings where standardized CSI feedback mechanisms may fall short due to varying hardware capabilities or protocol implementations. The ML-based CSI feedback approach enables adaptive, low-latency coordination between robots by improving beamforming precision while maintaining interoperability across diverse radio stacks. This is critical for ensuring seamless cooperation and task synchronization, even when robots from different vendors share the same wireless infrastructure.

3.4.3.2 Challenges identified (system-level) and next steps

Over-the-air data in low mobility and outdoor scenarios have been collected to train the ML models and evaluate performance. Further studies will focus on evaluating performance is over-the-air scenarios to ensure robustness and adaptability of the ML-based CSF approach.

3.4.4 Flexible modulation and transceiver design

The flexible transceiver system PoC is built to create a platform for experimental verification of mobile communication algorithms, concept and research. To allow for dynamic reconfiguration of the system, it is implemented on software-defined radio (SDR) devices, commonly used in research. This flexibility allows for testing of complex dynamic use-cases that exceed the limitations of common static testbed configurations.

Core components of the system are the USRP X310 SDRs, which are used for generation and reception of radio signals. The NI PXI, running a LabVIEW instance, acts as a controller to enable real-time reconfiguration of system parameters. In addition, the flexible transceiver system can be used by a host computer that can configure the system via an API. Interaction with the flexible transceiver via API is intended to allow for more user-friendliness of the system. For the RF frontend, different configurations are applicable, including beamforming networks, phased arrays, analogue multicarrier processing circuits, upconverters and transmission at higher frequency (mmWave) with dedicated antennas. In the flexible transceiver system, the

SDRs are co-located and synchronization between them is maintained through a common 10 MHz reference and 1-PPS trigger signal.

This flexible transceiver system allows for dynamic reconfiguration of RF parameters during runtime such as carrier frequency, bandwidth, gain, modulation schemes and waveforms. The transmission of IQ samples is handled via the API. This provides a comforting abstraction layer for users of the system, who can execute experiments with API calls in Python or MATLAB programming languages. The fully independent control over individual TX/RX chains provided by the flexible transceiver enables applications such as digital beamforming or carrier aggregation.

The first prototype is operational with essential functionality for waveform transmission and reception. While basic functionalities are implemented, dynamic parameter adjustment by remote procedure calls (RPCs) as part of the API is still in refinement. Most recent works on the flexible transceiver system focus on implementation of real-time phase calibration. The phase coherence archived allows for applications such as angle estimation. The PoC and specifically the feasibility of the implemented phase calibration procedure, was demonstrated and reported in [CNF25].

This flexible transceiver system provides a scalable and dynamically reconfigurable experimentation platform for mobile communication researchers. It facilitates testing and performance evaluation for mobile communication scenarios. A more detailed description of the flexible transceiver system can be found in [HEX2-BCN+24].

3.4.4.1 Alignment with the E2E system

The flexibility of the proposed PoC enables the configuration of hardware and software parameters, allowing for the analysis of customized radio architectures suited to specific use cases. While the proposed flexible transceiver system is not explicitly designed for the collaborative robot use case, it can be used for bi- and mono-static radar applications, to allow for passive object detection in industrial environments such as factory halls. As an E2E system, the alignment with the use case can be achieved through a tailored experimental setup, configured via the provided API interface. Since the system is designed as a flexible general-purpose platform for validating diverse communication and sensing applications, it does not inherently provide KPI metrics for passive object detection. The achievable sensing performance in this use case is highly dependent on the experimenter's configuration and signal processing algorithms on the host side, which are not intrinsic to the PoC itself.

3.4.4.2 Challenges identified (system-level) and next steps

The implementation of dynamic parameter adjustment and documentation of the API is still in process. A challenge remains the lack of clear metrics on judging the accuracy of the implemented phase calibration mechanism. The current phase calibration is dependent on the RF frontend configuration and assumes a previously known reference position. To the best of available knowledge, no known methods exist to circumvent this limitation, as it is inherent to any testbed or experimental setup that relies on phase calibration of RF components. As a next step, the flexible transceiver system is planned to be implemented on the USRP X410, which supports increased bandwidth and a higher number of TX chains per SDR device.

3.4.5 Radio propagation measurements to collect data for radio channel modelling

This PoC comprises a targeted measurement campaign in cobot-equipped factory environments to harvest radio-channel characteristics under realistic operational conditions. The industrial setting—marked by metallic, reflective surfaces and rapidly moving obstacles—induces pronounced multipath richness and intermittent blockages. Systematic sub-THz propagation measurements, as detailed in [Section 3.3, HEX225-D45], yield path-loss parameters and delay-spread profiles tailored to both factory halls and warehouse layouts. These empirically derived channel statistics form the basis of site-specific models that drive link-level simulations, enabling precise forecasts of throughput and latency. In addition, the same dataset supports virtual sensing experiments, essential for cobot applications requiring environmental awareness and adaptive control.

Finally, the measurement results feed into radio-coverage mapping, highlighting signal-strength variations and dead zones to guide optimal placement of access points and related infrastructure, thus securing robust, high-capacity connectivity throughout the facility.

3.4.5.1 Alignment with the E2E system

The radio-propagation measurement PoC integrates directly into the E2E 6G blueprint by feeding empirical channel metrics into the infrastructure layer instantiating sub-THz path-loss and delay-spread profiles within both RAN and X-haul components. These derived parameters then enhance beyond communication functions in the network functions layer like the SeMF and AIaaSF, empowering adaptive radio control and predictive link management. At the application enablement platform tier, coverage maps and site-specific channel models are exposed via standardized APIs, facilitating dynamic service composition and multi-stakeholder orchestration. Finally, these measurement outputs emphasise pervasive capabilities like intent-based management, closed-loop resilience controls, and AI-orchestration, embedding factory-tailored radio insights into the modular, scalable, and secure E2E 6G system.

3.4.5.2 Challenges identified (system-level) and next steps

System-level challenges include accurately capturing the rapid multipath dynamics induced by reflective surfaces and moving objects, ensuring sub-THz transceiver calibration and synchronization across heterogeneous factory layouts, generalizing site-specific path-loss and delay-spread profiles beyond the initially measured halls and warehouses, and embedding coverage maps and channel models into real-time planning and orchestration workflows. To address these, upcoming steps are to broaden measurements across varied geometries and mobility patterns, automate coverage-map generation for on-the-fly dead-zone detection, refine parameter-extraction methods under dynamic obstruction scenarios, integrate the updated channel models into link-level and sensing simulators within the E2E 6G framework and establish a feedback loop between live network-performance metrics and channel-model refinements.

3.5 6G Devices Components in PoCs

3.5.1 Crowd-detectable zero-energy devices

The Component PoC#C.3 (a) demonstrates how backscatter-based zero-energy Ambient IoT (AIoT) devices can be integrated into mobile communication systems without requiring hardware modifications to base stations (BS) or user equipment (UE). In the proposed system, a backscatter device (BD), powered by ambient light, modulates signals transmitted by the mobile communication system. The BD employs frequency shift keying (FSK) to encode its information into the channel.

During channel estimation, the BD appears as an additional multipath component. However, due to the frequency shift introduced by the BD, its signal can be distinguished from natural multipath effects in the Doppler frequency domain. This concept is further elaborated in D5.5 [HEX225-D55], Sections 6.1.2 and 7.

The PoC implementation was based on an LTE system. In the downlink direction, the BS (eNodeB) transmitted Cell-Specific Reference Signals (CRS). A software-defined radio (SDR)-based UE receiver performed channel estimation using the received CRS. The FSK-modulated BD signal was detected after filtering out natural multipath components.

For the uplink direction, a commercial mobile phone transmitted Sounding Reference Signals (SRS). An SDR-based eNodeB receiver estimated the channel and similarly detected the BD signal by distinguishing it from natural multipath effects. The working principle of the PoC is illustrated in Figure 3-15. The PoC evaluation results are presented in Section 4.4.4.

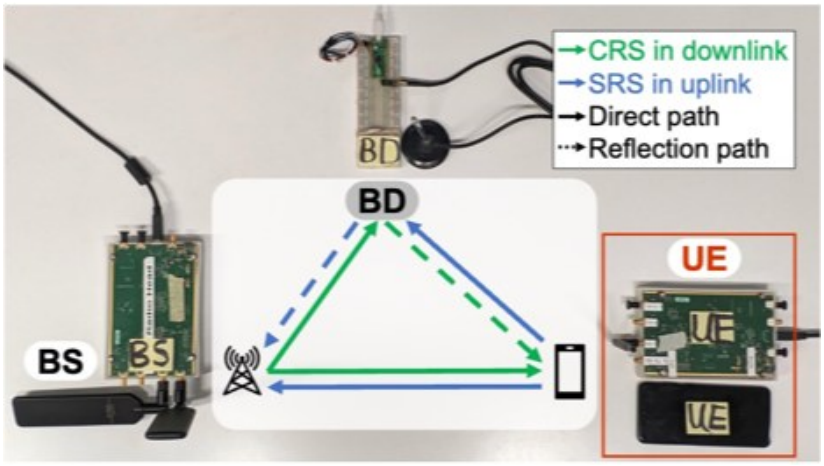


Figure 3-15 Operation principle of the crowd-detectable zero-energy devices integrated to LTE network.

3.5.1.1 Alignment with the E2E system

Different deployment strategies for backscatter-based AIoT devices are discussed in D5.5 [HEX225-D55], Section 6.1.4. This PoC aligns with the bi-static configuration, where the signal source and receiver are spatially separated.

In the current 3GPP AIoT work [38.848], the focus is on systems that generate a dedicated continuous carrier wave to interrogate backscatter devices. In contrast, this PoC demonstrates that a low-data-rate AIoT service can be provided using standard cellular signals, eliminating the need for a separate carrier wave. This results in higher spectral and energy efficiency. In the proposed setup, the BD is illuminated by existing mobile communication system signals and shares the spectrum with the mobile network. The system has been successfully demonstrated in LTE network as described in D5.5 [HEX225-D55], Section 7.

The primary application of crowd-detectable zero-energy devices in collaborative robotics is proximity-based indoor localization. A robot can gain a rough estimate of its location whenever it comes into proximity with a BD device that has a fixed and known position. This approach is conceptually similar to Bluetooth Low Energy (BLE) beacon-based positioning, but instead of a BLE beacon transmitter, it relies on an ultra-low-power BD. A key advantage of this method is that a cellular modem or base station can determine proximity-based location as a byproduct of communication, eliminating the need for additional power consumption. Beyond serving as positioning anchors, BD devices can also be attached to objects with which collaborative robots interact. This capability allows robots to identify objects, functioning similarly to RFID technology, further enhancing their ability to operate autonomously in dynamic environments.

3.5.1.2 Challenges identified (system-level) and next steps

Several challenges and open issues have been identified in integrating crowd-detectable zero-energy devices into cellular systems. One significant limitation is the BD reading range, which is constrained by current implementation methods. To improve detection range, Forward Error Correction (FEC) coding could be applied, with convolutional codes combined with Cyclic Redundancy Check (CRC) emerging as a particularly attractive approach. Further discussion on this topic can be found in D5.5 [HEX225-D55], Sections 6.1.4 and 7.1.1.

Although standard UE modems could potentially be used to demodulate BD signals, firmware updates would be required to accommodate this functionality. Specifically, the UE must remain active while reading BD signals, even when there is no active traffic, to prevent it from entering sleep mode. Additionally, the UE's automatic gain control (AGC) should either maintain stable levels while processing BD signals or communicate any level changes to the BD receiver to ensure accurate decoding.

Another challenge arises from differences in reference signal availability between LTE and 5G NR. In LTE, CRS are continuously transmitted, making them suitable for illuminating backscatter devices. However, in 5G NR, there are fewer readily available reference signals, which poses a challenge for BD integration. This challenge is likely to persist also in 6G. One potential solution is to use Positioning Reference Signals (PRS)

for BD illumination, while another approach would involve defining a new reference signal type specifically designed to support BD-modulated message reception.

To facilitate BD message reception, a dedicated signaling mechanism would be beneficial to instruct the UE to begin demodulating BD signals. Furthermore, the integration of BD technology may necessitate the introduction of a new core network function to handle BD message reception. The need for this entity would largely depend on the intended application. A particularly promising use case is indoor proximity-based localization, where the system detects when a UE is in close proximity to a BD with a known, fixed location. In such scenarios, the Location Management Function (LMF) should be updated accordingly to reflect the UE's position.

Finally, synchronization challenges must be carefully considered to mitigate potential interference. In the PoCs, the BD was not synchronized with the mobile communication system, which could lead to inter-carrier interference (ICI) if the BD symbol changes mid-OFDM symbol. While the impact of this interference is expected to be minimal due to the characteristics of BD signals, it could still degrade the performance of legacy receivers. A thorough investigation is needed to understand the extent of this issue and evaluate possible trade-offs to ensure seamless integration without compromising existing network performance.

3.5.2 Energy harvesting IoT proximity devices

This subsection introduces the energy harvesting (EH) device prototype developed for the cobot-powered inventory management warehouse PoC described in Section 3.3.1, building on the work reported in [Section 6.1.2, HEX225-D55]. The EH IoT device, shown in Figure 3-16, has been specifically designed and integrated into the cobot-powered inventory management warehouse PoC. Equipped with proximity sensor, the device detects the occupancy of warehouse selves. When the sensor identifies the presence of a box or item, it triggers an alert, prompting the cobot to be sent for further inspection.

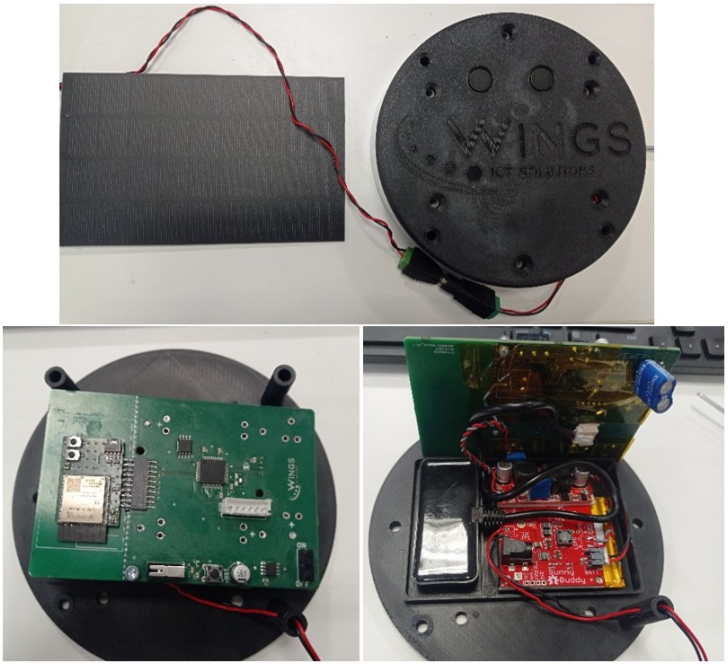


Figure 3-16 Warehouse inventory management scenario new EH devices for advanced inventory operations based on real-time occupancy information.

The device consists of a solar panel (9 V, 3 W) which is theoretically capable of producing up to ~ 300 mA under ideal sunlight where in real-world conditions yields less than ~ 150 mA. Since it is aimed to work even indoor with unsure light conditions at sometimes, it is equipped with a LiPo battery (2400 mAh) which stores energy for operation and a sunny buddy solar charger for managing LiPo battery charging. Additionally, a DC-DC converter (4.2 V → 3.3 V) is incorporated adding ~ 9 mA during sleep mode and normal modes.

As already mentioned, the device is equipped with a proximity sensor DFRobot A02YYUW ultrasonic sensor, which measures distance from 3 cm to 400 cm (UART interface). The sensor data processing and control is

handled by a STM32 microcontroller. The communication module used for supporting wireless data transmission is the ESP32 C6 Wi-Fi module. Finally, a weatherproof enclosure (IP67) is utilised for the protection of electronics from the water, dust and harsh conditions. Table 3-1 shows the power production and consumption analysis of the various components described above.

Table 3-1 Power analysis of the various EH device's components.

Component	Power Analysis	Time Active per Cycle	Notes
Solar Panel (9 V, 3 W)	Produces up to 300 mA (theoretical)	Continuous (daytime)	Real output <150 mA typically
Solar Panel Under Lamp	Produces ~27-30 mA +	Continuous (lamp on)	Lamp above the panel at 10 cm
Sunny Buddy Charger	Consumes 100 µA -	Continuous	Manages LiPo charging
2400 mAh LiPo Battery	N/A	Continuous	Stores energy
DC-DC Converter (4.2 V→3.3 V)	Consumes ~9 mA in sleep mode -	Continuous	Dominates sleep consumption
STM32 MCU	Consumes ~30 mA Consumes (Active), - <0.5 mA (Sleep) -	2 s every 30 s cycle 28 s every 30 s cycle	Data acquisition & control
ESP Wi-Fi Module	Consumes ~50 mA - (Transmission)	~ 45 s-1 m every change of state	High consumption for sending data
Ultrasonic Sensor (A02YYUW)	Consumes ~30 mA with STM during measurement	2 s every 30 s	Distance measurement

3.5.2.1 Alignment with E2E system

The EH device developed for the cobot-powered inventory management warehouse PoC is a crucial component in aligning with the E2E system goals. As outlined above, the IoT device integrates a proximity sensor that enables real-time detection of warehouse shelf occupancy, prompting the cobots to inspect and manage inventory based on these inputs. This device, which operates autonomously within the warehouse, plays a vital role in optimising the efficiency of the cobot system.

The EH device's ability to provide real-time, low-power sensor data while minimising energy consumption enhances the collaborative operation of the cobots, ensuring they can perform their tasks effectively without compromising their operational duration.

This device's integration and energy-efficient design are fundamental to the overall alignment of the PoC with the larger E2E system, where optimising energy usage and ensuring seamless communication between devices are key objectives for successful implementation.

3.5.2.2 Challenges identified (system-level) and next steps

The main challenges identified are primarily related to the limited distance between the lamp and the solar panel, which impacts its performance. In typical warehouse environments, lighting conditions are often insufficient to provide optimal energy harvesting. To address this, further testing will be conducted with different panels, batteries, and chargers to ensure the system can perform effectively under standard warehouse lighting conditions. Additionally, reducing data transmission and implementing a flexible topology node, such

as a drone, to provide connectivity for data transmission will help minimise the power consumption of the device. The next steps will focus on further exploring the overall energy consumption of the system, ensuring that it operates efficiently while maintaining reliable performance in real-world warehouse environments.

3.5.3 End-to-End Extended Reality

This PoC aims to demonstrate the potential benefits of Scalable, or distributed, Compute, where heavy processing tasks, such as augmented reality (AR) video rendering, are offloaded to a dedicated remote server. This technique provides power saving gain at the device. Such remote compute solution is downlink heavy and prone to 5G link quality deterioration. Therefore, there is a need to dynamically adjust the encoding rate according to the channel requirements in terms of throughput and latency. This PoC leverages the low-latency, low-loss scalable throughput (L4S) feature as our media rate control framework, improving user QoE, compared to on-device (local) compute.

A testbed, illustrated in Figure 3-17, was implemented for this PoC where the server renders the video frames, while the Mobile Test Platform (MTP) runs the rate-control algorithm and provides the desired target bitrate as feedback to the server. An evaluation of the nefits of exploiting L4S for adaptive rate control and improved user experience was conducted in a lab testing environment. In the considered evaluation scenario, the RF condition is approximately -105 dBm. The load starts with 3 users in the cell pushing full buffer traffic for 60 seconds to generate network load then stops.

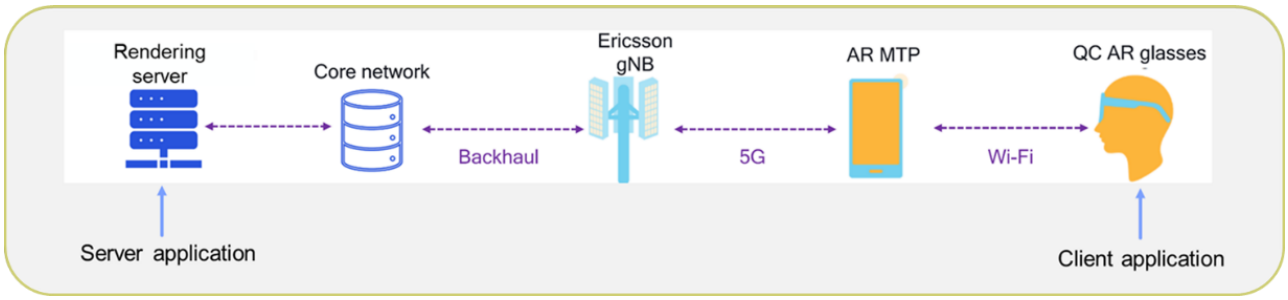


Figure 3-17 The testbed comprises of a user with an AR glass device and a phone running the client application, and with a 5G connection to an Ericsson gNB and a PC running the server-side application.

Without XR on-device adaptive control (baseline), the server application streams video frames with a fixed data rate. As the NW load is introduced and creates congestion over the 5G channel, the video frame latency increases, which in turns triggers frame loss for the AR application. This materializes a stall in the video stream, leading to a considerable degradation in user quality of experience.

With XR Distributed Compute, the application can perform adaptive rate control based on the NW load by leveraging congestion feedback from the network as part of the L4S framework to ensure minimal queueing latency at the gNB, hence reducing the number of lost frames and greatly improving the user QoE.

The results are reported in Section 4.4.6.

3.5.3.1 Alignment with the E2E system

In modern factory and warehouse environment, cooperating autonomous robots are increasingly used for various complex tasks. These battery-powered devices communicate with one another and other machines for safety and cooperation. They rely on sensors, such as cameras and LiDARs, to enable computer vision algorithms at regular intervals for navigation and positioning tasks. These sensors are also essential for cobots applications, such as object detection and object avoidance, as they operate alongside humans, other robots, and equipment.

One challenge for these devices is the processing of computer vision algorithms locally while maintaining a low power consumption. This is crucial to limit the battery size and provide compact form factor and ensure optimal operation time. Distributed Compute offers a natural solution to address those challenges by offloading heavy processing tasks, such as vision-based processes, to a dedicated remote server. The Cobot uploads sensors inputs and, in return, receives the corresponding positioning, object detection information (or other

Cobots required inputs). This approach reduces the power usage at the device while still providing the necessary computer vision results for the cobots operation.

3.5.3.2 Challenges identified (system-level) and next steps

Next steps include the selection and capability of offloading specific processing components. For example, computing depth map locally and offloading the 3D scene rendering. Another challenge can be that rate control can only decrease the bitrate up to so some level without considerably impacting the user experience and serving many such users can be challenging in terms or wireless resource management. A next step is dynamic spatial compute where users in poor conditions use an on-device feature to transition to local compute mode, improving their user experience while freeing up wireless resources than can be used to serve additional UEs, thereby increasing the network capacity.

4 System- and Component-PoC #C validation and evaluation results

This section provides an overview of key performance and value-related indicators, presenting the results from a selected set of previous metrics and their alignment with respective service requirements. Since System-PoC #C addresses evolved versions of the management and orchestration aspects found in System-PoC #A, and 6G network beyond communication and intent-based mechanisms aspects found in System-PoC #B, certain KPIs related to these aspects remain critical. Simultaneously, new KPIs must be introduced to address the radio and device related aspects. The following sections include: (i) indicators for evaluating the performance of the network, application and infrastructure layer components, as well as various management and orchestration mechanisms; (ii) indicators for assessing environmental, social, and economic sustainability aspects; (iii) summary of Systems PoC #A and #B results; (iv) the final evaluation results of System PoC #C; and (v) the impact of the Hexa-X-II PoC-driven validation activities.

4.1 KPIs associated to System-PoC #C

The elements and components of System-PoC #C, as discussed in subsection 3.2, correspond to various KPI requirements across the network, compute, application, and infrastructure layers. A comprehensive set of technical KPIs related to the implementation of the System-PoCs is outlined in D2.2 [HEX223-D22] and in D2.4 [HEX223-D24]. Among these, metrics such as bitrate and E2E latency are particularly important due to their significant impact on compute-intensive tasks, including inventory audits and real-time computer vision operations.

For example, swarm management operations, which include data processing, path planning, and navigation, also rely heavily on these metrics. Both upload and download bitrate, network latency, and edge/remote server processing latency are all critical factors. Many applications involve robotic platform teleoperation, which requires ultra-low latency between the remote user (tele-operator) and the robotic platforms in the field. As such, uplink and downlink network latency are key measurements. Teleoperation also includes computer vision-related tasks, such as compressing real-time high-definition video streams. This process is influenced by on-board processing latency, upload bitrate, network latency for transmitting compressed video, and edge/remote server processing latency for inferencing and decision-making. Furthermore, the response (or enforcement) to robotic devices, which is affected by download bitrate and network latency, is essential. Finally, service availability, including minimising downtime, is of utmost importance.

4.2 Social, environmental, and economic sustainability aspects

System-PoC #C in Hexa-X-II explores the feasibility of achieving social, environmental and economic sustainability -besides performance- within the framework of 6G radio and device components. Trustworthiness, as part of the social sustainability dimension encompasses the network's reliability, security, and overall integrity, and its provided services. The incorporation of advanced radio technologies, real-time device feedback, and autonomous decision-making processes ensures the system’s resilience, thereby reinforcing trustworthiness in dynamic and unpredictable environments.

In terms of environmental sustainability, System-PoC #C addresses both the operational efficiency of the application and the performance of the underlying networking and device infrastructure in the 6G ecosystem. By utilising EH devices and ZEDs, the system extends operational lifetime, minimising energy consumption and contributing to the broader sustainability goals. The intelligent orchestration of network resources across the 6G continuum, coupled with adaptive radio features, enhances resource efficiency, ensuring that the system operates sustainably even in environments with limited resources. This is achieved through dynamic network management policies that optimise power consumption and overall network resource usage.

Economic sustainability in System-PoC #C is focused on minimising operational costs while maintaining service reliability. The resilience built into the system, powered by advanced radio and device capabilities, such as AI-native air interface, ML-based channel state feedback compression, flexible modulation and transceiver design, energy harvesting and zero energy devices, etc., reduces downtime, ensuring consistent and dependable service. The system’s energy efficiency not only lowers operational expenses (OPEX) but also

reduces capital expenditures (CAPEX) through scalable and flexible infrastructure. Together, these aspects contribute to a solid economic foundation, ensuring the long-term sustainability, cost-effectiveness, and competitiveness of the 6G ecosystem.

4.3 Summary of key results from Systems PoC #A and #B

Concerning the previous work that took place reporting System PoCs' design specification and evaluation activities:

D2.1 [HEX223-D21]: This deliverable primarily outlines the design principles and structuring requirements for the 6G system blueprint. It does not include specific evaluation results for PoCs or components. Its focus is foundational, setting the stage for subsequent PoC developments.

D2.2 [HEX223-D22]: D2.2 focuses on preliminary results related to System-PoC #A, emphasizing sustainability and orchestration mechanisms. Key aspects include: (i) Sustainability Goals: Evaluation of environmental and social sustainability targets, such as energy-efficient operations for cobots and trust-driven orchestration, to maximise operational lifetime and resilience; (ii) KPI Assessment: Introduction of specific KPIs like reliability, latency, provisioning time, recovery time, power consumption, and trust metrics, providing a foundation for measuring orchestration performance and energy efficiency; (iii) Preliminary Results: Validation of the functionality allocation mechanism using a metaheuristic algorithm, demonstrating power consumption reductions (8.8–28.6%) and improved recovery times for orchestration events compared to baseline approaches.

D2.3 [HEX224-D23]: This deliverable includes more extended evaluation results for System-PoC #A, which focuses on single-domain network configurations. It presents detailed scenarios such as automated warehouse inventory management, energy-efficient functionality allocation, and trust-based resource allocation. Evaluation results include metrics like energy efficiency improvements and the trustworthiness of system operations. More specifically, D2.3 key results included (i) Execution Time and Scalability: The metaheuristic functionality allocation mechanism demonstrated linear scalability in execution time as the number of workloads increased. This contrasts with the exponential increase observed using a Mixed Integer Programming (MIP) algorithm, showcasing the practicality of the metaheuristic approach for larger-scale deployments; (ii) Optimisation Results: The metaheuristic mechanism achieved near-optimal scores for a multi-objective function that balances energy efficiency, E2E latency, and trustworthiness. These results highlight its ability to handle complex workload placement efficiently and maintain performance under growing workload demands; (iii) Advancement Over Baseline Algorithms: When compared to round-robin and random placement methods, the functionality allocation mechanism showed significant improvements in both energy consumption and trustworthiness, ensuring higher system reliability and sustainability.

D2.4 [HEX223-D24]: This deliverable extends the evaluation to System-PoC #B, which introduces multi-domain orchestration and advanced features such as flexible topologies, closed-loop automation, and beyond communication capabilities. It provides both functional and performance validation results, demonstrating alignment with the 6G system blueprint in key areas like resource management and security. In D2.4, the evaluation emphasized the progression of System-PoC #B, with a focus on advanced management and orchestration (M&O) mechanisms, flexible topologies, and beyond-communication features. Key results include: (i) Energy Efficiency and Trustworthiness: The functionality allocation mechanism extended from PoC A demonstrated up to 50.9% energy savings and a 43% increase in trustworthiness under specific weight configurations. These metrics were influenced by optimised workload placement based on energy, latency, and trust parameters; (ii) Dynamic Flexible Topologies: The evaluation of UAV battery usage under varying configurations illustrated the potential for real-time task reallocation to sustain operations during network disruptions. Flexible topology nodes (FTNs) ensured seamless communication, even in scenarios with compromised connectivity; (iii) Intent-Based Service Provisioning: Initial evaluations of intent deployment latency for cobot-based surveillance applications demonstrated promising results, with the majority of intent provisioning times within expected bounds. However, outliers in provisioning times revealed areas for further optimisation, particularly for vertical service instance creation; (iv) AI-Assisted Orchestration: Hybrid deployment strategies enabled by AI-assisted orchestration minimised end-to-end latency for latency-sensitive applications, dynamically balancing workloads between edge and cloud resources. This approach ensured SLO satisfaction even under varying workload conditions; (v) Simulation activities: Comparison of RAN

architecture-related simulations, that compared monolithic RAN and disaggregated RAN (split DU/CU) architectures. The performance metrics collected and compared included latency, processing time, energy consumption, task completion time, and scalability.

4.3.1 System-PoC #A key results

This evaluation highlighted the development and preliminary results of the FA mechanism, a key component of System-PoC #A designed to optimise the placement of computational workloads, tasks, and services across robotic units, edge servers, and cloud resources. The primary objective of the FA mechanism is to enhance energy efficiency while maintaining system performance, making it a critical part of the sustainable AI/ML-based control enabler.

The FA mechanism operates by analyzing input parameters such as the computational and functional requirements of workloads, the capabilities of available compute nodes, and data from edge devices. Based on this analysis, the mechanism produces an optimised placement plan, which is then enforced by the orchestrator. The system dynamically triggers the FA mechanism through the monitoring system when specific conditions arise, such as increased latency, elevated power consumption, or hardware malfunctions in robotic units.

The current implementation of the FA mechanism employs a metaheuristic algorithm, which demonstrates high scalability and suitability for large-scale experimentation. Validation scenarios included seven compute nodes (three robotic units, two edge servers, and two cloud servers) and tested increasing workloads ranging from 4 to 28 tasks. Comparative results illustrated Figure 4-1, showed the FA mechanism outperforming baseline approaches such as Round-Robin Placement (RRP) and Feasible Random Placement (FRP) in terms of energy savings. The FA mechanism achieved power consumption reductions of 8.8% to 28.6%, with higher gains observed as the number of workloads increased.

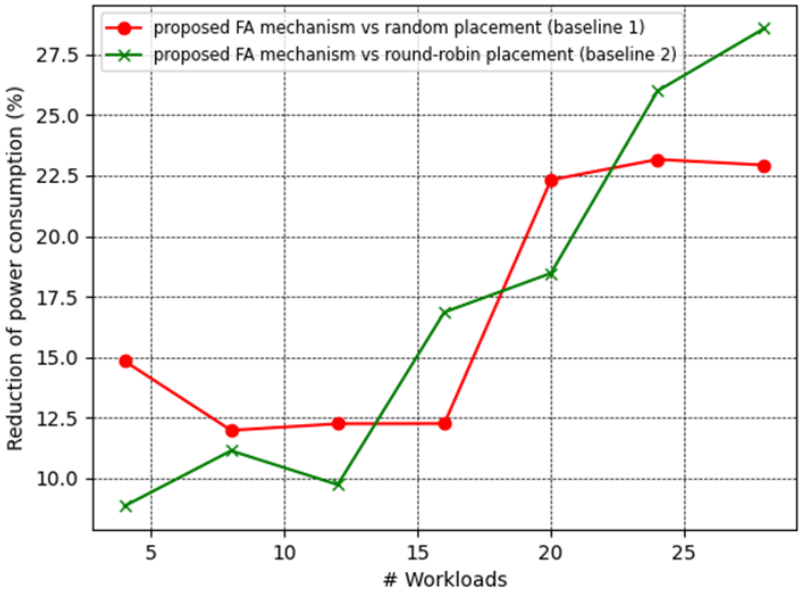


Figure 4-1 Reduction of power consumption with increasing number of compute workloads of the proposed FA mechanism compared with two baseline algorithms.

These results underlined the effectiveness of the FA mechanism in balancing workload placement while significantly improving energy efficiency. This progress sets the stage for its further refinement and integration into more complex multi-domain orchestration scenarios in 6G systems.

Regarding the use of intents in the System PoC, the IBM-IME previously presented in section 3.3.6.1 was used and the latest results related to the "*E2E implementation scenario: Zero-touch Cobot-based Video Surveillance*" [HEX224-D24] are resumed in Figure 4-2. With the set of 15 tests done to evaluate how much latency an requested intent to deploy the cobot-based service in this scenario, it was possible to see that 80% of the times (i.e., 12 samples) required a short period of time to completely deploy the requested service with a value under 255 s (i.e., around 4 min), with the exception of 3 tests that required more time. Based on this,

we can consider that the IBM solution developed has been properly integrated with other solutions implemented by other partners in this scenario with a stable behaviour and it has properly delivered the expected outcomes.

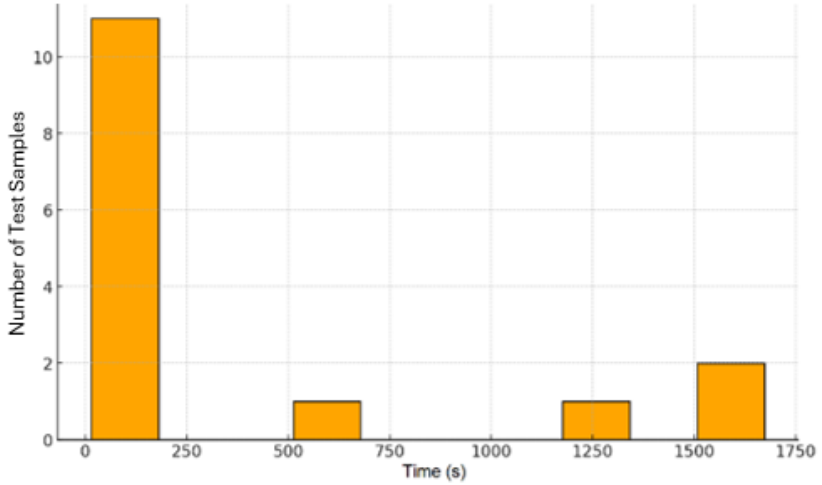


Figure 4-2 Latency to deploy the intent-based service request

Compared to the work done in D2.4 [HEX224-D24], the IBN-IME solution has been integrated in two of the PoC-System implemented scenarios: the "Synergetic Monitoring and Orchestration" and the "Intent-based Service Provisioning with Integrated Closed Loop Deployment and Reactivity for a Cobot Service Migration". To do so, the main integration was with enabler called "Management capabilities exposure framework" [HX224-D63], which offers an integration fabric solution to let multiple enablers placed in different domains to communicate among them using a subscription/publication technology. Moreover, the IBN-IME solution has improved its internal algorithms to Generative Artificial Intelligence (GenAI) to make the whole DSM-IME more efficient.

4.3.2 System- and Component-PoC #B key results

4.3.2.1 Component-PoC #B.1

Component-PoC#B.1 as described in [HEX224-D24], focuses on evaluating orchestration mechanisms for latency sensitive applications in setups where more than one clusters are available. It includes a common *service description* to identify the relevant KPIs, an *AI-assisted orchestration mechanism* to optimally tackle service migration and scaling and the service deployment capabilities of a Karmada Multi-cluster testbed to deploy service chains across heterogeneous infrastructure. The *Monitoring and Telemetry Framework* is used for the real-time observation of the services' deployment during training of the agents and during evaluation of the management lifecycle.

Initial experimentation included two different clusters and how horizontal scaling influences performance. As shown in Figure 4-3, none of the machines can achieve optimal performance for all kinds of workloads. At the Cloud, scaling up successfully reduced computation latency by 10%, but requests suffered from additional communication latency delays (~15 ms). At the Edge deployment: At the Edge, communication latency is lower, but scaling up actually increased latency (up to 3100%), since not enough CPU was available, and concurrency hindered performance severely.

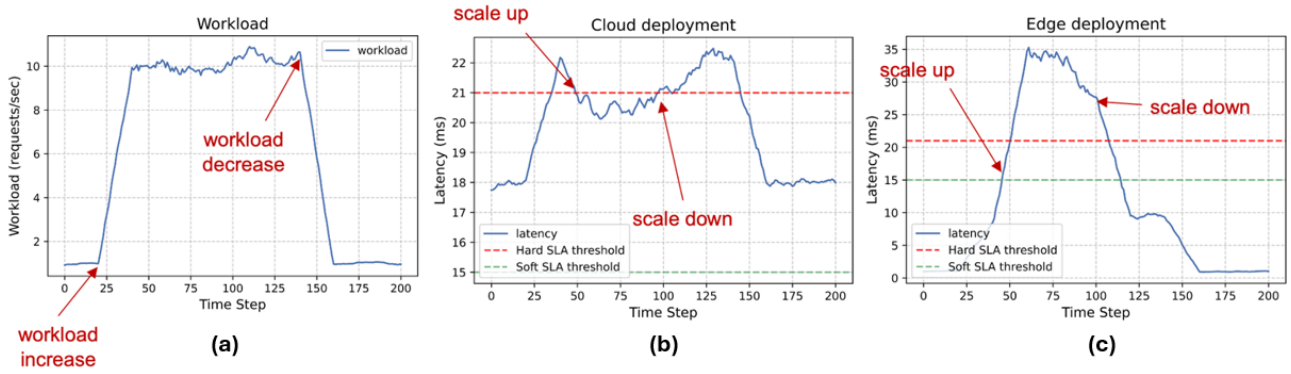


Figure 4-3 (a) Requests per second and latency (ms) measurements on (b) Cloud- and (c) Edge deployment.

During this first round of experiments, a centralised, RL-based, joint autoscaling-migration mechanism was implemented to manage the application. Its goal was focused on minimising resource consumption while satisfying the constraints given by the requested SLA, E2E latency in this case. The latency monitored in comparison with the corresponding SLA value with the mechanism's operation is shown in Figure 4-4 (measured in ms).

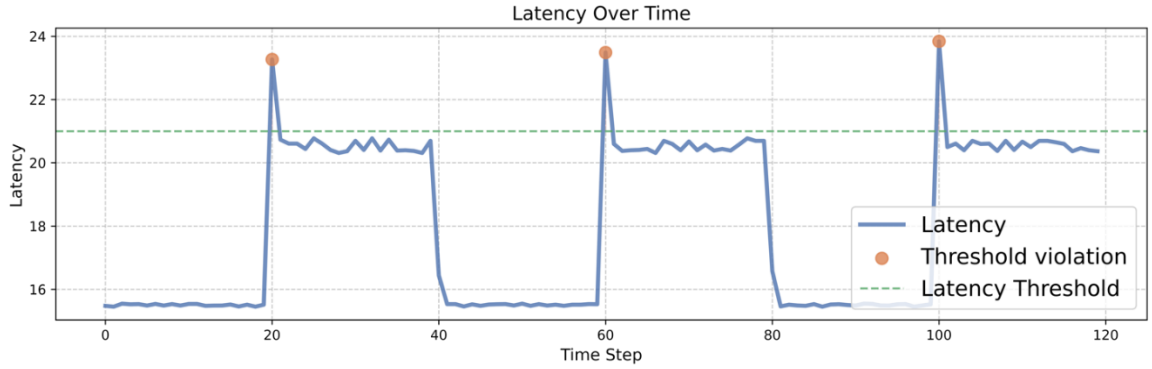


Figure 4-4 Joint autoscaling-migration algorithm runtime performance

Results demonstrated how exploitation of multiple clusters with different geographical location and resources can offer different benefits and finding an optimal trade-off between them can increase efficiency.

Closed-Loop automation (Component-PoC #B.1)

This subsection describes the result in terms of demonstration of a Zero-touch automation based on the implementation of reactive closed-loops in the reference scenario "*E2E implementation scenario: Zero-touch Cobot-based Video Surveillance*" described in Deliverable D2.4 [HEX224-D24]. In this scenario, a specific cobot-based video surveillance service is provisioned by exploiting edge computing and a dedicated cobot-management platform managing two cobots, labeled as *robo-pi1* and *robo-pi2.* The provisioning of the service, performed by exploiting service and resource orchestration is described and demonstrated still in D2.4. Figure 4-5 shows the Kubernetes pods provisioned for CL stages (Analysis, Decision, and Execution, while monitoring is realised through a programmable monitoring platform) and the *streaming-app* for video surveillance provisioned on the cobot *robo-pi1*

Figure 4-5 CL provisioned and streaming-app for video surveillance provisioned on cobot-pi2

Once the surveillance service is deployed i.e., a suitable cobot is selected and provisioned with a video streaming application and the dedicated closed loop is up and running, the status of the cobot's battery is continuously observed by the Monitoring stage. As mentioned, the closed-loop here considered is reactive i.e., it reacts to a certain event, which is in this case the downwards cross of a given threshold of battery level. When the positive event is correctly detected, the reaction time of the CL is in the order of milliseconds. This can be appreciated in the following figures reporting the logs of the Analysis (Figure 4-6), Decision (Figure 4-7, and Execution (Figure 4-8) stages of the closed-loop.

```
[INFO] - 2024-06-04 10:49:02,497 - Retrieved battery metrics
[INFO] - 2024-06-04 10:49:03,431 - Retrieved battery metrics
[INFO] - 2024-06-04 10:49:04,413 - Retrieved battery metrics
[INFO] - 2024-06-04 10:49:05,411 - Retrieved battery metrics
```

Figure 4-6 Analysis stage log

```
[INFO] - 2024-06-04 10:49:04,862 - Average battery level of latest 20 messages: 28.38
[INFO] - 2024-06-04 10:49:04,978 - Sent RESOURCE_LEVEL_MIGRATION event, migration node: ro
[INFO] - 2024-06-04 10:49:05,862 - Average battery level of latest 20 messages: 97.125
[INFO] - 2024-06-04 10:49:06,862 - Average battery level of latest 20 messages: 96.742
```

Figure 4-7 Decision stage log

```
...
[INFO] - 2024-06-04 10:49:05,002 - Sent SERVICE_MIGRATION request for serviceInstanceId: 5d
```

Figure 4-8 Execution stage log

The Analysis stage retrieves battery level samples each second, buffers them and calculates the average over 20 samples. This logic purposefully ignores potential false positives, preventing unnecessary CL executions and consequent service reconfiguration. Indeed, one thing to take into account is that isolated spikes and fluctuations are possible and induce the risk of false positives, especially when the battery level is approaching the given threshold.

The Decision stage checks the average values received by the Analysis with the threshold and when it is crossed, a service migration towards another cobot is decided and timely requested to the Service Orchestrator through the Execution stage.

Focusing on Figure 4-7, it is worth noting that after the decision is taken, the Decision stage starts receiving higher average values of the battery. This is due to the fact the service reconfiguration includes also the CL reconfiguration: the battery values monitored from now on are the one published by the new cobot.

In Figure 4-9 is again reported the dashboard of the Kubernetes cluster, still showing the CL stages and the video surveillance application, which is now provisioned on the second cobot, called *robo-pi2*.

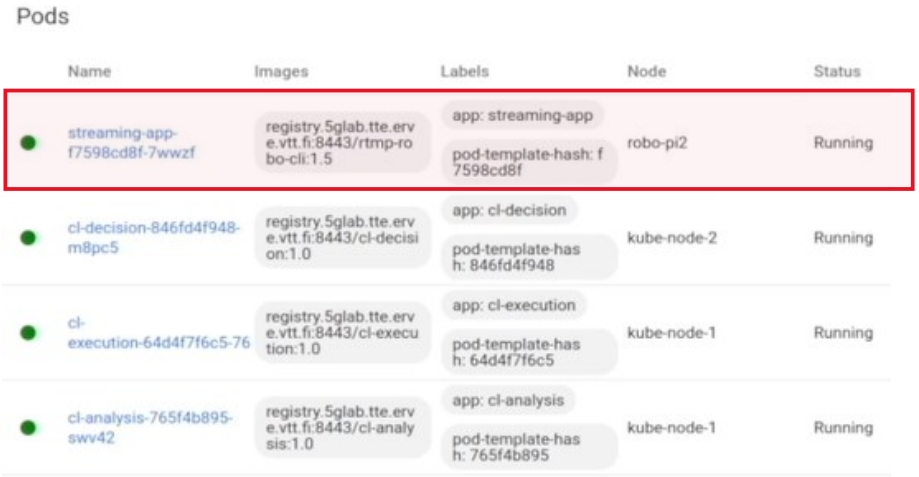


Figure 4-9 Streaming application for video surveillance migrated to cobot robo-pi2

4.3.2.2 Component PoC #B.2

The results of Component-PoC #B.2 demonstrate the efficacy of split learning in distributed environments, with input, generalization, and output nodes deployed as isolated Kubernetes pods to emulate local datasets and workloads. Training a generalization node at the network for two different tasks simultaneously showcased the capability of model generalization to support multi-task learning, improving accuracy for video bitrate and delay estimation simultaneously. The experiment results are shown in Figure 4-10. Figures are referred to as (row#, column #). During the experiment, model layers were dynamically offloaded between the generalization and output nodes. Offloading decreased computational overhead at the output nodes, reducing energy consumption from 22 to 5 J as given in (2,2), and then a temporary impact on the accuracy, at t~20 s as given in (1,2), was observed due to the aggregation of model layers, which then started increasing again over the training rounds and stabilized at high accuracy for both tasks (t >150 s) simultaneously. The offload from generalization node to the output nodes does not have impact on the accuracy (at t~40 s) as the overall neural network weights are kept unchanged and are just relocated.

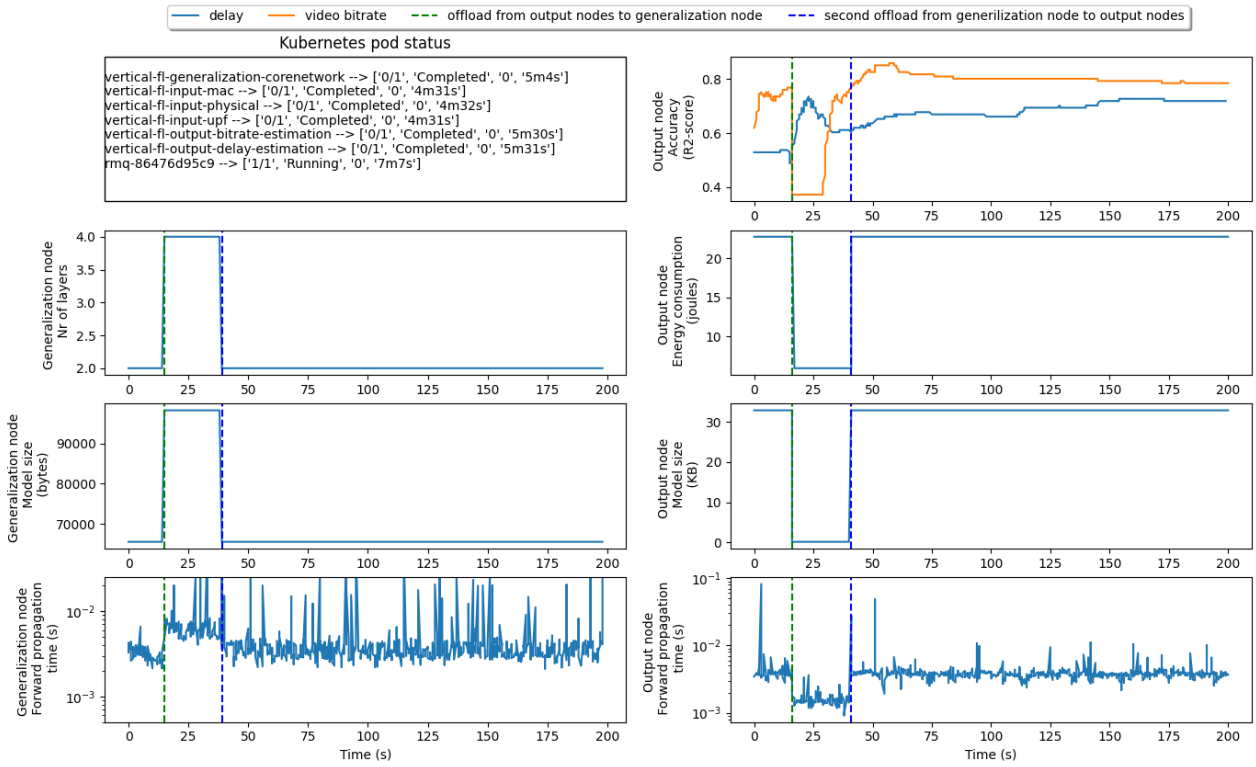


Figure 4-10 (row#, column #): (1,2) model accuracy on two tasks (video bitrate and delay estimation). (2,1) and (2,2) number of model layers in the network and in the application respectively. (3,:) corresponding model size in Bytes. (4,1) and (4,2) illustrate the impact of training time with the changing number model layers in the network and application, respectively. [HEX225-D35].

The role of model layer aggregation during offload in energy efficiency: The reduction in the energy consumption is mainly expected to be at the device side during the offloading, however the model layers that are offloaded to the network increases the energy consumption at the network side. This impact can be rectified with the model layer aggregation process as we demonstrate in this Component-PoC #B.2. Assume that there are 2 layers from one device running two different tasks, delay and video playout bitrate estimation. When 2 layers from each tail model of the task is moved from the device, the energy reduction at the device will be caused by 4 less layers. At the network side, these offloaded layers are aggregated and merged therefore yielding 2 additional layers at the network side instead of 4 additional layers. This means the increase in energy consumption at the network side does not depend on the number of tasks and devices as these layers are aggregated after the offloading process. The network and overall energy consumption before and after offloading is depicted in Figure 4-11. Therefore, it can be concluded that with the right model layer selection mechanisms the energy consumption increase at the network can be kept minimal, and the overall energy reduction is expected to increase during model layer offloading with the increasing number of devices and tasks.

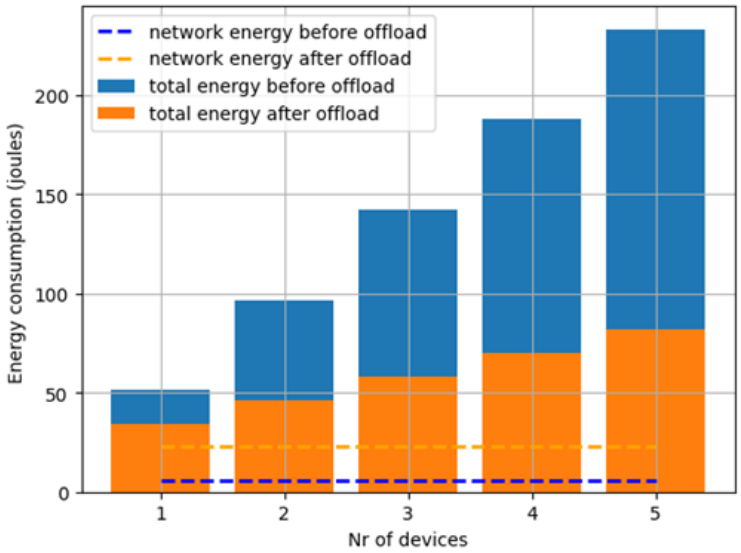


Figure 4-11: The overall energy consumption after model layer offloading from multiple devices to the network [HEX225-D35].

These results confirm the effectiveness of split learning for energy-efficient, adaptive AI applications while maintaining high model accuracy and energy efficiency across tasks.

4.3.2.3 System PoC-B & Component PoC#B.3 – Warehouse inventory management scenario

The main elements showcased as part of System-PoC #B warehouse inventory management scenario comprised advanced management & orchestration features, business intent-driven resource allocation, advanced application features, as well as novel architectural enablers, i.e., trustworthy flexible topologies, and beyond communication enablers, such as the exposure of compute / communication resources and sensing/localisation capabilities. A set of functional as well as performance aspects were validated and discussed. At the application layer of the system, the PoC incorporated various application components such as the cobots' camera video streaming server/client services, the computer vision/object detection components, the navigation and swarm planning components, etc. The network functions layer was extended with the introduction of the trustworthy flexible topologies, enabling the dynamic use of temporary communication and computational resources as needed, which in the System PoC-B was realised with the use of a UAV being deployed in an adhoc manner, when the ground cobot was reported as out-of-coverage. The beyond communications functions included advanced cloud-based services like compute-as-a-service and AI-as-a-service, which offered in real-time the available resources for the FA algorithm to consider and re placement.

The integration fabric, spanning the infrastructure layer, was designed to ensure seamless interaction and coordination between the diverse systems and services within the PoC. This layer also included devices and computational resources as foundational elements. At the pervasive functionalities layer, advanced management and orchestration capabilities—such as intent-based management, closed-loop controls, and multi-platform orchestration—were conceptualized. These were complemented by trust evaluation mechanisms within the security and privacy framework to ensure robust and reliable system performance. Finally, the network functions layer introduced software-defined networking capabilities aimed at managing the transport domain and enhancing system flexibility and efficiency. The above aspects have been specified in System PoC #B partially but are to be elaborated and implemented as part of the System-PoC #C.

Key results of the warehouse inventory management scenario-based System PoC developments included (i) trustworthiness- and energy consumption-related gains, leveraging the FA algorithm (see Figure 4-12); (ii) flexible topology node (FTN) energy consumption (see Figure 4-13); (iii) on-device (UAV/AMR) vs offloading performance in terms of inference times and energy consumption (see Figure 4-14). In particular, the proposed FA mechanism compared to round-robin placement (baseline) showed up to 43% trustworthiness gains and up to 50.9% energy consumption gains as shown in Figure 4-12 and detailed in [Section 3.5.3.1, HEX224-D24]. Figure 4-13 indicates that UAVs can achieve longer operational periods when processing data on-board instead of transmitting it to nearby edge resources, and that efficient UAV-to-edge communication enabled by 6G technology helps balance performance and battery consumption. By strategically allocating

tasks based on UAV capabilities, the function allocation mechanism optimises resource use, reducing power consumption while maintaining high efficiency. Figure 4-14 (a) shows that edge vs on-board compute demonstrated a reduced battery depletion of approximately 20% under heavy ML operations in the experiment detailed in [Section 3.5.3.1, HEX224-D24] and the Figure 4-14 (b) shows the marked reduction in CPU usage from 80% to 30% when AI models are offloaded.

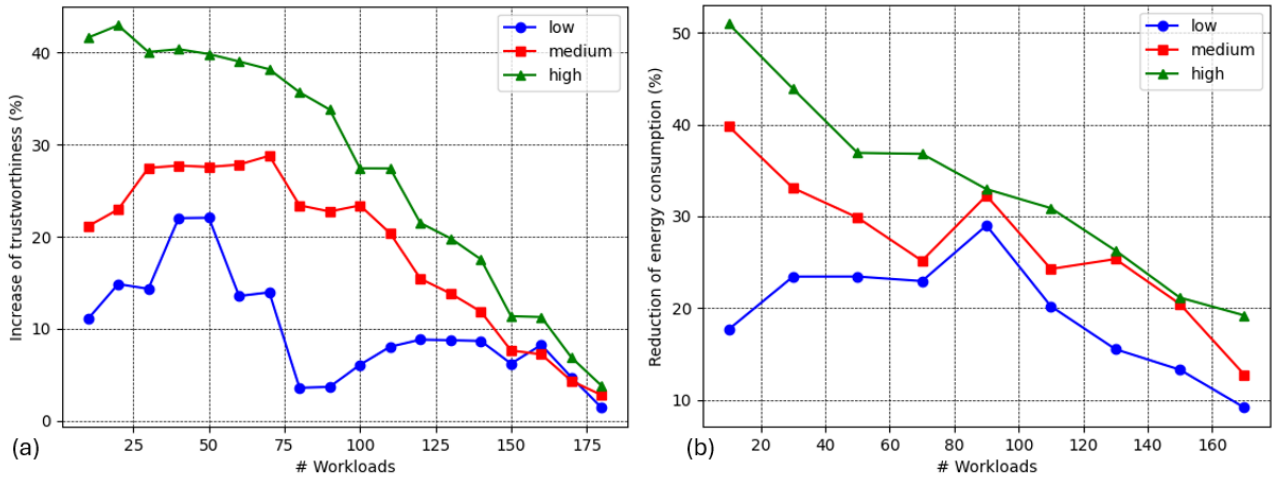


Figure 4-12 Trustworthiness measurements (a) and energy consumption (b) of the FA algorithm.

Conf. #	Setup
1	Transmission Load: 2Mbps, CPU Utilization: 22%, CPU cores: 7, Battery Autonomy: 0.32h
2	Transmission Load: 100Mbps, CPU Utilization: 22%, CPU cores: 7, Battery Autonomy: 028h
3	Transmission Load: 0Mbps, CPU Utilization: 60%, CPU cores: 7, Battery Autonomy: 0.30h

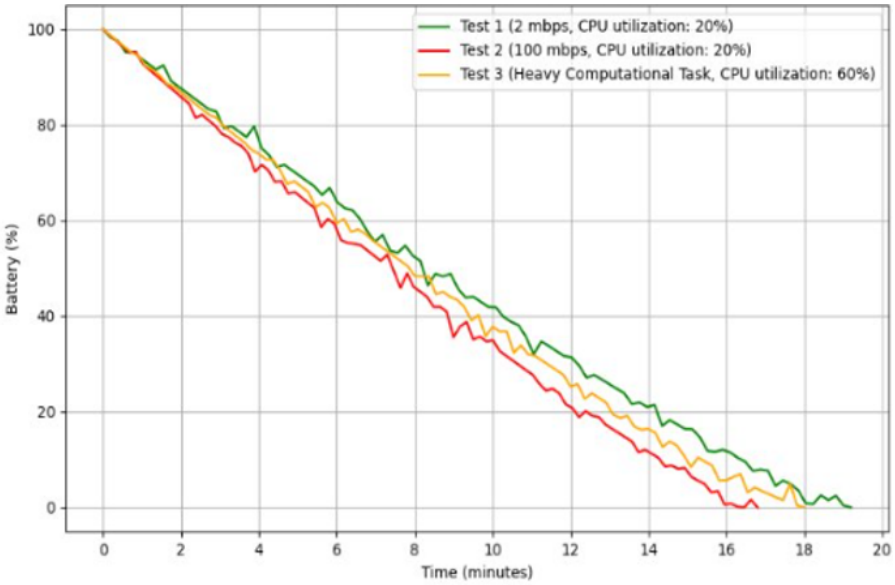


Figure 4-13 UAV (flexible topology node) battery depletion under different configurations.

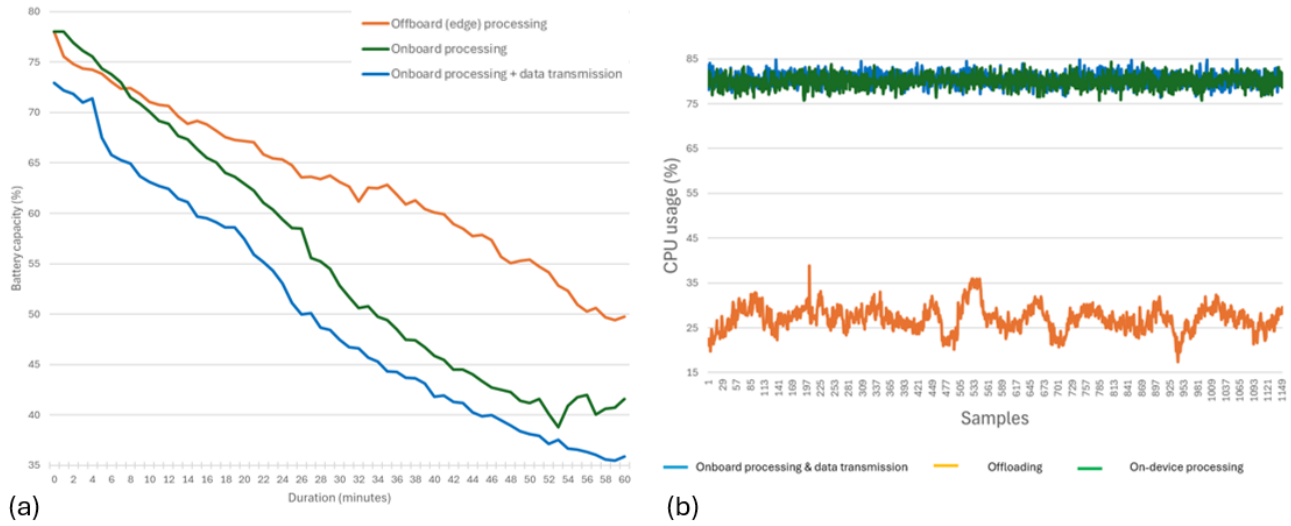


Figure 4-14 (a) Power consumption comparison of on-device (AMR/UAV) vs edge computing setup; (b) Performance benefits of offloading computation via 5G on CPU utilisation.

4.4 Evaluation results

This section provides the actual detailed results of the System- and Components- PoC #C.

4.4.1 6G based sensing algorithms and concepts with real-time performance

According to the objectives of the project, we could achieve close-to-real time detection performance up to 10 frames per second. As shown in Figure 4-15, for the percentile we could achieve higher than 90 percentiles for targets at 4 and 6 bi-static sensing distance. By definition, bi-static distance is considered as the distance from transmitter to the target plus the distance from target to the receiver. We expect to see the same for 10 meters bi-static distance but due to logistic limitations we cannot present results for that.

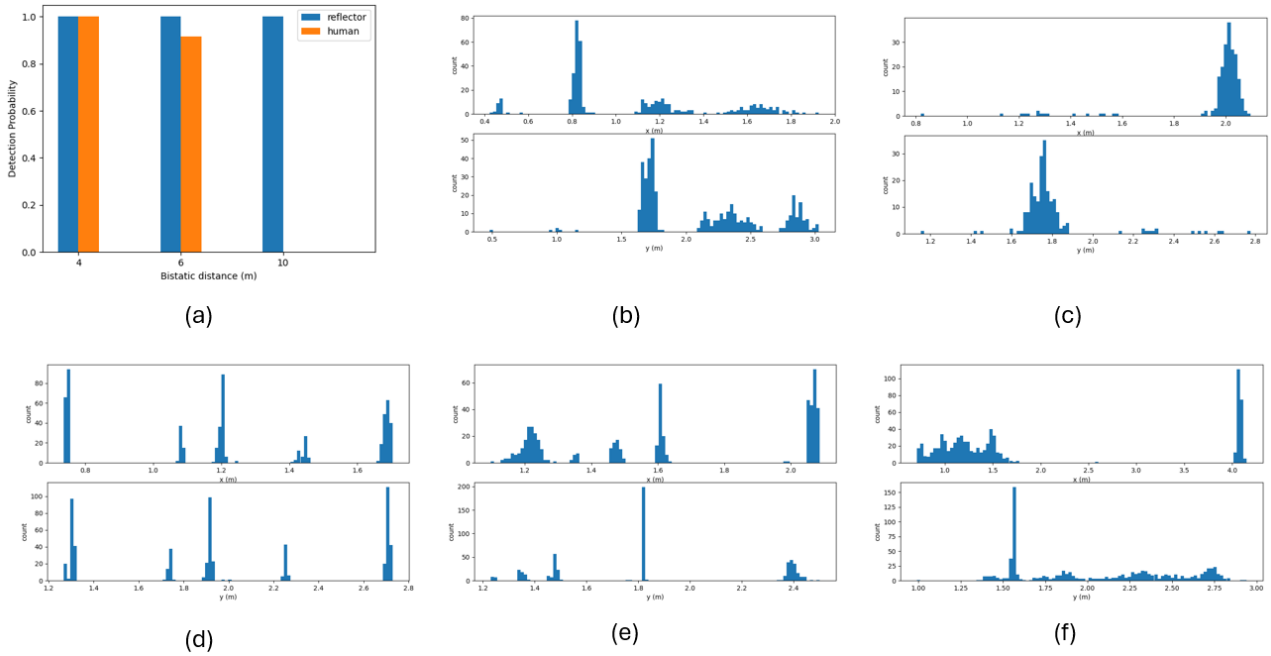


Figure 4-15 a) Detection percentile for human and reflector in different bi-static distance. Detection histogram based on raw signal detection before post processing for b) a human target at 4 m, c) at 6 m, d) a reflector target at 4 m, e) at 6 m, f) at 10 m bi- static distance.

4.4.2 AI-Native Air Interface

The PoC, whose detailed description can be found at [HEX224-D43], was tested in two different setups: through a channel emulator, and over-the-air (OTA) inside the laboratory. The channel emulator experiments were repeated with three different mobility levels, corresponding to pedestrian, car, and train scenarios. Moreover, each measurement was also repeated with non-ML algorithms, which correspond to a 5G air interface. The resulting throughput gains are depicted in Figure 4-16, where the highest throughput gain of 30% is achieved with the highest velocity scenario (train). With lower velocities, the throughput gain is in the order of 15-20%. The gain is mostly caused by the reduced overhead of the AI-based approach, as it has been trained to communicate without DMRS pilots. Since the proposed approach achieves similar bit error rate performance without the DMRS overhead as the conventional approach achieves with DMRS, a higher spectral efficiency is achieved.

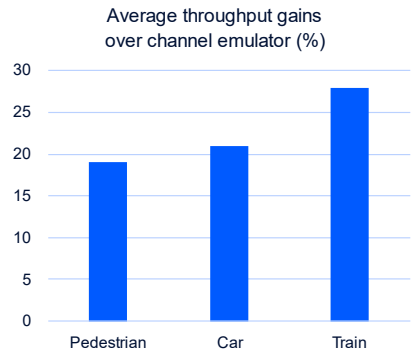


Figure 4-16 Throughput gains measured over channel emulator.

In the OTA measurements, a throughput gain of 19.2% was achieved, which is similar to the low-velocity pedestrian scenario in the channel emulator experiments. This also demonstrates that the proposed approach performs with true physical radio channels.

4.4.3 ML-based channel state feedback compression in a multi-vendor scenario

A testbed for PoC presented in section 3.4.3 has been implemented, and an evaluation of the downlink throughput gain was conducted in a lab testing environment to evaluate performance at varying Doppler frequencies. Results show that ML-based CSF produces a more accurate beamforming as it can achieve, on average, a ~14% DL throughput gain over the baseline CSI feedback method.

Additional results have been collected to evaluate performance of synthetized models. The considered synthetized model is a model trained on a dataset generated from a link-level simulator using clustered delay line channel models with randomized channel parameters covering diverse set of scenarios, both indoor, with mixed line of sight (LoS) and non-LoS and) and outdoor (mostly LoS). The performance is evaluated on different OTA datasets. Simulation-based performance comparison was conducted with a Common model, trained with diverse datasets, not with data corresponding to a specific location, and Type 1-based CSF. The results (see Figure 4-17) show that synthetized models are only slightly worse than OTA models but still provide significant gains over Type 1. The synthetic model performed [1-6] % worse in indoor scenarios but achieve same performance in outdoor scenarios.

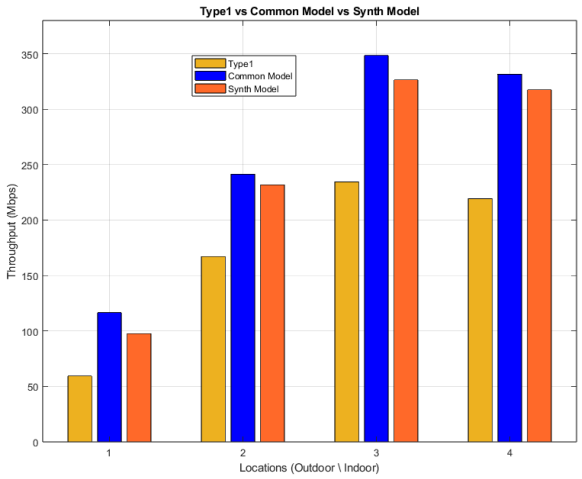


Figure 4-17 Comparison of the downlink throughput between the Type 1, ML-based model with common training and synthetized-based ML model CSF, at 4 different locations.

4.4.4 Crowd-detectable zero-energy devices

The evaluation results of the crowd-detectable zero-energy devices described in Section 3.5.1 are presented in this subsection. Figure 4-18 presents the measured bit error rate (BER) performance in a laboratory environment. It also compares the performance with the theoretical BER curve of a non-coherent FSK receiver.

Figure 4-19 illustrates two of the field trials described in D5.5 [Section 7, HEX225-D55]. The field study was done using commercial LTE network and SDR implementation of the UE. The results indicate that BD could be read by a mobile phone even in case the base station is outside, and the BD and UE are inside. The measured reading distance was 0.5 meters.

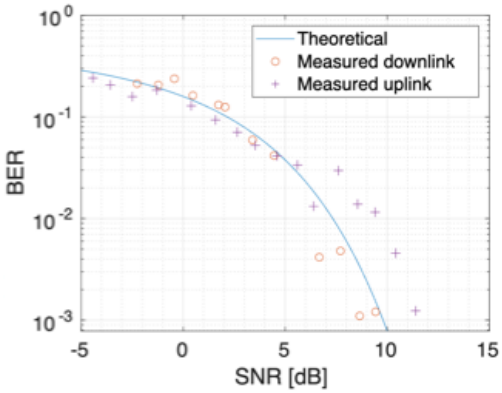
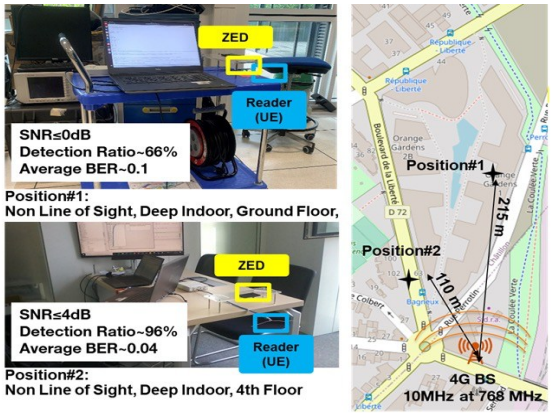


Figure 4-18 Measurement result of the crowd-detectable zero-energy devices integrated to LTE network.

Two field trials were conducted to evaluate the system's performance in the downlink direction: one at Orange Gardens in Paris and another at the Aalto University campus in Espoo (Figure 4-19). In both cases, a commercial LTE base station was used to illuminate the backscatter device, and a software-defined radio (SDR) implementation of the UE channel estimator served as the receiver for the backscatter-modulated messages.

Field trial in Paris



	Position 1	Position 2
Average 4G SNR	≤0 dB	≤4 dB
Observation duration	4171s	4171s
Transmitted frames	852	867
Detected frames	561	835
Detection Ratio	65.8%	96.27%
Average Data BER	0.1008	0.0439

Field trial in Espoo

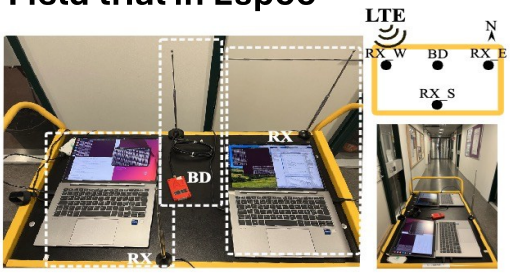


TABLE I
MEDIAN AND 90TH PERCENTILE BER VALUES BY ENVIRONMENT AND FLOOR (TWO-DIGIT ACCURACY)

Environment	Floor	Median	90th Percentile
Bridge	1st	1.1×10^{-2}	1.2×10^{-1}
Bridge	2nd	2.3×10^{-2}	9.7×10^{-2}
Bridge	3rd	8.4×10^{-3}	7.9×10^{-3}
Corridor	1st	4.7×10^{-2}	1.3×10^{-1}
Corridor	2nd	2.1×10^{-2}	7.4×10^{-2}
Corridor	3rd	2.1×10^{-2}	7.8×10^{-2}

Figure 4-19 Field trials in Orange Gardens in Paris (left figure) and Aalto University campus in Espoo (right figure).

4.4.5 Energy harvesting IoT proximity devices

Two scenarios were explored for evaluating and testing the developed EH IoT proximity device described in Section 3.5.2: the indefinite operation of the device under a lamp and the slow drain of the device's battery.

In the first scenario, the main assumptions are that the sensor has a 30s time cycle, where sensor is active for 2 s at 30 mA (see Figure 4-20) and then sleeps for 28 s at 9 mA (see Figure 4-21). Every 10-time cycles, a Wi-

Fi transmission occurs for 45 s at 50 mA. The lamp provides a continuous current of 27-30 mA (see Figure 4-22) is positioned 10 cm away from the panel. As a result, in one 30s cycle, the device draws (30 mA × 2 s) + (9mA × 28 s) = 312 mA·s. Over 10 time cycles (300 s total), the device draws 312 mA·s × 10 = 3120 mA·s. The Wi-Fi transmission that occurs every 10 cycles for 45 s at 50 mA (see Figure 4-23) draws 50 mA × 45 s = 2250 mA·s. Thus, the total current drawn by the device for 10 cycles and a transmission is 3120 mA·s + 2250 mA·s = 5370 mA·s. The total time duration is 300 s + 45 s = 345 s. Therefore, the average current consumption in this scenario is 5370 mA·s / 345 s ≈ 15.6 mA. Compared to the 27-30 mA that the lamp provides, the device will slowly recharge and will not drain the battery.

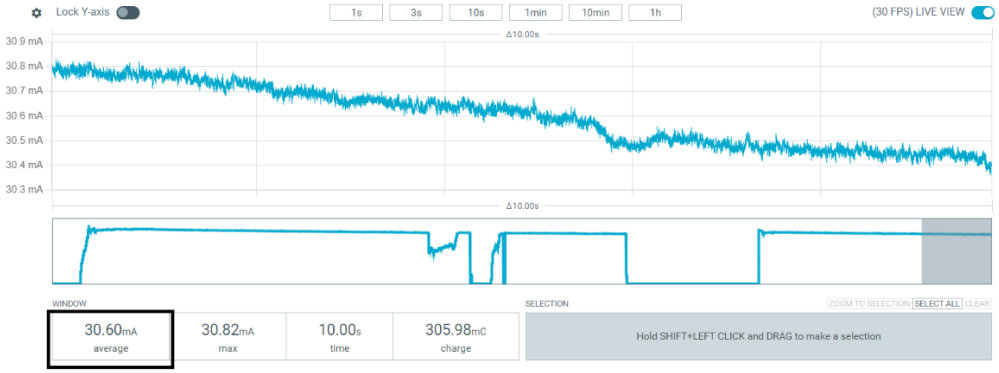


Figure 4-20 Sensor active current consumption.

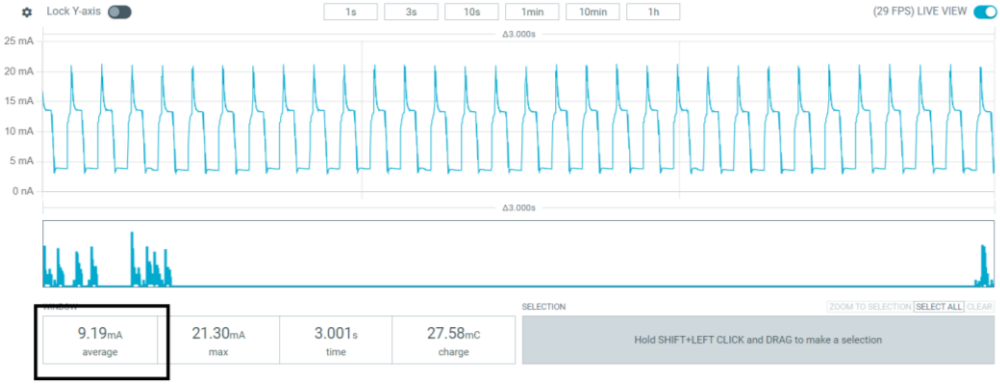


Figure 4-21 Sleep current consumption.

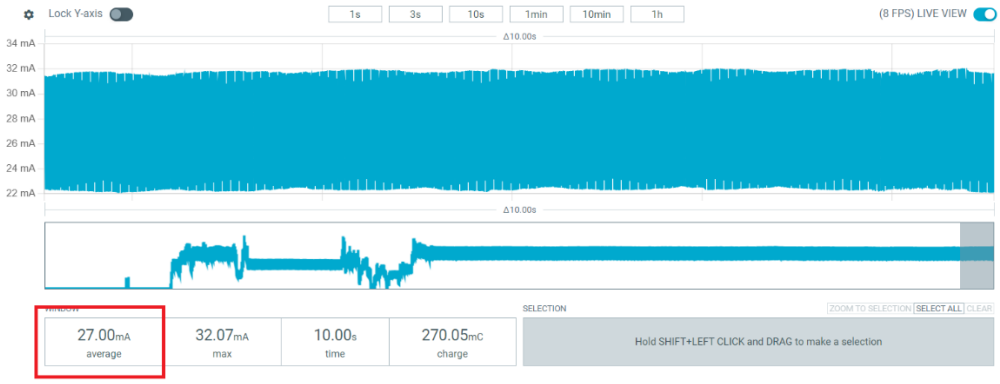


Figure 4-22 Solar current production.

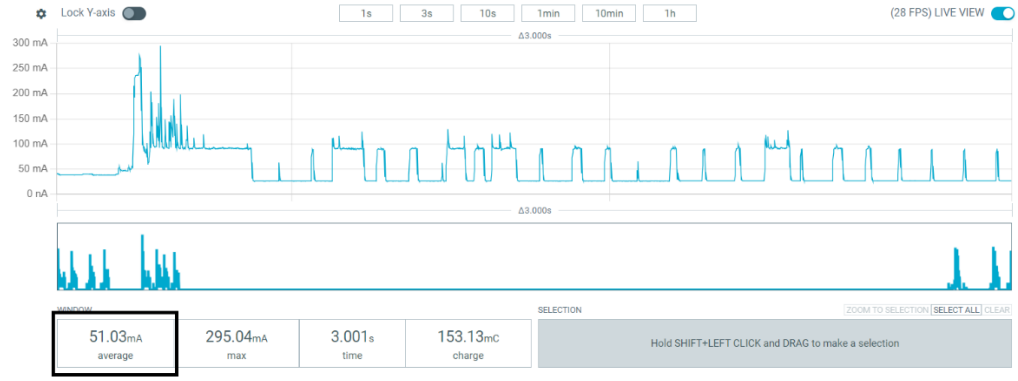


Figure 4-23 Current consumption of Wi-Fi data transmission.

In the second scenario, the sensor time cycle is also assumed to be 30 s, with the sensor active for 2 s at 30 mA and sleeping for 28 s at 9 mA. However, a Wi-Fi transmission occurs every 2 sensor cycles, lasting for 30 s at 50 mA. The lamp produces a continuous current of 27-30 mA is positioned 10 cm away from the panel. The battery capacity is 2400 mAh, and the device's current draw is 35.4 mA while the lamp provides 27 mA. The network current drain, which is the difference between the device consumption and the lamp supply, is 35.4 mA – 27 mA = 8.4 mA. Thus, the battery life, given the network current is 2400 mAh / 8.4 mA = 285.7 hours. Therefore, if the DC-DC converter, which consumes 9 mA, is removed, a better autonomy solution could be achieved.

4.4.6 End-to-end Extended Reality

The results of the PoC presented in Section 3.5.3 are summarized in the Figure 4-24, comparing the video stall duration due to video frame loss with and without L4S enabled in a scenario with varying NW load. The results demonstrate that by adjusting the application data rate, XR Distributed Compute can reduce the number of impacted frames thus improving the user QoE. While a few frames are lost when the load starts, the impact on the user experience is minimal compared to the no L4S scenario.

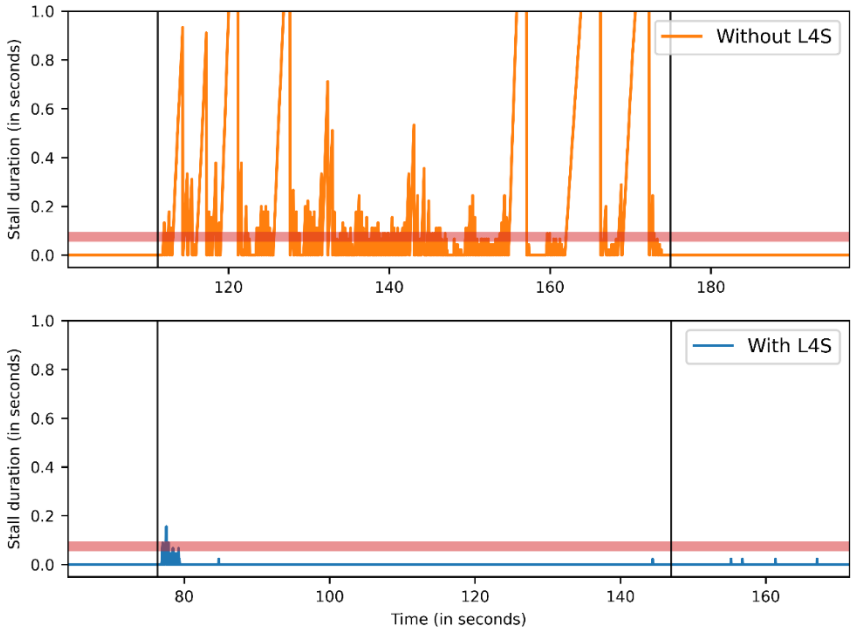
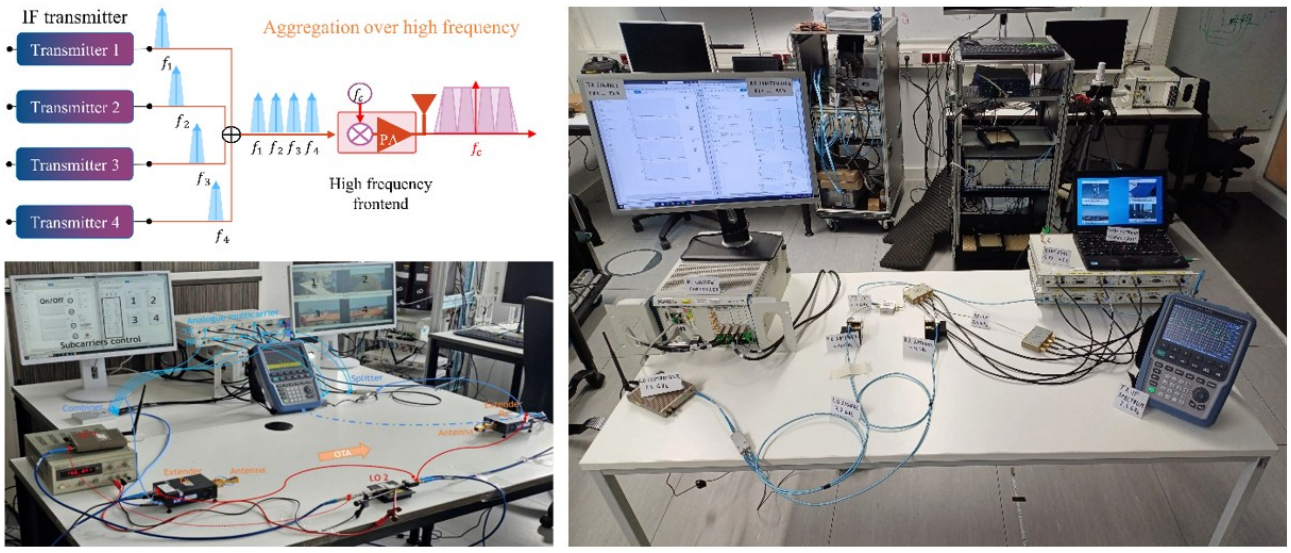


Figure 4-24 Comparison of the number of consecutive duplicated frames with and without L4S enabled in a scenario with varying NW load. The red line indicates a threshold (subjective to each user) for which the user experience is impacted. Here the threshold is set to a stall duration of [50, 100] ms. The black line vertical lines indicate the start and end of the user loading.

4.4.7 Flexible modulation and transceiver design

The flexible transceiver system is designed to enable hardware reconfigurability, by allowing adaption of RF frontends for different frequency bands and use cases. This flexibility allows researchers to investigate diverse communication scenarios, validate signal processing algorithms and test sensing applications.



(a) (b)

Figure 4-25 Data transmission (video stream) with high frequency RF frontends. (a) 140 GHz RF frontend and (b) 10 GHz RF frontend.

Reconfigurable RF Frontends

One of the key features of the flexible transceiver system is its ability to support multiple RF frontend configurations (reconfigurability) [HEX2-BCN+24]. The system has been tested with frontends operating at different frequency bands, to demonstrate its versatility:

- **140 GHz (Sub-THz):** This RF frontend has been developed in WP4 alongside channel modelling studies for the sub-THz frequency range. Integration of the 140 GHz frontend from Oulu is shown in Figure 4-25(a), demonstrating feasibility for over-the-air (OTA) data transmission. Further details are documented in [HEX225-D45].
- **10 GHz (FR3):** A PoC for data transmission using the flexible transceiver system with a generalized frequency division multiplexing (GFDM) modem has been successfully conducted, showcasing the transceiver's capability in this frequency range. The setup is represented in Figure 4-25(b).
- **3.75 GHz (Beam Management and Angle Estimation):** The system, illustrated in Figure 4-26, has been used to estimate the Angle of Departure (AoD) and facilitate beam management, specifically as PoC for fast initial beam acquisition. The approach builds on [HEX225-D45] section 11.2 and offers in addition real-time phase calibration techniques, as outlined in [HEX2-CNF25].
- **2 GHz (Carrier Aggregation):** The transceiver has demonstrated the ability to perform carrier aggregation both OTA and via cable-based setups in the sub-6 GHz band (Figure 4-27). This flexibility supports investigations into advanced aggregation strategies for enhanced spectral efficiency.

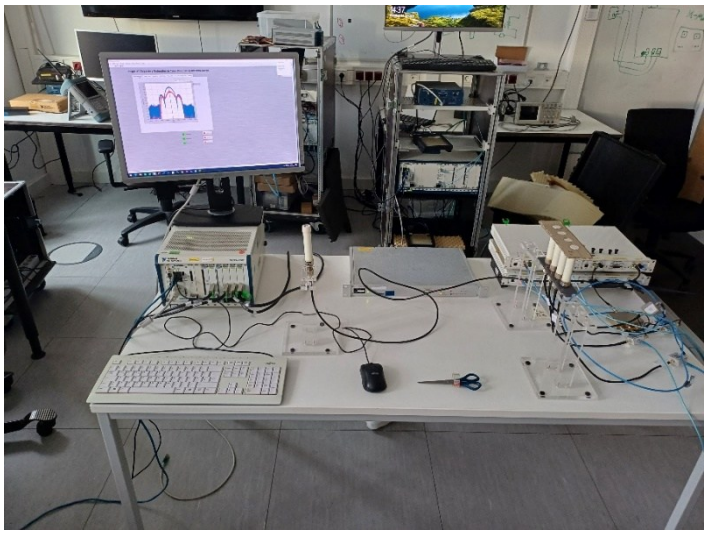


Figure 4-26 Angle estimation at 3.75 GHz for fast initial beam acquisition.

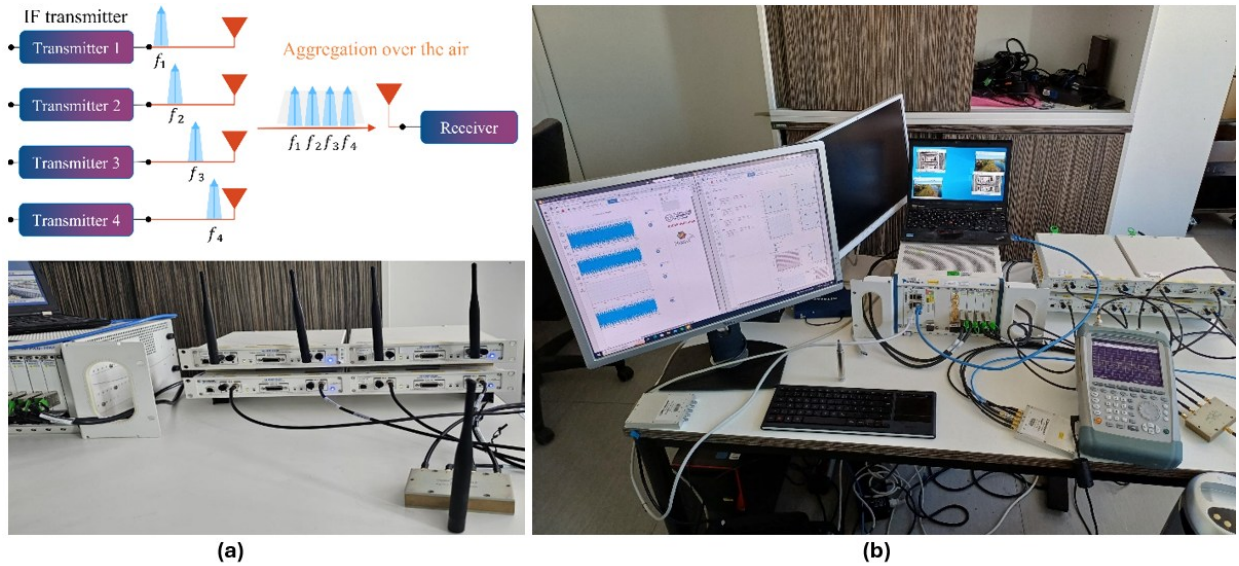


Figure 4-27 Carrier Aggregation at sub-6 GHz frequency band (2 GHz). (a) Carrier Aggregation OTA and (b) Carrier Aggregation over Cable.

Key Findings and Performance Insights

The evaluation of the flexible transceiver system confirms that the reconfigurable hardware enables a wide range of experimental setups. The ability to dynamically switch between different RF frontends and adjust system parameters in real time provides researchers with a powerful tool for testing new concepts in mobile communications and radar applications. Furthermore, the system's modular nature ensures that additional RF frontends can be integrated with minimal effort, expanding its applicability to new research domains.

4.4.8 Radio propagation measurements to collect data for radio channel modelling

Propagation campaigns were conducted at 15 GHz, 142 GHz, 234 GHz and 318 GHz using both USRP-based and VNA-based setups across indoor and outdoor scenarios. At 15 GHz, linear-route sweeps were performed in residential streets, a university campus and a parking lot using a USRP testbed as shown in Table 4-1 for sub-THz operation, vector-network-analyzer measurements targeted an entrance hall, suburban and city-centre sites at 142 GHz [DHK23], and factory halls plus warehouse layouts at 234 GHz and 318 GHz. At each Tx–Rx location pair, discrete azimuth and elevation scans captured the full multipath angular structure.

Table 4-1 Simulation settings and parameters for the channel measurement campaign

Frequency	Environment	Measurement Description
142 GHz [DHK23]	Entrance hall, suburban, residential, and city centre	VNA, single-directional, discrete Tx/Rx locations
234 and 318 GHz	Factory hall and warehouse	VNA, bi-directional, discrete Tx/Rx locations
15 GHz [ATK+25]	Residential, campus, and parking lot	USRP-based, linear route measurements

Preliminary findings from the 234 GHz and 15 GHz measurements are presented in Figure 4-28 and Figure 4-29. The experimental configuration and channel analysis are described in Section 3.3 [HEX225-D45] for the sub-THz dual-band indoor campaign, and in Section 3.7 [HEX225-D25] for the 15 GHz outdoor campaign.

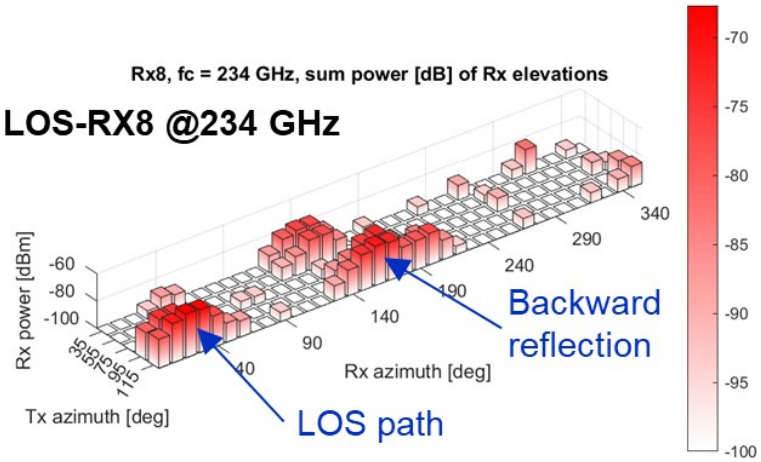


Figure 4-28 Bi-directional power angular spectrum of Tx-Rx8 link at 234 GHz in factory hall.

Figure 4-29 Coverage map showing path gain (dB) along the measurement routes in a UMi (left) scenario (residential area) and a UMa (right) scenario (university campus/ parking lot) at 15 GHz.

4.4.9 Integration fabric general analysis on performance and impact

The Integration Fabric, introduced in the context of System-PoC #C, represents a significant upgrade in the integration layer compared to previous Proofs of Concept. It transitions from traditional client-server or RPC-based paradigms to an event-driven architecture at the service level. This transformation enables improved ease of integration by democratising data communication access, shifting from punctual 1-to-1 communication to shared topic-based communication that allows for broader connectivity within the system. The democratisation of communication enhances interoperability while improving security through encrypted channels and controlled access mechanisms such as Access Control Lists (ACLs). Furthermore, this approach fosters a loosely coupled architecture where components autonomously trigger and respond to events without centralised control. However, it should be noted that the introduction of an event-driven model does bring challenges, particularly in terms of increased process overhead and latency.

As illustrated in the following paragraph, the results closely align with the overarching goals and challenges introduced by the Integration Fabric in System-PoC #C. The evaluation of the API endpoints, as illustrated in Figure 4-30 through Figure 4-33, provides a comprehensive and detailed understanding of the system's ability to handle dynamic operations under varying workloads. Figure 4-30 presents a summary of the general statistics for onboarding and offboarding operations, highlighting how the system manages the introduction of 50 new IDs alongside the creation of 20 additional topics from a fixed ID. The high success rates in this figure highlight the reliability of the system, demonstrating its capacity to handle frequent registration and deregistration events without significant errors or failures. This performance is particularly important in scenarios where dynamic user activity is expected, as it ensures that the system can respond promptly and effectively to changes in demand.

Route	Avg Latency	Median Latency	P95 Latency	P99 Latency	Throughput	Error Rate	Availability
register	1.864 s	1.838 s	1.883 s	2.582 s	0.54 req/s	0.00%	100.00%
registerNewTopic	1.663 s	1.637 s	1.846 s	1.889 s	0.60 req/s	0.00%	100.00%
deleteTopic	3.776 s	3.653 s	4.745 s	4.823 s	0.26 req/s	0.00%	100.00%
unregister	3.476 s	3.471 s	3.753 s	3.785 s	0.29 req/s	0.00%	100.00%

Figure 4-30 Overview of onboarding and offboarding events statistics.

Moving to Figure 4-31, we see an analysis of the temporal evolution of endpoint latency, which tracks how response times fluctuate over the duration of the test. Initially, there are noticeable variations in latency as the system adjusts to the increased load caused by onboarding new IDs and generating additional topics. However, these fluctuations gradually subside, with latency stabilizing at consistent levels. This stabilization highlights the system's ability to adapt to workload spikes and maintain steady performance over time, which is critical for ensuring a seamless user experience even during periods of high activity.

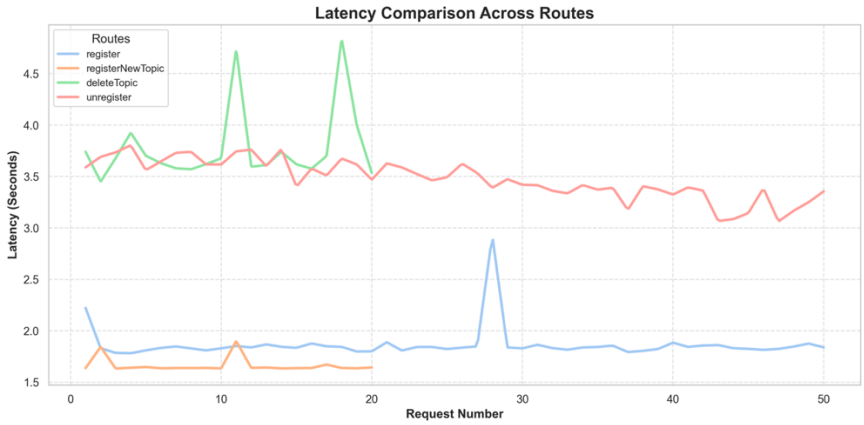


Figure 4-31 Latency trends during Integration fabric API endpoints testing.

Figure 4-32 provides further insight into latency behaviour by illustrating its distribution across all API calls during the test. The data shows that most response times fall within an acceptable range, confirming that the majority of requests are processed efficiently and without significant delays. However, there are occasional outliers where latency exceeds typical thresholds, potentially corresponding to moments of peak workload or specific operations such as topic creation. These outliers suggest areas where further optimisation could enhance performance consistency and reduce variability in response times.

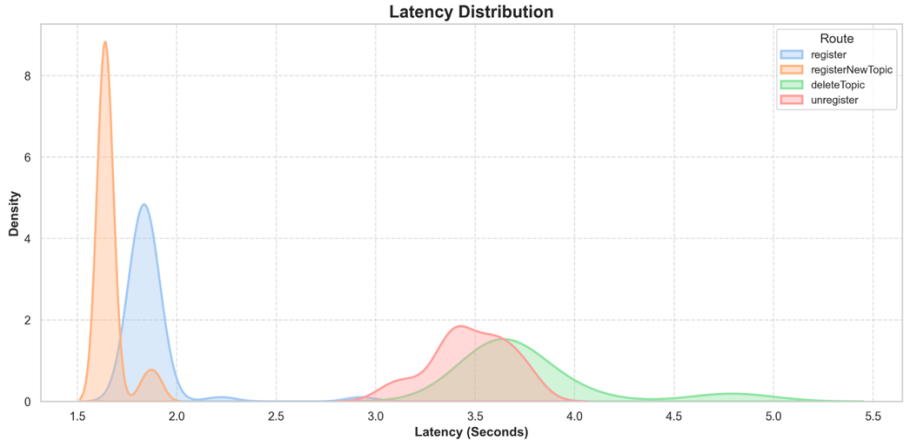


Figure 4-32 Distribution of latency values across API endpoints.

Finally, Figure 4-33 integrates average latency and throughput metrics to offer a comprehensive view of how well the system balances responsiveness with processing capacity. Despite the additional load introduced by new IDs and topics, the system maintains strong throughput levels while keeping average latency within manageable limits. This balance underscores its ability to deliver high performance even under challenging conditions, ensuring that both speed and efficiency are preserved.

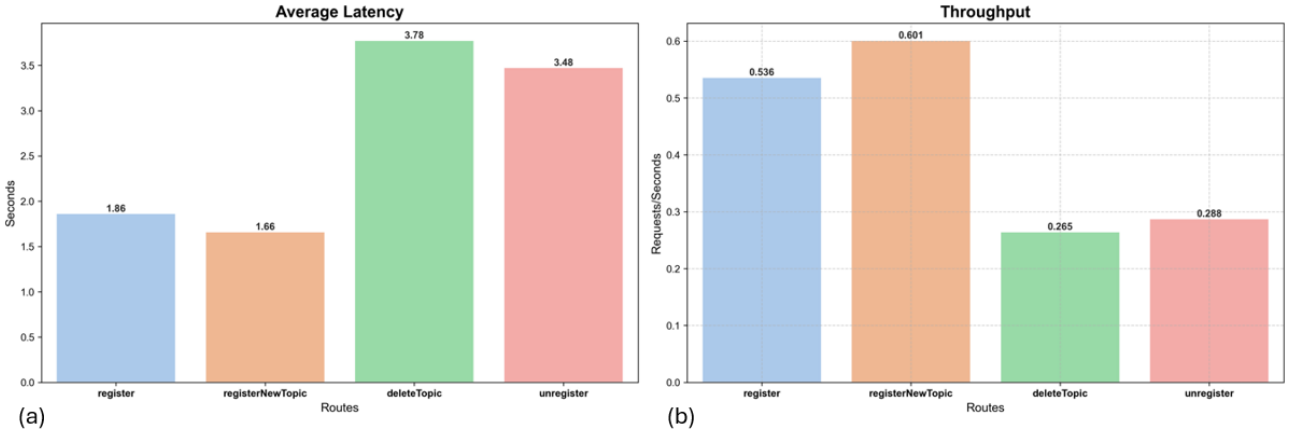


Figure 4-33 Comparison of average latency (a) and throughput (b) for API endpoints.

The analysis of producers and consumers, represented in Figure 4-34 through Figure 4-36, offers equally compelling evidence of the system's robustness in handling large-scale message processing tasks. Figure 4-34 focuses on producer performance when dispatching messages in batches of varying sizes, ranging from 100 to 5,000 messages. The results demonstrate that producers operate efficiently across this range, with dispatch times remaining relatively stable regardless of batch size variability. This consistency suggests that producers are well-optimised for handling unpredictable workloads without significant degradation in performance. Additionally, it highlights their scalability, as they can manage both small and large batches with equal effectiveness.

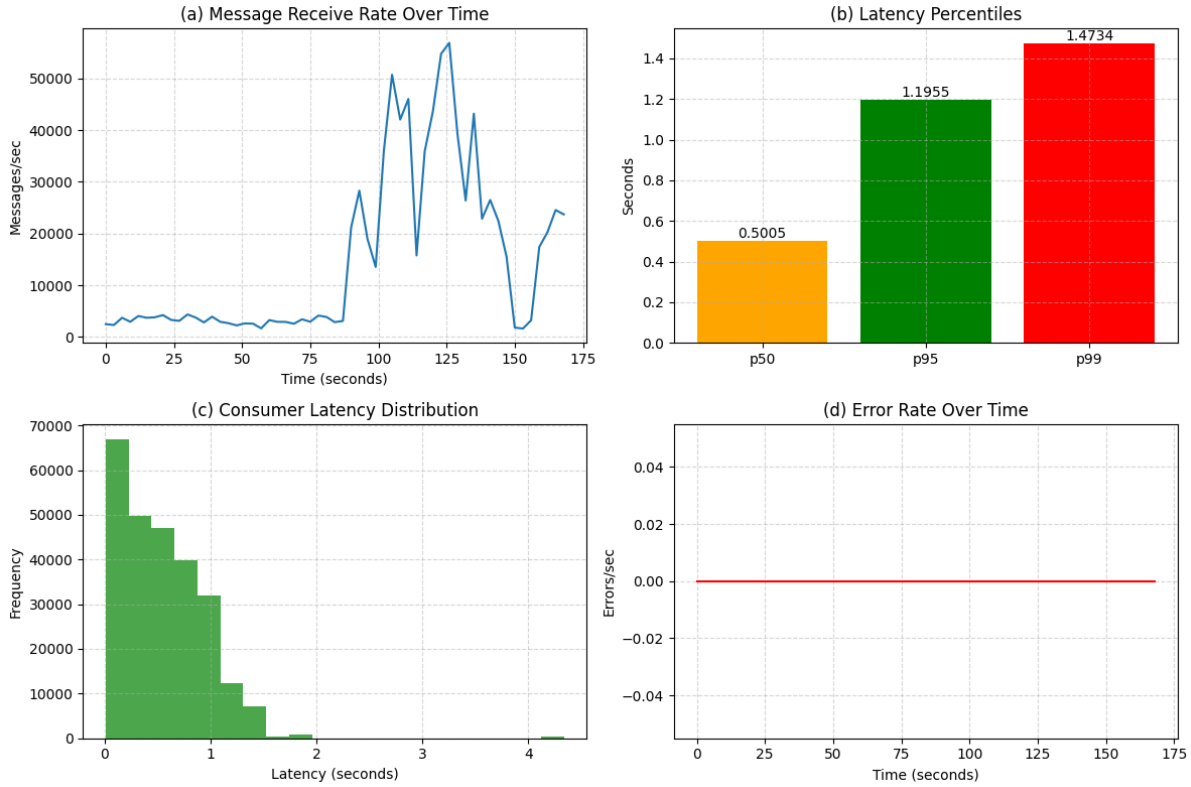


Figure 4-34 Performance metrics for message producers across batch sizes: (a) Message sent and delivery rate over time; (b) Latency percentiles; (c) Batch size distribution; (d) Error rate over time

Figure 4-35 shifts attention to consumer performance, showcasing how messages are retrieved and processed under similar conditions. The data reveals that consumers are equally capable of managing variability in batch sizes while maintaining strong throughput and low latency levels. This finding reinforces confidence in the

system's end-to-end processing capabilities, as it shows that consumers can keep pace with producers even when message volumes fluctuate significantly.

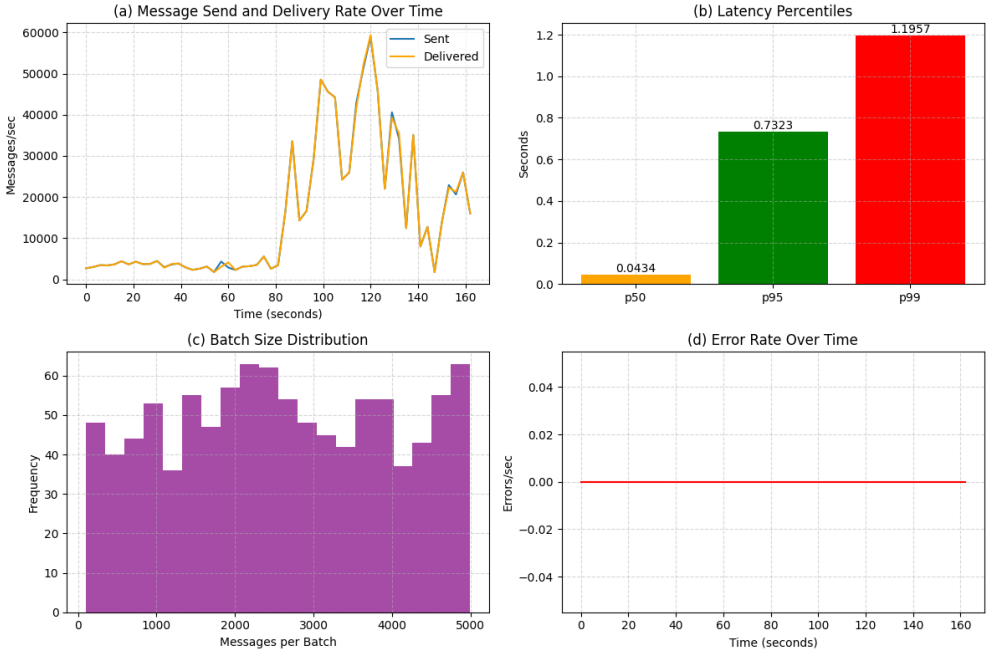


Figure 4-35 Analysis of consumer efficiency in message processing: (a) Message receive rate over time; (b) Latency percentiles; (c) Consumer latency distribution; (d) Error rate over time.

Finally, Figure 4-36 provides an integrated view of producer and consumer statistics, offering a comprehensive perspective on how these two components interact within the messaging pipeline. The figure demonstrates that communication between producers and consumers is smooth and efficient even under challenging conditions involving random batch sizes and high message volumes. By balancing producer dispatch rates with consumer processing speeds, the system ensures that bottlenecks are minimised and throughput remains consistent throughout the pipeline. This synchronization between producers and consumers is critical for maintaining overall system efficiency and avoiding delays or backlogs in message processing. Together, these results emphasize that the system is not only reliable but also scalable, making it well-suited for real-world operational demands where dynamic workloads and high message volumes are common challenges.

Entry	Message Metrics	Latency Avg	Latency P50	Latency P95	Errors	Error Rate	Message Rates	Duration
Producer	Sent: 2783838 messages Delivered: 2394616 messages	0.2348 s	0.0434 s	0.7323 s	0	0.00 err/sec	Send Rate: 15651.67 msg/sec Delivery Rate: 13463.87 msg/sec	152.22 s
Consumer	Received: 2936826 messages	0.5438 s	0.5005 s	1.1895 s	0	0.00 err/sec	Receive Rate: 14839.73 msg/sec	197.99 s

Figure 4-36 Integrated producer and consumer performance statistics

These results highlight the Integration Fabric's ability to balance scalability and security with processing efficiency. For endpoint onboarding/offboarding, it simplifies integration processes while maintaining scalability and security but introduces moderate computational overhead that must be managed as topic complexity grows. For producer-consumer interactions, it supports high throughput and reliable delivery but requires careful handling of large message sizes to minimise latency impacts. These findings validate the Integration Fabric's role as a robust enabler for scalable and secure communication in distributed systems. This level of performance ensures that the system can meet user expectations for responsiveness and reliability even under demanding conditions.

4.4.10 Cobot-powered Warehouse Inventory Management scenario leveraging BCS exposure and Trustworthy Flexible Topologies

The warehouse inventory management scenario exemplifies the practical application of advanced network architectures in industrial settings, combining collaborative robots (cobots), UAVs, and dynamic network components to deliver real-time operational efficiency. This scenario leverages both BCS exposure and

trustworthy flexible topologies (Flextop) to create a resilient and adaptive system for inventory inspection and management. In this deployment, as shown in Figure 4-37 ground-based AMRs are tasked with conducting detailed inspections of ground-level inventory using optical payloads. In parallel, worker UAVs are deployed to inspect high shelves, thereby covering all areas within the warehouse. Additionally, a specialized Flextop UAV acts as a mobile access point, extending network connectivity dynamically in places where standard communication links are compromised. This layered approach ensures continuous coverage and connectivity, even if disruptions occur. In cases where connectivity is lost, the system is designed to hold the current task state until the network is re-established; if no alternative connectivity is available, the robots safely return to a designated home base after a predetermined timeout.

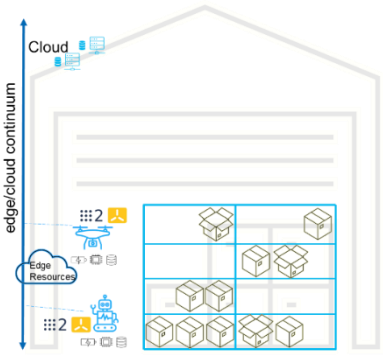


Figure 4-37 Cobot-powered warehouse inventory management setup.

The orchestration of the warehouse inventory management scenario is achieved through a multi-layered process. The user interface (UI), depicted in Figure 4-38 captures the operational intent, allowing operators to select target zones and performance modes, such as maximum performance or minimal resource utilization. The UI includes a map widget that displays the last-known two-dimensional locations of all operational units, along with critical information such as battery levels, network KPIs, and video streams. Following the input from the UI, a *Taskplanner* module generates an optimised task list based on robot availability and current state information. This task list is subsequently managed by a *FleetManager*, which orchestrates the deployment and real-time coordination of AMRs and UAVs, ensuring that the inspection process proceeds safely and efficiently. At the core of this scenario is the exposure of network functions through standardized CAMARA-compliant APIs. Exposed Network Functions (ENFs), such as the Sensing/Localisation Function, Device Status/Location, and Quality-on-Demand, provide the necessary data streams for real-time decision-making. These functions facilitate the integration of high-precision sensor data, enabling enhanced navigation, obstacle detection, and inventory auditing. By offloading computationally intensive tasks, such as object detection and video processing, to edge or cloud servers, the framework ensures that local devices remain energy efficient while still benefiting from centralized processing power.

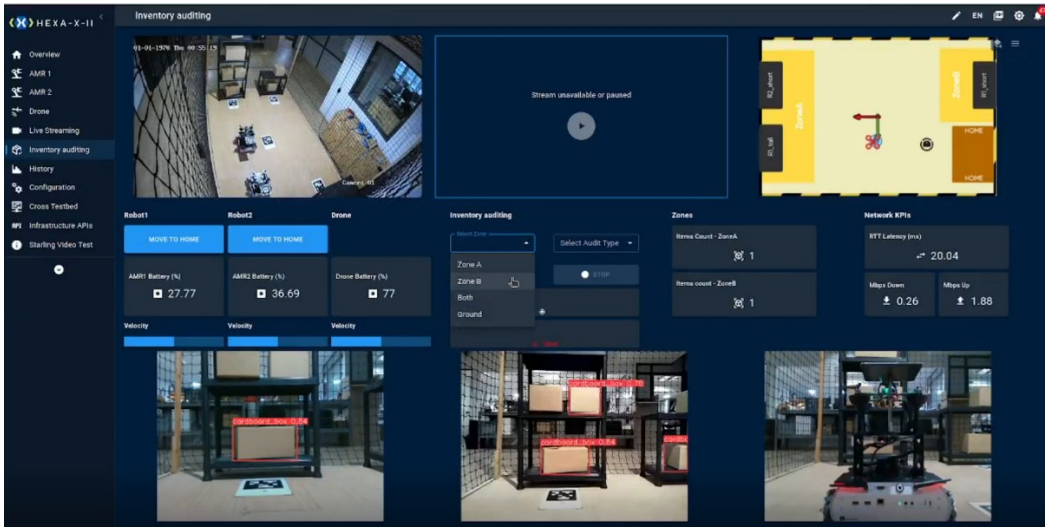


Figure 4-38 Warehouse Inventory Management Application User Interface [wi-SUPPLY].

Connectivity recovery is robustly addressed through an onboard network switching service. This service maintains a preconfigured list of network interfaces and automatically transitions to alternative networks when a disruption is detected. In scenarios where the Flextop UAV is active, robots remain in an idle state at their last assigned tasks until connectivity is re-established, thereby minimising downtime. Additionally, the system provides real-time alerts and logs on the UI, ensuring that any connectivity issues are promptly identified and addressed. The dynamic reconfiguration capabilities, combined with energy-efficient operation, result in sustained throughput and minimised energy consumption. Furthermore, the use of flexible topologies ensures that the network can adapt in real time to changes in the operating environment, thereby supporting uninterrupted and efficient inventory management. A series of controlled deployments were conducted to evaluate the end-to-end performance of the warehouse inventory management architecture featuring a varying number of ground AMRs and multiple Flextop UAVs. Tasks were allocated according to real-time heuristics and current network availability, while the Flextop UAVs was positioned to restore connectivity whenever base-station links were insufficient. As seen in Figure 4-39, direct connections to the base stations consistently showed lower latency variance, whereas UAV-extended links offered broader coverage at the expense of slightly higher and more fluctuating latency. This trade-off allowed even remote zones of the warehouse to be served effectively, and no major service interruptions were recorded.

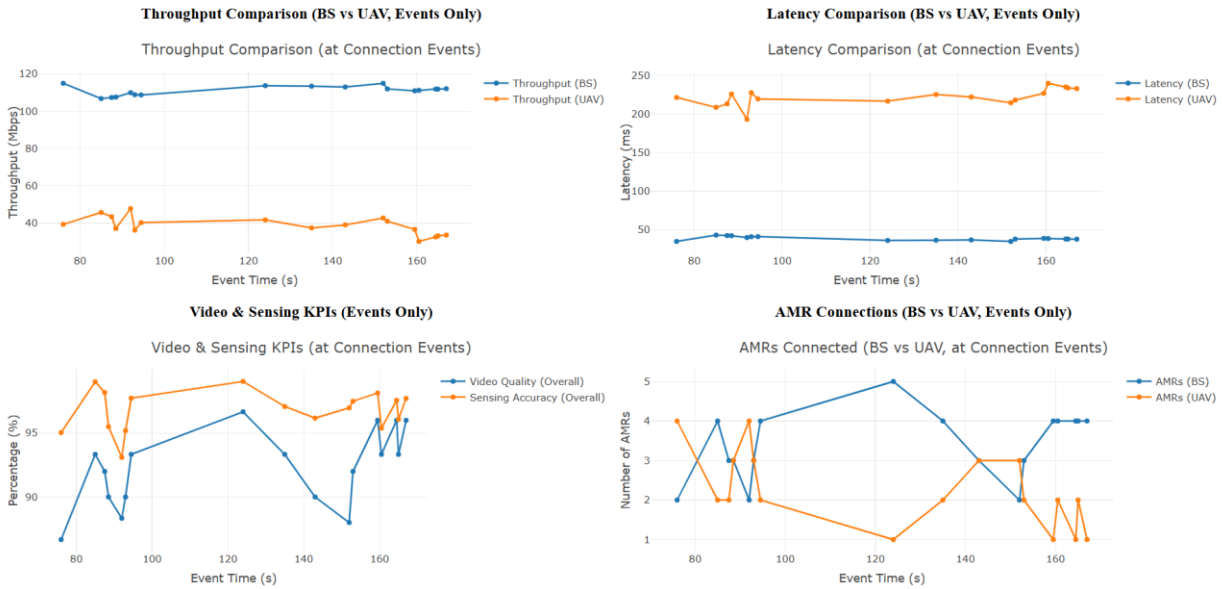


Figure 4-39 Performance Evaluation of Warehouse Inventory Management Architecture with Ground AMRs and Flextop UAVs

Additionally, video transmission quality was found to be near optimal for AMRs connected directly to a base station, while streams routed through UAV relays exhibited moderate degradation whenever network topologies were forced to reconfigure under load. Nonetheless, significant task failures were not observed, since the system's resilience mechanisms promptly resumed inspection procedures once links had been restored. These outcomes underscore the benefits of combining BCS exposure with dynamic, trustworthy topologies. Overall, an adaptive level of coverage and performance was maintained across varying conditions, demonstrating that robust connectivity management can be achieved in a complex, multi-robot industrial setting.

To further validate the robustness of this architecture in diverse connectivity scenarios, two complementary setups were tested using Flextop UAVs connected to a 5G SA network:

- In Setup 1, the drone remained connected to the 5G SA network directly and was periodically taken in and out of coverage every 2.5 minutes. As shown in Figure 4-40, the power consumption (orange line) generally ranged between 4 W and 10 W and exhibited distinct spikes during the reconnection intervals (highlighted in red). The latency plot in the lower panel of Figure 4-40 shows that these spikes (reaching up to 175 ms) coincided closely with times the drone was leaving or re-entering coverage. Similarly, the shaded green and blue regions depict the upload and download throughputs respectively, where noticeably high throughput segments tended to align with higher power draw.

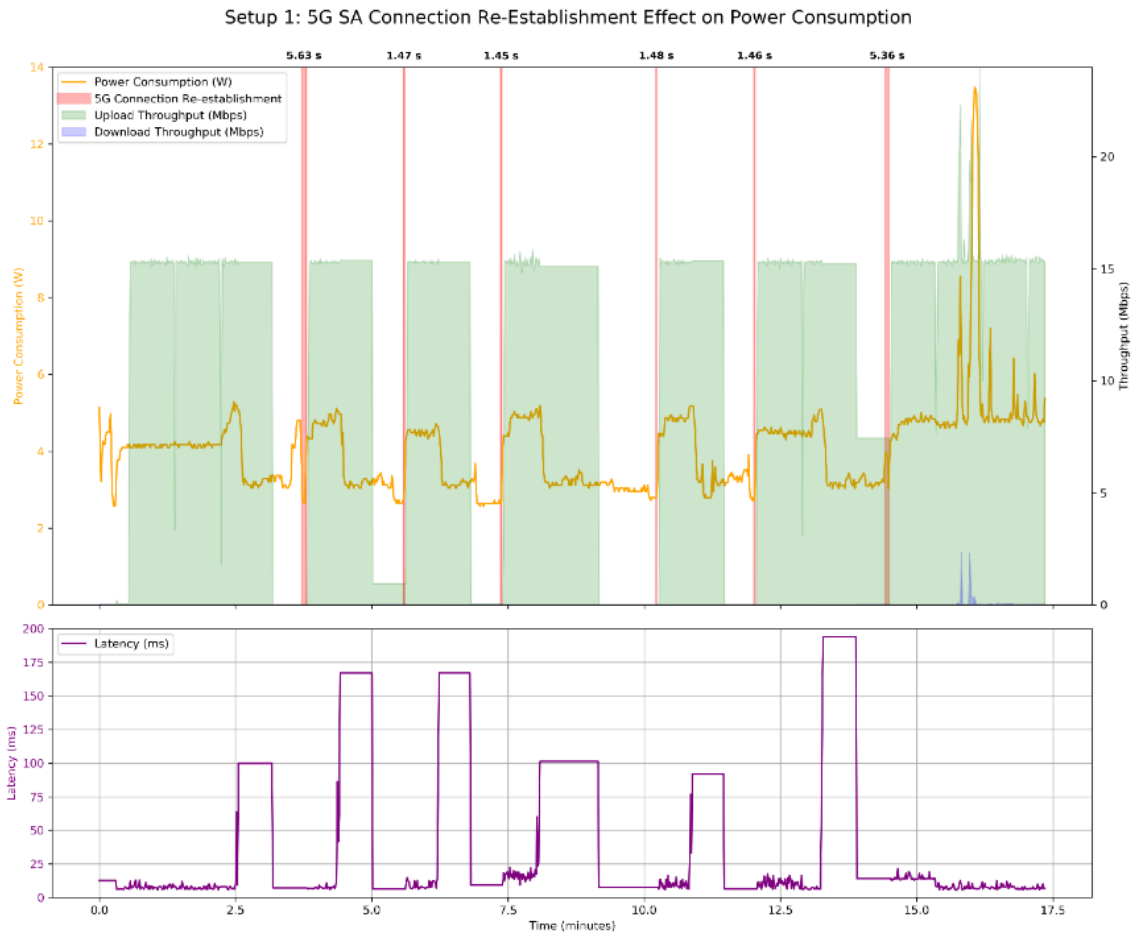


Figure 4-40 Experimental Setup 1: Power and latency variations during periodic coverage loss in direct 5G SA connection

- By contrast, in Setup 2, the drone switched between two Wi-Fi networks that each backhauled to the 5G SA network. As shown in Figure 4-41, the power consumption stayed consistently lower—typically between 2.6 W and 3.8 W. However, the red-shaded reconnection intervals (ranging from roughly 6.5 s to 20.4 s) indicate that network re-establishment required significantly more time compared to Setup 1. The bottom panel of Figure 4-41 also reveals more frequent but generally lower-magnitude latency spikes, reflecting the overhead of Wi-Fi handovers rather than abrupt coverage loss.

Figure 4-41 Experimental Setup 2: Power and latency behavior during Wi-Fi handovers with 5G SA backhaul

Overall, Figure 4-40 and Figure 4-41 illustrate the key trade-offs between the two setups: Setup 1 offers faster reconnection but higher power usage and larger boundary-related latency spikes, whereas Setup 2 operates more efficiently in terms of power but incurs a longer network switching overhead.

4.4.11 Intent-based Service Provisioning with Integrated Closed Loop Deployment and Reactivity for a Cobot Service Migration

Figure 4-42 reflects the performance of a 5G-powered VR digital twin system utilizing robots for real-time SLAM point cloud data transmission presented in section 3.3.6. The graph primarily shows a relatively stable throughput, indicating a generally consistent 5G connection. However, the graph also reveals noticeable fluctuations, including dips and peaks in throughput. These variations likely correspond to dynamic changes in the system's operation, such as increased data processing loads, robot movements. Overall, the graph demonstrates that while the system maintains a reasonably stable connection suitable for real-time data transmission, there are also periods of variability. These fluctuations emphasize the importance of continuous network monitoring and optimisation to ensure consistent and reliable performance for the VR digital twin system.

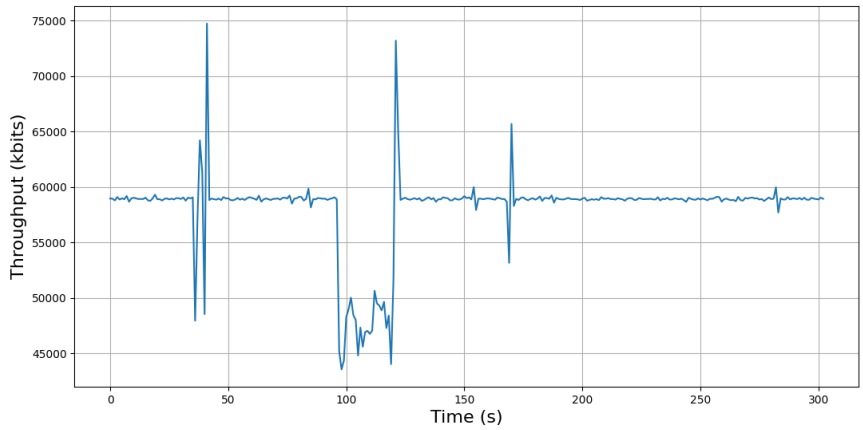


Figure 4-42 LiDAR point cloud uplink throughput.

This Figure 4-43 depicts the uplink delay, measured in milliseconds (ms), of a 5G connection over time. The data reveals a relatively stable baseline delay around 35-40 ms, indicating generally low latency. However, significant spikes in delay are evident, particularly around 50, 100-120, and 150 seconds, with some reaching above 80 ms and even 100 ms. These latency spikes pose a potential challenge for real-time applications like the described VR digital twin system, where responsiveness is crucial. While the baseline latency supports a degree of real-time interaction, the observed spikes could lead to noticeable delays in rendering and interaction, negatively impacting the user experience. This highlights the need for continuous monitoring and network optimisation to minimise latency and ensure a seamless, immersive experience.

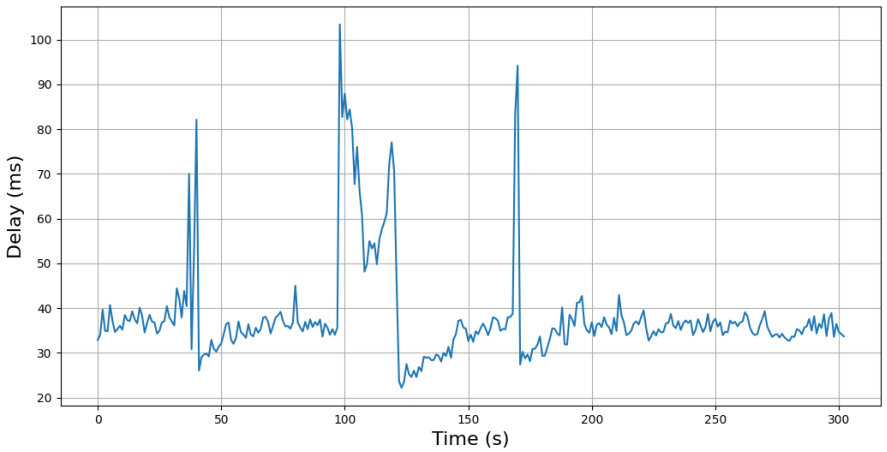


Figure 4-43 Lidar point cloud uplink delay.

As described in the section 3.3.6, several enhancements have been made to the monitoring platform and the closed-loop analysis and decisions functions. In particular, for what concerns the enhancement of the monitoring platform, further data sources have been collected. In the first stage, only the cobots' battery has been collected, while in the second stage, the position and orientation of the cobot have been collected too. Figure 4-44 shows raw data about the cobots battery, position and state.

```
[2/5/2025] [8:52:12 AM] › payload: {"header": {"seq": 265, "stamp":
{"secs": 1738745532, "nsecs": 735625505}, "frame_id": "map"},
"pose": {"pose": {"position": {"x": 3.719723381739802, "y": 5.308382867763124,
"z": 0.0}, "orientation": {"x": 0.0, "y": 0.0, "z": -0.9880643061429506, "w": 0.15404196482208896}},
"covariance": [0.012494956409425434, 0.001651613664800102, 0.0, 0.0,
0.0, 0.0, 0.001651613664800102, 0.027671285993083425, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.007823605366670 46]}}

[2/5/2025] [8:52:13 AM] › payload: {"timestamp": 1738745533.19488,
"node_id": 1, "voltage": 12.972, "percent": 93.8, "overall_capacity": 84}

[2/5/2025] [8:52:13 AM] › payload: {"timestamp": 1738745533.336145,
"node_id": 2, "voltage": 12.036, "percent": 72.5, "overall_capacity": 84}

[2/5/2025] [8:52:13 AM] › payload: {"data": "reached D"}

[2/5/2025] [8:52:13 AM] › payload: {"data": "State: ReturnToStation"}
```


Figure 4-44 Cobots' battery status, position and state raw data.

Specifically, on February 5th at 8:52 AM (GMT), the cobot number one has a battery level around 93%, in a specific position in a bidimensional space (x and y coordinates) and orientation toward west by around 0.15 degree. Finally, the cobots state is "State: ReturnToStation", meaning that the cobot has accomplished the mission and has returned to its station. Moreover, the cobots can support other states:

- IdleOK means that the cobots are doing nothing, so they do not move, and they are not executing any kind of action
- ReturnToStation means when they have either completed the mission or their battery level is low, so they are executing some back-home procedure
- IdleError means some error occurs
- Surveillance means they are correctly executing the mission
- "Reached a point", meaning that a point has been reached

As mentioned previously, an enhanced AI-enabled closed-loop analysis function has been implemented, exploiting the Adaptive Neuro-Fuzzy Inference System (ANFIS) developed under WP6 and reported in [HEX225-D65] for analyzing the input data provided by the monitoring function. The ANFIS was trained using offline cobot data, including quaternion cobot position, battery status, and cobot state. Using these inputs on labelled data, the ANFIS was trained to determine the optimal handover of a running cobot service application to reduce service downtime and maximise available cobot battery resources. The service exposed an API endpoint for inference requests; the analysis function was configured by the closed-loop governance at the time of instantiation with this endpoint.

A small enhancement was also made on the decision, which used a sliding window approach to weight the inference received by the analysis stage to avoid false triggers due to the systematic variation in battery level provided by the cobots while running.

The described enhancements of the closed-loop monitoring, analysis, and decision functions have been validated with the cobots via integration tests. Similarly to the first integration stage, both closed-loop functions have been deployed within the VTT testbed, specifically to a K8S cluster. At run-time, two cobots started continuously sending their information about battery, position and status. One cobot began the surveillance, moving around one area. In the meantime, the closed-loop monitoring function gathered and stored these data; consequently, the AI-enabled closed-loop analysis function analysed the provided cobot metrics via an ANFIS hosted as a containerized service located on the same cluster as the closed-loop components. The received inference was then passed to the decision function. The active cobot ran the surveillance until the closed-loop decision function made the decision to migrate the service. The execution function then triggered the application migration via the Service Orchestrator. The dashboard for the closed-loop service can be seen in Figure 4-45, which displays the incoming percentage and position data from the monitored cobots, the

inference from the ANFIS, and the event information from the decision and execution functions at the time of migration request.

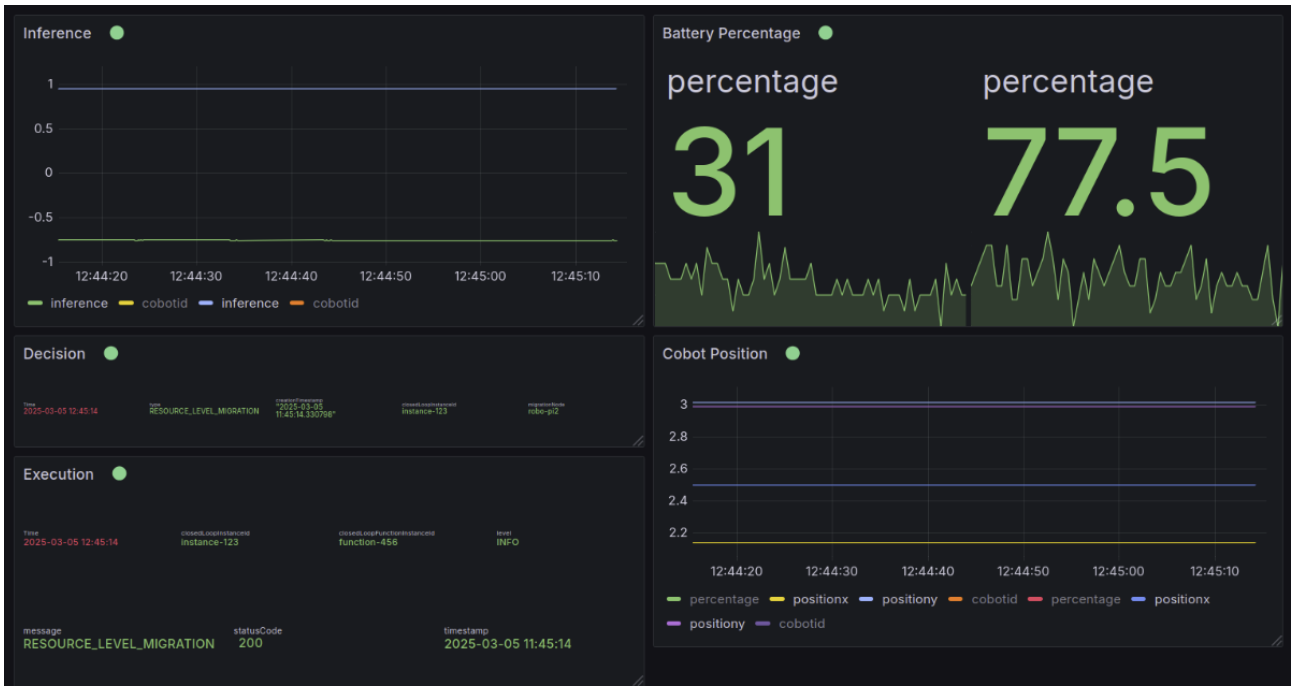


Figure 4-45 Dashboard for the cobot closed-loop service at the time of service migration.

4.4.12 Synergetic monitoring and orchestration

The synergetic scenario as described in Section 3.3.5, brings together different enablers to implement a realistic E2E workflow. For evaluating the scenario, two key aspects are considered, Service Orchestration and Intent-based Networking. These are discussed below.

4.4.12.1 Service Orchestration

As discussed in Section 3.3.53.3.5, the SP must balance specific factors to minimise cost while ensuring SLA compliance. The total cost is described in the following cost function:

$$C_{tot} = (1 - x) \cdot C_{pow} + x \cdot C_{cloud}(PL) + x \cdot C_{net}(NL)$$

where:

- x be a binary decision variable, where x=1 means deploying in the cloud, and x=0 means on-premises deployment.
- C_{pow} be the power consumption cost function, *power* is the measured power consumption in Watts.
- C_{cloud} be the cost of using cloud resources, depending on the chosen Processing Latency (PL) tier. Set by the CP.
- C_{net} be the cost of using the network, based on selected Network Latency (NL). Set by the NP.

The observed total latency must satisfy the SLA:

$$L_{tot} \leq SLA$$

where L_{tot} be the total observed service latency, which includes processing and network delays.

The experimental setup considers the following metrics for evaluating the deployment: service latency (s), frame transmission duration (s), inter-cluster network round-trip time (RTT) (ms), service replicas, power consumption (W), domain (on-premises/cloud). For the deployment's optimal configuration, the aforementioned cost function and the corresponding constraints were used. For the cost of power, cloud offloading and network services the following functions were used:

$$C_{pow}(power) = 3 \cdot power$$

$$C_{cloud}(PL) = -7 \cdot \frac{PL}{100} + 42$$

$$C_{net}(NL) = -7 \cdot \frac{NL}{10} + 42$$

where PL={100,200,300,400,500} ms, NL={10,20,30,40,50} ms and power as measured in watts.

The optimiser was executed for multiple values of the SLA threshold to identify the values for which the on-premises infrastructure is enough to satisfy the latency requirement, since the local server in this scenario has low resource availability and is unable to sufficiently handle deployment for low latency requirements. The results are shown in Figure 4-46.

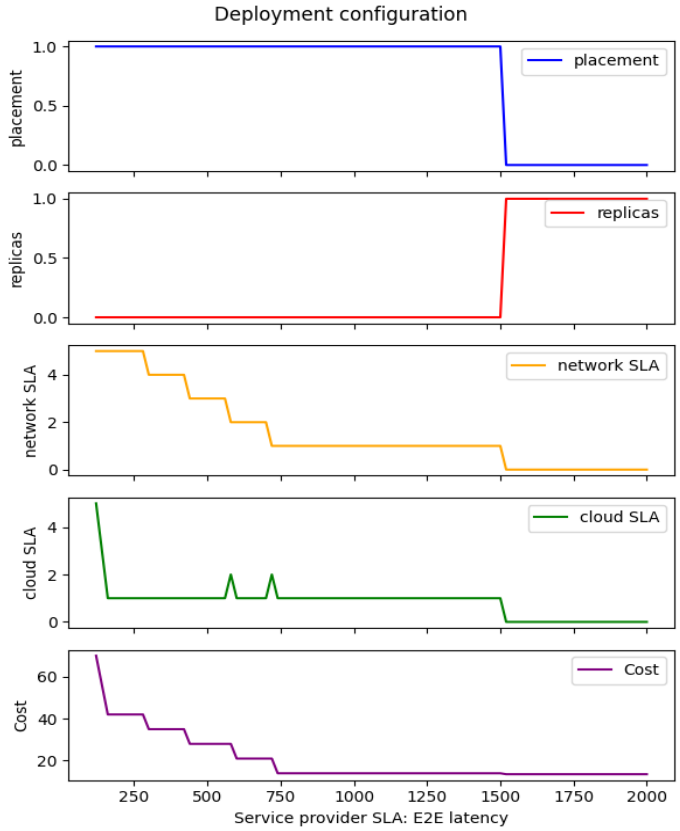


Figure 4-46 Deployment configuration parameters across latency constraints.

We observe that as the SLO threshold increases network and cloud SLAs are relaxed since lower tiers can support the service's execution. For E2E latency SLAs above 1.5 s the local server seems to be able to support the service execution.

We evaluate the setup over a week's period considering a latency SLA of 1 s for 80% of the time. The deployment optimiser selects a new deployment configuration every 4 hours based on resource availability and performance. The measured E2E latency and the corresponding cost are illustrated in Figure 4-47.

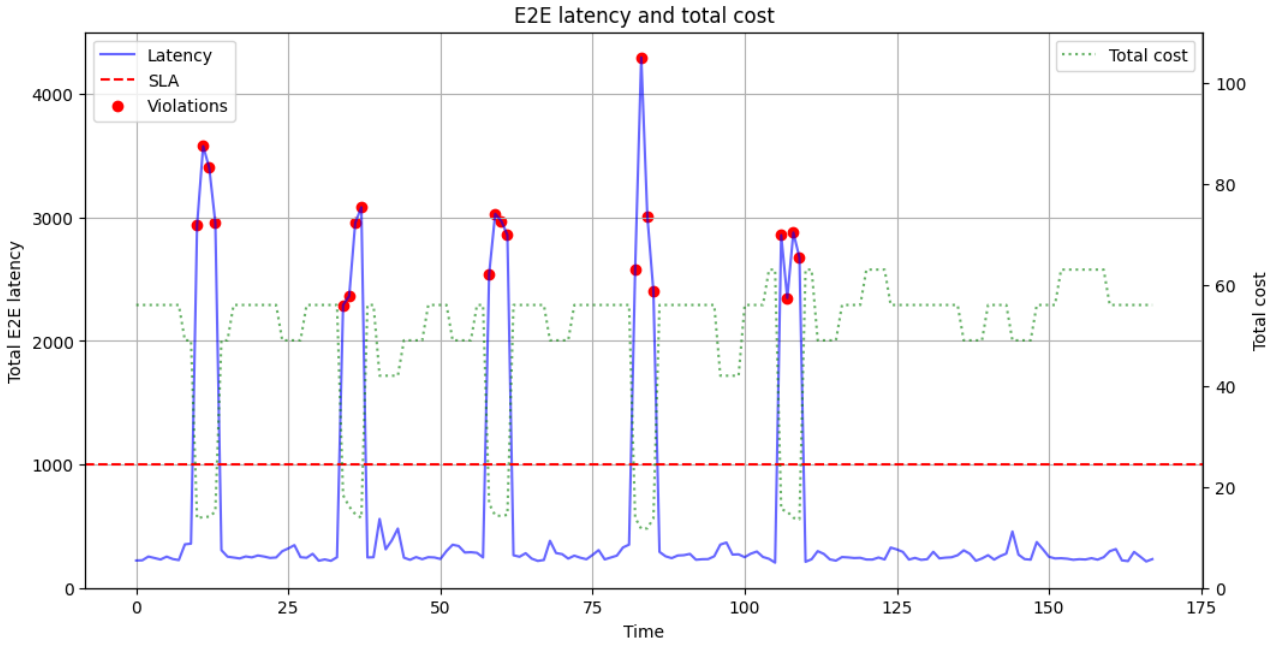


Figure 4-47 Synergetic scenario performance.

We observe that the latency constraint is generally satisfied with the exception of the peak hours where the remote server is not available and the service is migrated locally. In this case, since the server is overloaded latency is significantly higher and it would require additional resources to keep below the threshold. The SLA's percentile satisfaction in the rolling window (24 hours) is more than 83% of the time above the threshold, so the SLA still holds. The total cost is also demonstrated by the greed-dotted line. Notice that the cost drops significantly in the intervals where the SLO is not satisfied. The analysis of the sub-costs are shown in Figure 4-48.

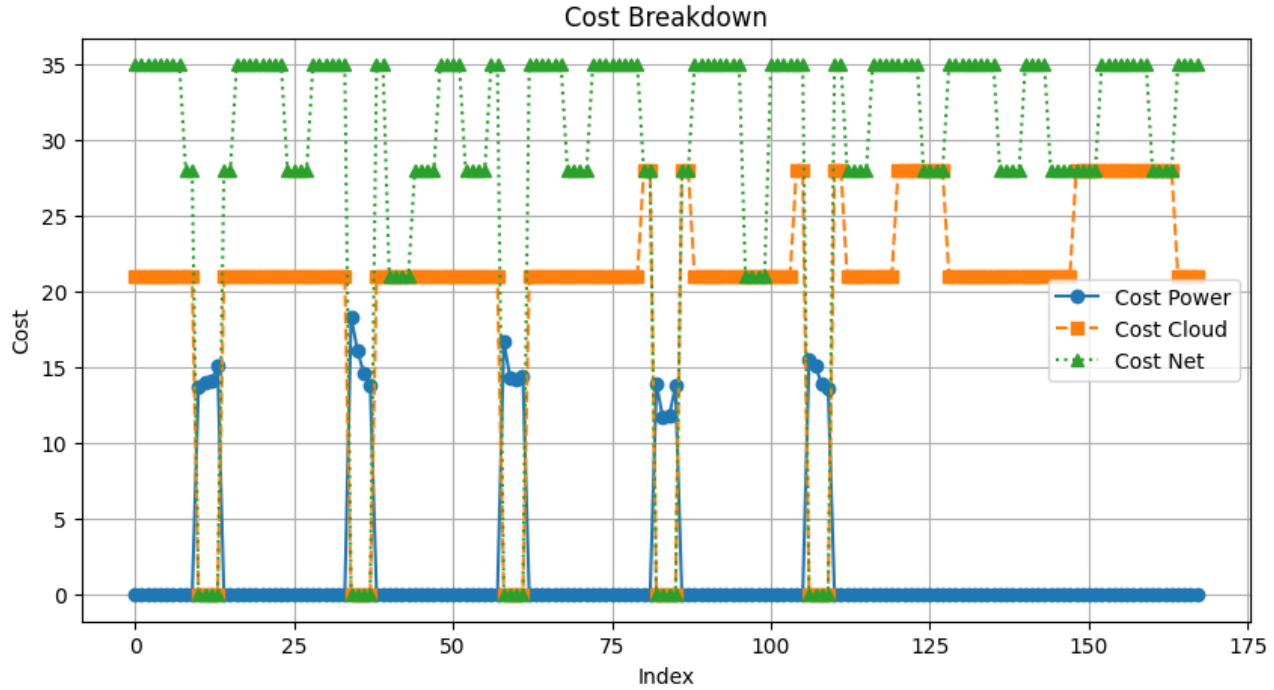


Figure 4-48 Synergetic scenario cost analysis.

Notice that as CP and NP costs decrease at the local deployment cases and the power cost increases. Compute and network thresholds (and thus, also the costs) fluctuate at the cloud deployment cases, keeping a minimal threshold as requested by the deployment optimiser for satisfying the SLA.

Additionally to the orchestration performance, the migration overhead introduced was measured to identify service drops due to the overlapping intents.

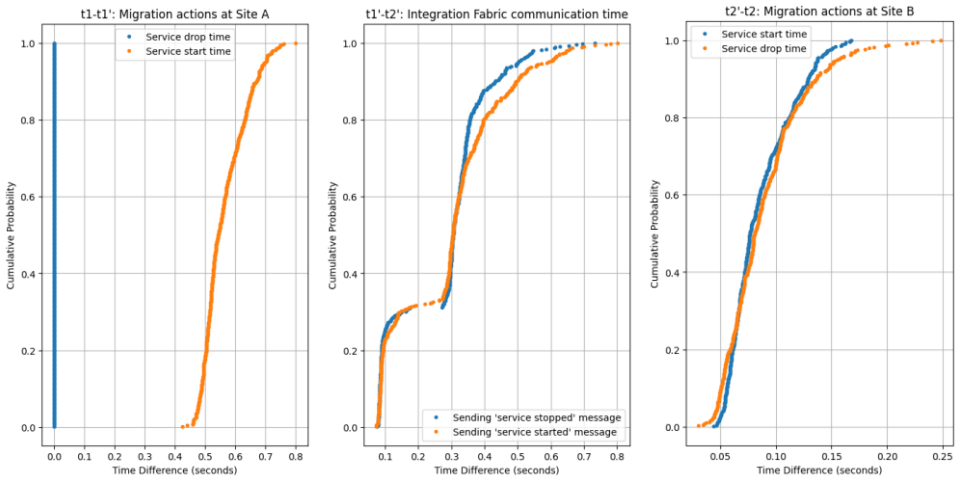


Figure 4-49 Migration delay Site A / Site B operations

As shown in Figure 4-49, messages to the Integration Fabric demonstrate average performance close to 100 and 300 ms, and in the worst case 800 ms. Service start is measured around 550 ms for Site A, while service drop is measured zero since the message to start the service at the local service is sent before actually dropping the service so that the drop time is minimal. In an extension of the scenario, a mechanism for assuring the service downtime to zero could be used. For Site B, service start and drop demonstrate similar duration and smaller than Site A, as expected since it is a lightweight server.

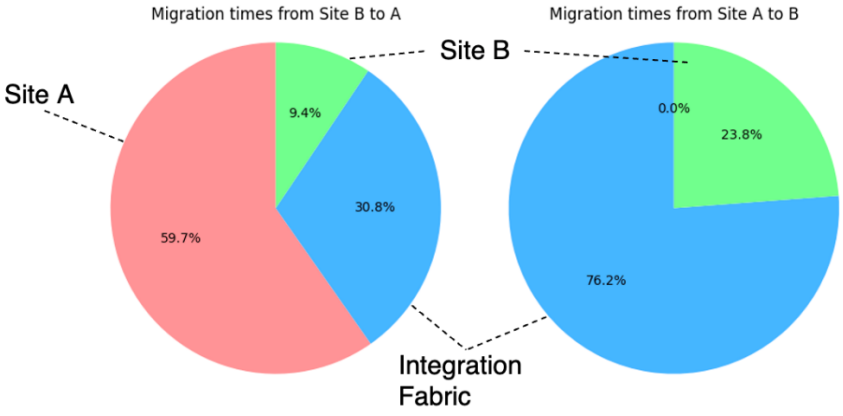


Figure 4-50 Migration delay local / remote deployment.

Site A operations when deploying the service at the cloud seem to greatly increase downtime (Figure 4-50), by 150%, and thus, should be considered in cases where SLAs include downtime requirements.

4.4.12.2 Intent-based Networking

The previously described solution has been executed with a set of 20 intents and the main results in terms of execution times were gathered in the graphs illustrated in Figure 4-51.

The graph (a) in Figure 4-51 presents the Cumulative Distribution Function (CDF) values for the total time implementing the solution, whereas the graph (b) on the right-side illustrates the relative values taken in each of the five actions that encompass the solution: to interpretate the received intent request, to stare the generated intent in the database, to apply the feasibility check process, to generate the TFS call, and finally, to publish the generate TFS call within a Kafka message.

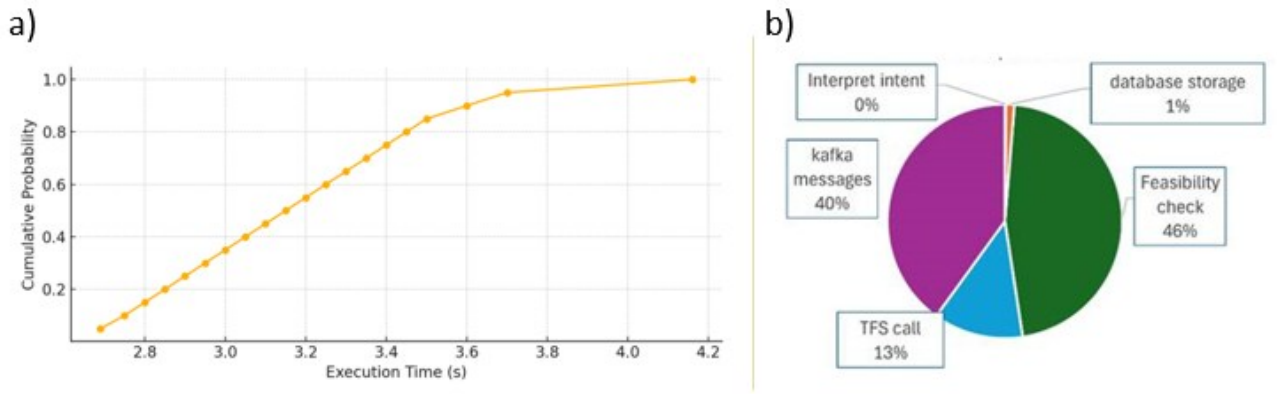


Figure 4-51 Total execution time CDF (a) and relative values for each task execution time (b).

The CDF -graph (a)- of Figure 4-51 presents the number of seconds taken to carry out the whole functionality (i.e., X axis) and their corresponding probability (i.e., Y axis). Based on these results, it is possible to appreciate a low variability of the data samples with one outlier and a range of values from a minimum of 2.69 s and a maximum of 4.16 s. The low variability of the samples can be attributed to the execution of actions like intent interpretation, database storage and the feasibility check which present a low variability in their respective tasks. However, the process of triggering the TFS call and the send of Kafka message, although they still occur in a short period of time, they present a bigger variability since the execution of those tasks rely on external entities.

In parallel, graph (b) in Figure 4-51 the whole functionality time is divided and the average of the relative percentage values of each task are presented. It is worth considering that the execution of the PoC occurs in a very short period with an average of 3,10 s to be executed, which means that a small change in terms of time implies a big change in the percentage distribution. However, it is also worth considering that the biggest execution time was in the tasks that were performing the feasibility check (46%) and sending the Kafka message (40%) with a total of the 86% of the time. That is because the algorithm relies on external entities which slightly delay its performance time. On the other hand, the database storage (1%) and the intent interpretation (< 0%) took a marginal proportion of the execution time, followed by triggering the TFS request which barely took an average the 13% of the time.

4.4.13 Evaluation of the Integration fabric Adapter implementation

Section 3.3.6.2 has described the high-level workflow for the intent-based service provisioning with the Integration fabric mediation. This subsection focuses on the results of the implementation, estimating the latency overhead of adopting this approach.
The Integration fabric Adapter has been implemented, integrated, and validated against Telefonica's integration fabric. As already described, the service provisioning workflow consists in two stages, i.e. the subscription stage executed once, and the service provisioning stage executed as many times as the IBN-IME requests the service provisioning.
In detail, the subscription stage consists of the Integration fabric adapter and the IBN-IME to authenticate and then be authorised before either subscribing to a topic or publishing messages. This is realised by first obtaining a certificate from integration fabric and then using it for creating the topic and executing operations on that, i.e. publishing and consuming messages.

In the service provisioning stage, the Integration fabric Adapter receives the notification and starts the service provisioning workflow as previously described. Figure 4-52 depicts the logs generated during the service provisioning stage by the Integration fabric Adapter.

```
3   Message received, creating new instance
4   2025-03-13 10:26:32.387673 Delta time: 0
5
6   {
7       "reqId":"9724fc1e-328c-4ed0-93ce-97374e2e30eb",
8       "serviceInstance":
9           {
10          "vsbId": "c668b87c-e25c-41a7-8c96-1d1d3bd158cb",
11          "vsdId": "",
12          "name": "nginx Instance",
13          "description": "new nginx instance",
14          "inputparams": {
15              "nginxtest.NODEPORT": "31000"
16          }
17      }
18  }
19
20  Response received from soccer
21
22  2025-03-13 10:26:32.573690 Delta time:  0:00:00.186047
23
24  Service ID: 34efd8df-671e-415d-9660-04f652af8a0d
25  2025-03-13 10:26:32.573792 Delta time:  0:00:00.000102
26
27  {"reqId":"9724fc1e-328c-4ed0-93ce-97374e2e30eb", "status":"CREATED"}
28  Message: b'68c2f71c-287d-49c4-a3ef-4dc3131c7c54' successfully produced to Topic: test.nextworks
29  2025-03-13 10:26:32.710040 Delta time:  0:00:00.136248
30
31  Response after instantiating
32  2025-03-13 10:26:32.767842 Delta time:  0:00:00.057802
33
34  Instance stauts changed in: INSTANTIATING
35  2025-03-13 10:26:32.826376 Delta time:  0:00:00.058534
36  Sent notification through kafka
37  {"reqId":"9724fc1e-328c-4ed0-93ce-97374e2e30eb", "status":"INSTANTIATING"}
38  Message: b'eaddc184-6e17-4f7d-8f28-0f7cb716361e' successfully produced to Topic: test.nextworks
39  2025-03-13 10:26:32.868912 Delta time:  0:00:00.042536
40
41
42  Instance stauts changed in: INSTANTIATED
43  2025-03-13 10:26:39.056986 Delta time:  0:00:06.188074
44  Sent notification through kafka
45  {"reqId":"9724fc1e-328c-4ed0-93ce-97374e2e30eb", "status":"INSTANTIATED"}
46  Message: b'845038e8-e7bc-4d82-b820-2d3502227e9c' successfully produced to Topic: test.nextworks
47  2025-03-13 10:26:39.106707 Delta time:  0:00:00.049721
```

Figure 4-52 Integration fabric Adapter logs during the provisioning stage.

From rows 6 to 18, the Integration Fabric adapter shows the log about request receival to provision a service. Specifically, the message contains the service instance request, including the vertical service blueprint identifier (vsbId entry in the JSON, the name of the service instance (nginx instance) and the input parameters, i.e. the nginx nodeport. Afterwards, the request is processed and forwarded to the service orchestrator. During the provisioning stage, the service goes through different statuses: CREATED, INSTANTIATING and eventually INSTANTIATED. All these statuses are notified to the IBN-IME through the Integration fabric, whose logs are reported at rows 27, 34 and 42, respectively.

In particular, the Integration fabric Adapter maps the request from IBN-IME with the service instance with within the service orchestrator (the identifiers are reported at row 24 in the Figure 4-52).

The asynchronous approach adopted by the Integration fabric Adapter and the Integration fabric can introduce an overhead in terms of latency because of the procedure itself. For measuring this overhead, several integration tests have been performed, measuring the time of publishing the notification on the Integration fabric. In particular, to have a significant sample, the service provisioning stage has been repeated 20 times, for a total of 60 notifications sent from the Integration fabric Adapter to the Integration fabric.

With the current Integration fabric adapter implementation, it takes, on average, around 0.133 seconds to publish a single notification related to the service status to the Integration fabric. This means introducing an average overhead of 0.4 seconds for all three notifications related to the service status. This average overhead depends on the network delay and Integration fabricAdapter processing time. On the IBN-IME side, there would also be some delay in publishing the message related to the service request and receiving the notification. The exact estimation of this delay strictly depends on the implementation of the IBN-IME, but in any case, it should be in the same order of the Integration fabric Adapter delay, i.e. a few tenths of milliseconds per message.
In conclusion, the realisation of the asynchronous approach relying on the Integration fabric adapter implementation introduces an estimated total overhead of less than one second. This overhead can be considered acceptable because it does not concern real-time or time-sensitive applications and does not affect at all the overall logic of the service provisioning.

4.4.14 Sustainability and trustworthy-oriented orchestration in 6G

This section provides key simulation results of the physical task planning FA mechanism and the time-related measurements of the trust management system, both integrated into the cobot-powered inventory management warehouse PoC and described in Section 3.3.1.

The experiments aimed to evaluate the performance of the physical task planning FA algorithm using a simulated environment with 30 heterogeneous virtual AMRs and UAVs executing vision-based tasks across a 20×20×4 m warehouse. The algorithm was tested against a nearest-neighbour baseline under varying task loads. Figure 4-53 illustrates the total energy consumption and the energy saving achieved using the ACO-based physical task planning FA algorithm, compared to the baseline nearest neighbour heuristic for increasing number of tasks (from 10 to 120). Figure 4-54 shows the total duration time of completing all the tasks and reduction of duration time. The results demonstrated up to 35.9% energy savings and 60% reduction in task completion time, highlighting significant gains in energy efficiency and scalability, particularly when sufficient robotic resources were available. However, performance gains decreased as robot availability dropped due to task saturation.

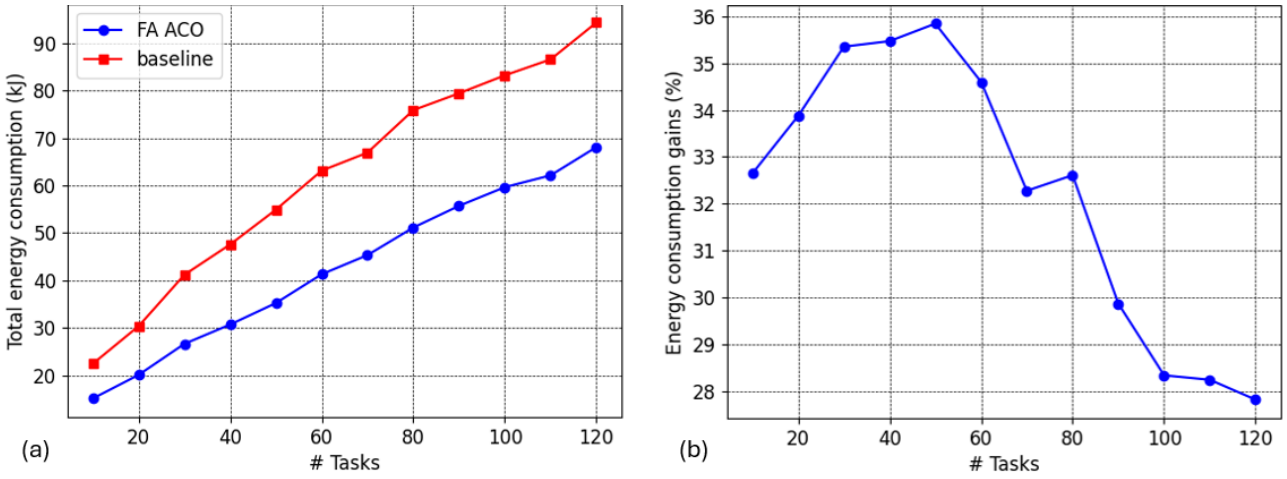


Figure 4-53 Total (a) energy consumption and the gains (b) using the physical task planning FA algorithm compared to the nearest neighbour heuristic.

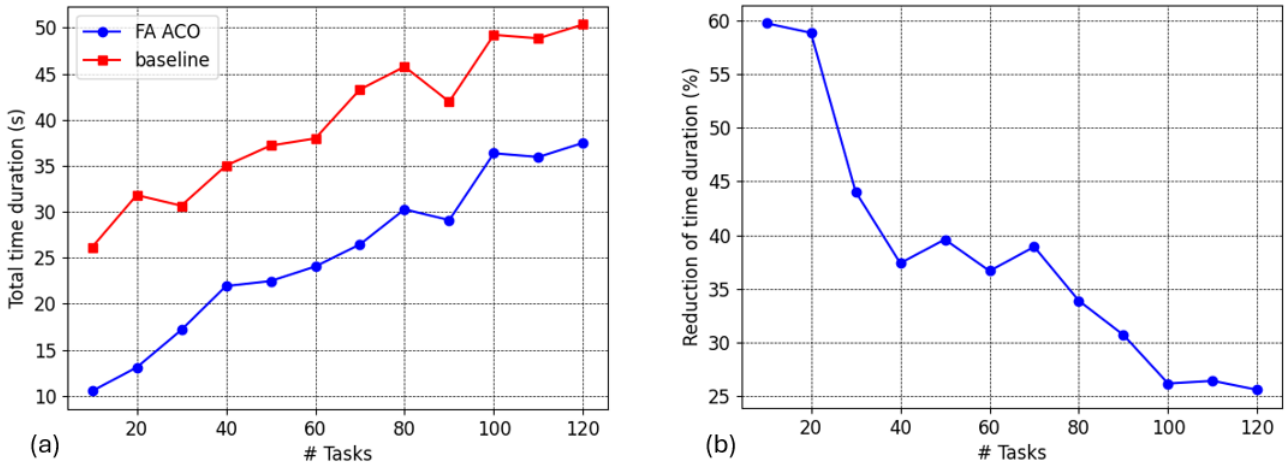


Figure 4-54 Total (a) duration time and the reduction (b) using the physical task planning FA algorithm compared to the nearest neighbour heuristic.

The trust management system, responsible among others for assessing the trust indexes of the compute nodes of the system interacts with both TEF and LoTAF for improved trustworthiness estimation communicating through MCE-integration fabric. Some of the time-related results of this integration are reported in Table 4-2. These measurements represent the average values obtained from 26 repetitions of the message sequence diagram operation show in Figure 3-2.

Table 4-2 Time-related results of trust management (utilising MCE).

Publish trust management request to MCE	~ 0 s	Publish TEF output to MCE	~ 0 s
Consume request from TEF	0.0402 s	Publish LoTAF output to MCE	0.0746 s
Consume request from LoTAF	0.0321 s	Consume TEF and LoTAF output	2.0250 s
TEF execution time (including infrastructure monitoring communication)	0.3247 s	Execution time of trust management calculations	0.0001 s
LoTAF execution time	0.0002 s	E2E latency (from publishing trust management request till its final calculations)	2.7369 s

4.5 Impact

This section focuses on the impact of the Hexa-X-II PoC-driven validation activities, highlighting aspects related to the Open API and Open-source contributions of the PoC components.

4.5.1 Open APIs in System PoC / Component PoCs

The following table shows the OpenAPIs available in the context of the different PoC, together with their usage and the link to the public repositories where they are available.

Table 4-3 Open APIs Specification

Title	SW Origin	Licence	Available in	PoC / Demo	Integration/usage context
DLT Service Federation Open API	New - Created in project	CC Attr. 4.0 International	[DLTUC324]	PoC B.1	The interfaces are used in the context of the Federated orchestration system
Monitoring jobs configuration API	Existing - Enhanced in project	Apache 2.0	[MonNXW24]	PoC B.1	Used for configuring the monitoring platform for retrieving the cobots data (battery, position and status)
Query historical monitoring data API	Existing	MIT	[InfluxDB]	PoC B.1	Provides time-series lookups on previously collected system/cobot metrics. Used by orchestration or AI modules for improved resource placement and scheduling decisions.
Multi-cluster extreme-edge resource orchestration API	New - Created in project	Apache 2.0	[MulNXW24]	PoC B.1	Enables orchestrators to manage services across multiple Kubernetes clusters, including edge clusters. In the PoC, it supports dynamic scaling and AI workflow migration between edge and cloud to meet low-latency requirements.
Closed Loop Governance – Catalogue API	New - Created in project	Apache 2.0	[CloNXW24]	PoC A, PoC B.1 PoC C	Allows for managing the applications that support the different stages of the CLs. In the PoC, it is used to onboard the specific CL functions required for the automatic control of the cobots' batteries and related remediation
Closed Loop Governance – Lifecycle Management API	New - Created in project	Apache 2.0	[LCMNXW24]	PoC A, PoC B.1 PoC C	Allows the consumer to request lifecycle operations on a given CL. In the PoC, it is consumed by the security orchestrator to request the provisioning of a CL specific for the cobot's service automation based on the battery level
Integration Fabric API	New– Created in project	CC Attr. 4.0 International	[NIntegr24]	PoC C	Provides three API endpoints used in the context of PoC for: • Retrieve keystore for the SSL connectivity with the Kafka Cluster • Listing the communication channel, onboarded entities

					• Onboard and offboard component and create and delete communication channels
Level of Trust Assessment Function API	New – Created in project	GPL 3.0	[VLoTAF24]	PoC C	Provides real-time trust metrics from compute nodes. These trust indexes guide the trust-aware orchestration functions.
Trust Evaluation Function API	New – Created in project	GPL 3.0	[LTEF25]	PoC B/C	Allows retrieving trust indexes after analyzing infrastructure data. Integrated in the PoC B and C.
IBN-IME API	New – Created in project	Apache 2.0	[AMV+25]	PoC B/C	A subset of the OpenAPI calls were used in both PoC B and PoC C, in the context of the intent management procedures.

4.5.2 Open-source tools

The Hexa-X-II PoC activities have integrated various open-source tools to support simulations, network orchestration, security, distributed computing, and AI-driven management across different 6G validation scenarios.

OpenAirInterface (OAI) has been extensively used for simulations, particularly for evaluating RAN architectures. The OAI 5G virtual RAN implementation was leveraged to compare latency between split-RAN (DU/CU) and monolithic architectures, with the OAI RF simulator replacing real radio hardware to analyse F1 interface delays and latency variations. These simulations demonstrated that disaggregated RAN architectures introduce additional latency due to control signalling overhead. The tests were conducted in a virtualized environment using Intel i7-11850H, 16 cores, 32 GB RAM, Ubuntu 20.06 LTS, n78 band (3.5 GHz), and 40 MHz bandwidth.

For cloud-native orchestration and workload management, Kubernetes has played a critical role in managing distributed workloads, orchestrating AI-driven applications, and ensuring optimal service placement. It has been used to deploy collaborative distributed machine learning models, ensuring data locality and communication efficiency, while also supporting latency-sensitive service orchestration across edge and cloud clusters.

For observability and monitoring, Prometheus has been integrated to collect performance metrics, including CPU utilization, memory consumption, workload processing rates, and communication latency. It has been used alongside Karmada for multi-cluster service propagation and Zipkin for distributed tracing, helping to analyze performance bottlenecks and optimise system execution.

Within the MCE Framework, Apache Kafka has been adopted for secure data exposure, implementing mutual TLS authentication (mTLS) to ensure secure communication between network functions and orchestrated services. OpenAPI (v3.1.0) has been used to standardize service interactions, ensuring seamless integration of automation and orchestration components.

In the domain of security, Qujata has been extended with post-quantum cryptographic (PQC) schemes, including Kyber, Frodo, and HQC, to validate quantum-resistant encryption mechanisms. It has also been integrated with SPQR cluster for hybrid PQC evaluations, ensuring secure communication in post-quantum environments.

For network orchestration and programmability, ETSI TeraFlow SDN (TFS) has been employed to enable multi-domain slicing and transport network reconfiguration. It has been complemented by CockroachDB, which provides resilient distributed state storage, and OpenConfig, which enables automated network configuration management.

For real-time telemetry and performance monitoring, gNMI telemetry streaming has been used to enhance packet-optical network observability, providing real-time insights into network behavior. Additionally, Sionna, an open-source PHY-layer simulation library, has been integrated to support AI-driven spectrum optimisation and RAN performance evaluation.

4.5.3 How the end-to-end system evaluation results of the overall 6G system design address the project objectives

The end-to-end evaluation of System-PoC #C provides strong evidence of how the Hexa-X-II system design supports the broader vision of a sustainable, trustworthy, and inclusive 6G platform (Objective #1). By integrating innovations across multiple technology domains—radio, sensing, AI, and compute—the E2E system validation demonstrates the feasibility and impact of a holistic architectural approach (Objective #2).

One of the key outcomes is the validation of intelligent, adaptive connectivity enabled by next-generation radio and sensing technologies. Features such as AI-native air interfaces, ML-based CSI feedback compression, and flexible transceiver designs enhance spectral efficiency and reduce latency (Objective #3). These improvements support context-aware, dynamic service delivery and reflect significant advancements in 6G radio design, tailored to meet the stringent requirements of emerging use cases (Objective #2).

The platform also showcases the integration of compute and AI into the system fabric, advancing the notion of networks beyond communication (Objective #4). Capabilities such as split learning, multi-agent reinforcement learning (MARL)-based orchestration, and real-time compute offloading (e.g., XR rendering in the warehouse scenario) are validated through concrete workflows (Objective #4). These mechanisms demonstrate how distributed intelligence can be harnessed to dynamically process and interpret data across the continuum, supporting both efficient system management and operation (Objective #5), as well as novel digital services (Objective #4). Equally important is the emphasis on operational automation and system resilience. The system supports intent-driven orchestration and zero-touch management through exposed service APIs—some CAMARA-compliant—enabling vertical applications to interact with and adapt the network in real time (Objective #5). Coupled with AI-powered closed-loop control, these mechanisms reduce operational complexity and promote self-managing behavior (Objective #5), contributing to the sustainability and trustworthiness pillars of the 6G vision (Objective #1).

In the System-PoC's warehouse scenario, for instance, proximity-sensing devices with energy harvesting capabilities demonstrate how intelligent automation and green design can be combined. These devices enhance shelf monitoring while optimising energy use via intelligent wake/sleep cycles (Objective #4). This practical illustration reinforces the project's commitment to energy-efficient, inclusive solutions deployable across real-world domains (Objectives #1 and #4).

Finally, the System-PoC #C implementation and its evaluation framework reflect the project's commitment to evidence-based system validation (Objective #2). By combining realistic testbeds, synthetic traffic, and tightly coupled PoC components, the project offers a robust foundation for KPI and KVI assessment across all system layers (Objective #2). This aligns directly with Hexa-X-II's ambition to transition from foundational research to validated architectural blueprints ready for future deployment (Objective #2).

In summary, the E2E system validation consolidates the key technological advances developed in Hexa-X-II and confirms their alignment with the overarching goals of designing a sustainable, intelligent, and resilient 6G system—one that not only enables new services and adaptive infrastructure but also meets the broader societal and environmental challenges of the coming decade (Objectives #6).

5 Conclusions

The evaluation results and analyses presented in this deliverable confirm the successful validation of the E2E 6G system design as envisioned in the Hexa-X-II project. Through System-PoC #C and its supporting Component-PoCs, the project has demonstrated the feasibility, adaptability, and robustness of the 6G architecture, covering key aspects of radio innovation, novel devices, orchestration mechanisms, and system-level integration. This conclusive phase of evaluation completes the progression from conceptualization to

real-world experimentation and simulation-based validation, ensuring alignment with the overarching project objectives.

System-PoC #C extends the capabilities of prior iterations (System-PoC #A and #B) by incorporating enablers derived from 6G radio access, sensing, AI-assisted orchestration, and energy-efficient device technologies. By combining real-time radio performance (e.g., AI-native air interface, flexible modulation), intelligent resource orchestration (e.g., intent-based and closed-loop mechanisms), and sustainable device operation (e.g., energy harvesting IoT proximity sensors), the system exemplifies how a cohesive, modular, and forward-compatible 6G platform can address diverse application scenarios such as collaborative robotics, immersive XR, and autonomous system control.

The system-level validation results illustrate that the E2E 6G system can meet critical KPIs such as increased coverage, enhanced capacity, and reduced latency. Furthermore, it addresses KVIs including trustworthiness, inclusion, and sustainability by integrating mechanisms for trust assessment, energy-aware service orchestration, and multi-stakeholder support through OpenAPIs and standardized service exposure.

A notable highlight of the final System- and Components'- PoC evaluation is the interplay between networks beyond communications capabilities and real-time applications. The exposure of compute, sensing, and network state information at various architectural layers supports distributed intelligence and cross-layer optimisation, ultimately enabling resilient and efficient service provisioning. The collaborative robot use cases, particularly the cobot-powered warehouse inventory management and surveillance scenarios, serve as comprehensive demonstrations of how the Hexa-X-II system blueprint enables vertical-driven innovation.

The results also underline the importance of tight integration between radio components (WP4), device components (WP5), and system architecture and orchestration frameworks (WP3/WP6). The seamless orchestration across distributed infrastructure, from edge to cloud, ensures service continuity and system performance even under dynamic operating conditions. Additionally, the adoption of CAMARA-compliant APIs strengthens the potential for global standardization and cross-domain interoperability.

In conclusion, the Hexa-X-II E2E system evaluation validates that the proposed 6G architecture and its enabling technologies are technically sound, adaptable, and ready for future deployment scenarios. By delivering demonstrable impact across connectivity, computation, sustainability, and trustworthiness, the project contributes significantly to shaping a globally harmonized, inclusive, and intelligent 6G era. The insights and results presented herein provide a strong foundation for further research, standardization, and industry adoption of 6G technologies.

6 References

[28.312] 3GPP TS 28.312, “Management and orchestration; Intent driven management service for mobile networks,” v19.0.0, December 2024.

[38.401] 3GPP, “NG-RAN Architecture Description,” 3GPP TS 38.401 version 18.1.0 Release 18 March 2024.

[38.801] 3GPP, “Study on new radio access technology: Radio access architecture and interfaces,” 3GPP TR 38.801 V14.0.0, Release 14, March 2017.

[38.848] 3GPP TR 38.848, “Study on ambient IoT (internet of things) in RAN (Release 18),” 2023.

[AMV+25] Alemany, P., Muñoz, R., Vilalta, R., & Adanza Dopazo, D. (2025). Designed API for the IBN-IME solution used in Hexa-X-II T2.3/T2.5 work. Zenodo. https://doi.org/10.5281/zenodo.15125456

[CAMARA] The Linux Foundation, Camara project, https://github.com/camaraproject

[CloNXW24] Nextworks (Italy). (2024). Closed Loop Governance - Catalogue API. Zenodo. https://doi.org/10.5281/zenodo.14193484

[CNF25] C. Collmann, A. Nimr and G. Fettweis, "On the Impact of Phase Impairments on Angle Estimation in True-Time-Delay Systems," 2025 IEEE 5th International Symposium on Joint Communications & Sensing (JC&S), Oulu, Finland, 2025, pp. 1-6, doi: 10.1109/JCS64661.2025.10880644.

[DHK23] M. F. de Guzman, K. Haneda, and P. Kyösti, “Measurement-based MIMO channel model at 140 GHz,” https://doi.org/10.5281/zenodo.7640353.

[DSS04] M. Dorigo, T. Stu, and T. Sttzle. "Ant colony optimisation." (2004).

[DLTUC324] Universidad Carlos III de Madrid (UC3). (2024). DLT Service Federation API. Zenodo. https://doi.org/10.5281/zenodo.14033410

[HEX223-D21] Hexa-X-II project, “Deliverable D2.1: Draft foundation for 6G system design,” June 2023. [Online]. Available: https://hexa-x-ii.eu/https://hexa-x-ii.eu/wp-content/uploads/2023/07/Hexa-X-II_D2.1_web.pdf

[HEX223-D22] Hexa-X-II project, “Deliverable D2.2: Foundation of overall 6G system design and preliminary evaluation results”, December 2023. [Online]. Available: https ://hexa-x-ii.eu/wp-content/uploads/2024/01/Hexa-X-II_D2.2_FINAL.pdf

[HEX223-D53] Hexa-X-II project, Deliverable D5.3, “Final 6G architectural enablers and technological solutions,” April 2023, available at https://hexa-x-ii.eu/

[HEX224-D23] Hexa-X-II, “Deliverable D2.3: Interim overall 6G system”, June 2024. [Online]. Available: https://hexa-x-ii.eu/wp-content/uploads/2024/07/Hexa-X-II_D2.3-v1.1.pdf

[HEX224-D24] Hexa-X-II project, “Deliverable D2.4: End-to-end system evaluation results from the interim overall 6G system,” September 2024. [Online]. Available: https://hexa-x-ii.eu/wp-content/uploads/2024/10/Hexa-X-II_D2_4_Final_2.pdf

[HEX225-D25] Hexa-X-II project, “Deliverable D2.5: Final overall 6G system design”, April 2025 [Online]. Available: https://hexa-x-ii.eu/wp-content/uploads/2025/04/Hexa-X-II_D2.5_Final_v1.0.pdf.

[HEX224-D33] Hexa-X-II project, Deliverable D3.3, “Initial analysis of architectural enablers and framework,” April 2024, https://hexa-x-ii.eu/wp-content/uploads/2024/04/Hexa-X-II_D3.3_v1.0.pdf

[HEX224-D43] Hexa-X-II project, Deliverable D4.3, “Early results of 6G Radio Key Enablers,” April 2024, https://hexa-x-ii.eu/wp-content/uploads/2024/04/Hexa-X-II_D4_3_v1.0_final.pdf

[HEX224-D63] Hexa-X-II, “Deliverable D6.3: Initial Design of 6G Smart Network Management Framework”, June 2024. Available at https://hexa-x-ii.eu/wp-content/uploads/2024/07/Hexa-X-II_D6-3_v1.0.pdf

[HEX225-D65] Hexa-X-II, “Deliverable D6.5: Final Design on 6G Smart Network Management Framework”, February 2025. Available at https://hexa-x-ii.eu/wp-content/uploads/2025/02/Hexa-X-II_D6-5_final.pdf

[HEX225-D35] Hexa-X-II project, Deliverable D3.5, “Final architectural framework and analysis”, Feb 2025. [Online]. Available: https://hexa-x-ii.eu/wp-content/uploads/2025/03/Hexa-X-II_D3.5_v1.0.pdf

[HEX225-D45] Hexa-X-II project, Deliverable D4.5, “Final Results of 6G Radio Key Enablers,” February 2025. [Online]. Available: https://hexa-x-ii.eu/wp-content/uploads/2025/03/Hexa-X-II_D4_5_v1_edit.pdf

[HEX225-D55] Hexa-X-II project, Deliverable D5.5, “Deliverable D5.5 Final design of enabling technologies for 6G devices and infrastructure”, March 2025. [Online]. Available: https://hexa-x-ii.eu/wp-content/uploads/2025/04/Hexa-X-II_D5.5_v1.0.pdf

[HEX2-BCN+24] B. Banerjee, C. Collmann, A. Nimr, and G. Fettweis, “Flexible Transceiver Platform for Holistic Radio Design Analysis,” in Proceedings of 2024 3rd International Conference on 6G Networking (6GNet 2024), Paris, France, pp. 5, Oct 2024.

[HEX2-CNF25] C. Collmann, A. Nimr, and G. Fettweis, “Reliable Angle Estimation in True-Time-Delay Systems with Real-Time Phase Calibration,” in Proceedings of 2025 IEEE 5th International Symposium on Joint Communications & Sensing (JC&S 2025), Oulu, Finland, (p. 2), Jan 2025.

[InfluxDB] InfluxDB Cloud API Service (2.x) Available: https://docs.influxdata.com/influxdb/cloud/api/v2/

[KLM+22] H. Kokkonen, L. Lovén, N. H. Motlagh, A. Kumar, J. Partala, T. Nguyen, V. C. Pujol, P. Kostakos, T. Leppänen, A. González-Gil, E. Sola, I. Angulo, M. Liyange, M. Bennis, S. Tarkoma, S. Dustdar, S. Pirttikangas, and J. Riekki, “Autonomy and intelligence in the computing continuum: Challenges, enablers, and future directions for orchestration,“ 2022. arXiv preprint arXiv:2205.01423.

[LSL+22] K. B. Letaief, Y. Shi, J. Lu, and J. Lu, "Edge Artificial Intelligence for 6G: Vision, Enabling Technologies, and Applications," IEEE Journal on Selected Areas in Communications, vol. 40, no. 1, 2022.

[LCMNXW24] Nextworks (Italy). (2024). Closed Loop Governance LCM API. Zenodo. https://doi.org/10.5281/zenodo.14193635

[LTEF25] Lamprousi, V. (2025). Trust Evaluation Function API. Zenodo. https://doi.org/10.5281/zenodo.14720466

[MonNXW24] Nextworks (Italy). (2024). Monitoring jobs configuration. Zenodo. https://doi.org/10.5281/zenodo.14192885

[MulNXW24] Nextworks (Italy). (2024). Multi-cluster extreme-edge resource orchestration API. Zenodo. https://doi.org/10.5281/zenodo.14193659

[NIntegr24] Nicolicchia, R. (2024). Integration Fabric API (v1.0.0). Zenodo. https://doi.org/10.5281/zenodo.14222126

[VLoTAF24] Jorquera Valero, J. M., & CyberDataLab. (2024). Level of Trust Assessment Function API. Zenodo. https://doi.org/10.5281/zenodo.14191242

[wi-SUPPLY] WINGS wi.SUPPLY platform-based Warehouse Inventory Management, https://www.wings-ict-solutions.eu/wi-supply/

[ZFF+24] A. Zafeiropoulos, N. Filinis, E. Fotopoulou, and S. Papavassiliou, "AI-Assisted Synergetic Orchestration Mechanisms for Autoscaling in Computing Continuum Systems," in IEEE Communications Magazine, 2024, doi: 10.1109/MCOM.001.2200583.