Distributed Artificial Intelligence-driven open and programmable architecture for 6G networks

# D5.6 Performance evaluation and replication guidelines - Final

Document Summary Information

| | | | |
|---|---|---|---|
| **Project Identifier** | HORIZON-JU-SNS-2022.<br>Project 101095363 | **Acronym** | ADROIT6G |
| **Full Title** | Distributed Artificial Intelligence-driven open and programmable architecture for 6G networks | | |
| **Start Date** | 01.01.2023 | **Duration** | 42 months |
| **Project URL** | https://adroit6g.eu/ | | |
| **Deliverable** | D5.6: Performance evaluation and replication guidelines - Final | | |
| **Work Package** | WP5: Integration & Validation | | |
| **Contractual due date** | 30/06/2026 | **Actual submission date** | 04/07/2026 |

Co-funded by the European Union 6GSNS

'ADROIT6G 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 101095363. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.'

| **Type** | Report | **Dissemination Level** | Public |
|---|---|---|---|
| **Lead Beneficiary** | IQU | | |
| **Responsible Author** | John Vardakas | | |
| **Contributions from** | Vasos Vassiliou (CYENS), Christophoros Christophorou (CYENS), Iacovos Ioannou (CYENS), Kleanthis Neokleous (CYENS), Florian Zeiger (Siemens), Markus Sauer (Siemens), Carlos Guimarães (Siemens), Elizaveta Boriskova (Siemens), Marios Sophocleous (eBOS), Navideh Ghafouri (IQU) Swastika Roy (IQU), John Vardakas (IQU), Marius Iordache (ORO), Matteo Ravalli (NXW), Giacomo Bernini (NXW), Mounssif Krouka (OULU), Giulio Franzese (EURECOM), Ilias Chatzistefanidis (EURECOM), Thanasis Charemis (NOVA), Carla Fabiana Chiasserini, Antonella Molinaro (CNIT), Juan Jose Vegas Olmos (NVIDIA), Maido Hiiemaa (CAFA), Märten Rannu (CAFA), Tanel Järvet (CAFA), Poulcheria Zervou (ISI/ATH), Eirini Lidia Keramidi (ISI/ATH), Ioannis Kyriakopoulos (ISI/ATH), Anastasios Pandis (ISI/ATH), Dimitrios Selis (ISI/ATH), Kyriakos Stratakos (ISI/ATH), Grigoris Tzortzakis (ISI/ATH), Alaa AlZailaa (ISI/ATH). | | |
| **Peer reviewer** | Christos Verikoukis (ISI/ATH) | | |

***Revision history (including peer reviewing & quality control)***

| Version | Issue Date | % Complete | Changes | Contributor(s) |
|---|---|---|---|---|
| V0.1 | 31/01/2026 | 5% | Initial Table of Contents | John Vardakas (IQU) |
| V0.2 | 31/03/2026 | 25% | Definition of the experiments based on D5.4 | All partners |
| V0.3 | 30/04/2026 | 40% | Updates on the experiments' definition based on D5.4 | All partners |
| V0.4 | 31/05/2026 | 90% | Description of the experiments of all PoCs and replication guidelines | All partners |
| V0.5 | 10/06/2026 | 95% | Review from IQU and ISU | John Vardakas (IQU), Christos Verikoukis (ISI/ATH) |
| V0.6 | 25/06/2026 | 99% | Updates on the review | All partners |
| V1.0 | 03/07/2026 | 100% | Final Version | John Vardakas (IQU) |

© ADROIT6G, 2023

***Disclaimer***

The content of the publication herein is the sole responsibility of the publishers, and it does not necessarily represent the views expressed by the European Commission or its services.

While the information contained in the documents is believed to be accurate, the author or any other participant in the ADROIT6G consortium make no warranty of any kind with regard to this material including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

Neither the ADROIT6G Consortium nor any of its members, their officers, employees or agents shall be responsible or liable in negligence or otherwise howsoever in respect of any inaccuracy or omission herein.

Without derogating from the generality of the foregoing neither the ADROIT6G Consortium nor any of its members, their officers, employees or agents shall be liable for any direct or indirect or consequential loss or damage caused by or arising from any information advice or inaccuracy or omission herein.

***Copyright message***

© ADROIT6G Consortium, 2023-2025. This Deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both. Reproduction is authorised provided the source is acknowledged.

# Table of Contents

Table of Contents ..... 4
List of Figures ..... 5
List of Tables ..... 7
Glossary of terms and abbreviations ..... 8
Executive Summary ..... 10
1 Introduction ..... 11
1.1 Mapping ADROIT6G Outputs ..... 11
1.2 Deliverable Overview and Report Structure ..... 12
2 Performance Evaluation of PoC1 ..... 14
2.1 Introduction ..... 14
2.2 Evaluation Results ..... 14
2.2.1 Experiment 1: Infrastructure Setup Phase ..... 14
2.2.2 Experiment 2: Resources Setup Phase for Holographic Teaching session ..... 15
2.2.3 Experiment 3: Real-time Immersive Holographic Teaching Communication ..... 17
2.2.4 Experiment 3.1: Regeneration of a lost camera stream ..... 18
2.3 KPI/KVI evaluation ..... 19
2.4 Replication Guidelines ..... 24
3 Performance Evaluation of PoC2a ..... 26
3.1 Introduction ..... 26
3.2 Evaluation Results ..... 26
3.2.1 Experiment 1: Federated Predictive Analytics with CAFA ..... 26
3.2.2 Experiment 2: Integrated Teleoperation and Federated Analytics with BDIx/UE-VBS 28
3.3 KPI/KVI evaluation ..... 30
3.4 Replication Guidelines ..... 34
4 Performance Evaluation of PoC2b ..... 36
4.1 Introduction ..... 36
4.2 Evaluation Results ..... 36
4.2.1 Experiment 1: Closed-Loop Operations for Service Migration ..... 37
4.2.2 Experiment 2: UE connection enhancement ..... 41
4.2.3 Experiment 3: Automated selective SLA monitoring ..... 43
4.2.4 KPI & KVI summary ..... 46
4.3 Replication Guidelines ..... 49
5 Performance Evaluation of PoC3 ..... 50
5.1 Introduction ..... 50
5.2 Evaluation Results ..... 52
5.2.1 Experiment 1: Autonomous Transport with Dynamic Re-Routing Loading ..... 52
5.2.2 Experiment 2: Drone–AGV Collaborative working with D2D ..... 60
5.3 KPI/KVI evaluation ..... 62
5.4 Replication Guidelines ..... 65
6 Conclusion ..... 67

© ADROIT6G, 2023

# List of Figures

Figure 2-1 PoC1 Experiment #1 BDIx Agent desires onboarded in the Closed-Loop Catalogue. ..... 15

Figure 2-2 PoC1 Experiment #2 Holographic Teaching Service Blueprint onboarded in the Service Catalogue. ..... 16

Figure 2-3 PoC1 Experiment #2 Service components deployed on Kubernetes cluster. ..... 16

Figure 2-4 PoC1 Experiment #2 EMBB session. ..... 17

Figure 2-5. CPU usage in experiment 3 of PoC1. ..... 18

Figure 3-1. Federated learning convergence during PoC2a Experiment #1, showing stable NRMSE evolution across training rounds. ..... 28

Figure 3-2. PoC2a traffic-pattern results showing throughput and jitter for robot patrol, server-to-robot downlink, and 4K cooperation uplink flows. ..... 29

Figure 4-1: PoC2b Performance Evaluation (left: NTN emulation; top-right: moving trains and location of SCI-Points; bottom-right: scale-size SCI-Point connected to the setup). ..... 36

Figure 4-2. PoC application monitoring interface. ..... 37

Figure 4-3 PoC2b Experiment #1 Interlocking Service VSB onboarded in the Service Catalogue. ..... 38

Figure 4-4 PoC2b Experiment #1 Closed-Loop descriptor onboarded in the Closed-Loop Catalogue. ..... 38

Figure 4-5 PoC2b Experiment #1 Test application. ..... 39

Figure 4-6: PoC2b Experiment #1 Closed-Loop functions deployed through the Resource Orchestrator. ..... 39

Figure 4-7: E2E communication delay across different scenarios. ..... 41

Figure 4-8: Packet loss across different scenarios. ..... 41

Figure 4-9. Experiment #2 topology visualization: start of the experiment with no inter-UE connection (left) and end of the experiment with a fully connected UE swarm (right). ..... 42

Figure 4-10. Experiment #2 PoC application traffic: UE topology converged at 14:37:15. ..... 43

Figure 4-11. Experiment #3 visualization tool. ..... 44

Figure 4-12. Experiment #3 training phase monitoring. ..... 45

Figure 4-13. Experiment #3 inference phase monitoring. ..... 46

Figure 5-1. CAFA Robot application dashboard, showing the experiment control panel and a robot view with control buttons and live command/video latency. ..... 51

Figure 5-2 Running Robot app containers on three hosts: the Control Station host (top) runs the CS container (cafa_robot_cs:cyclone_gpu_2) and a Zenoh bridge; each relay host (middle, bottom) runs a relay-switch (VRo) container and a Zenoh bridge. ..... 51

Figure 5-3. PoC3 Experiment #1 Collaborative Construction Robots Service Blueprint onboarded in the Service Catalogue. ..... 53

© ADROIT6G, 2023 

Figure 5-4. PoC3 Experiment #1 Closed-Loop descriptor onboarded in the Closed-Loop Catalogue. .....54

Figure 5-5. PoC3 Experiment #1 Application components and Closed-Loop functions deployed through the Resource Orchestrator. .....54

Figure 5-6. PoC3 Experiment #1 AI models deployed in the Kubernetes cluster. .....55

Figure 5-7. PoC3 Experiment #1 Robot application scenario. .....55

Figure 5-8. PoC3 Experiment #1 Grafana dashboard monitoring the traffic and showing the current congestion score. .....57

Figure 5-9. PoC3 Experiment #1 Closed-Loop functions logs showing network slice re-configuration request. .....57

Figure 5-10. CAFA Robot application inter-node transport — FastDDS/WireGuard compared with Zenoh/CycloneDDS on an identical high-load scenario (median and 95th-percentile latency). .....59

Figure 5-11. MJPEG application payload (per-second bins) and its relationship with video latency during a high-quality video run. .....59

Figure 5-12. Application payload bitrate by video setting (per-second bins): mean, 95th percentile, and maximum. .....60

Figure 5-13. CAFA Robot application dashboard during the Experiment #2 (D2D) run, with two robots active and all Zenoh bridges configured as routers (see the experiment notes field). .....61

Figure 5-14. All-bridges-router (D2D) configuration: per-robot video latency and command RTT over a 10-minute run. .....62

# List of Tables

Table 1-1. Adherence to ADROIT6G's GA deliverable & tasks descriptions ..... 11
Table 1-2. Links to other project's outputs/ work ..... 12
Table 2-1 PoC1 Experiment #2 Performance of Service Orchestrator. ..... 17
Table 2-2. Results of experiment 3 of PoC1. ..... 18
Table 2-3. Results of the PoC1 Experiment 3. ..... 19
Table 2-4. Service KPIs for PoC1. ..... 20
Table 2-5. Network KPIs for PoC1. ..... 21
Table 2-6. Service KVIs for PoC1. ..... 22
Table 2-7. Network KVIs for PoC1. ..... 22
Table 3-1. Service KPIs for PoC2a ..... 30
Table 3-2. Network KPIs for PoC2a. ..... 32
Table 3-3. KVIs for PoC2a. ..... 32
Table 4-1 PoC2b Experiment #1 Performance of Service Orchestrator and Closed-Loop Governance. ..... 39
Table 4-2: Summary of prediction model inference performance, data scope, and latency metrics. ..... 40
Table 4-3: Service KPIs for PoC2b. ..... 46
Table 4-4: Network KPIs for PoC2b. ..... 47
Table 4-5: Service KVIs for PoC2b. ..... 47
Table 5-1 Performance comparison of the MERGE-aggregated model versus the best baseline model under the same computational budget. ..... 56
Table 5-2 Computational cost reduction of the MERGE-aggregated model compared to a baseline model of equivalent quality. ..... 56
Table 5-3. PoC3 Experiment #1 Performance of Service Orchestrator and Closed-Loop Governance. ..... 58
Table 5-4. Service KPIs for PoC3. ..... 62
Table 5-5. Network KPIs for PoC3. ..... 62
Table 5-6. KVIs for PoC3. ..... 63

© ADROIT6G, 2023

# Glossary of terms and abbreviations

| Abbreviation / Term | Description |
|---|---|
| eMBB | Enhanced Mobile Broadband |
| AI | Artificial Intelligence |
| API | Application Programming Interface |
| BDIx | Belief-Desire-Intention eXtended |
| CLG | Closed Loop Governance |
| CSP | Computing & Storage Powerhouse |
| DAI | Distributed Artificial Intelligence |
| E2E | End-to-End |
| EE | Energy Efficiency |
| FL | Federated Learning |
| IIoT | Industrial Internet of Things |
| KPI | Key Performance Indicator |
| KVI | Key Value Indicator |
| M&O | Management and Orchestration |
| ML | Machine Learning |
| mMTC | Massive Machine-Type Communications |
| MQTT | Message Queuing Telemetry Transport |
| MSE | Mean Squared Error |
| NMSE | Normalized Mean Squared Error |
| NMRSE | Normalized Root Mean Squared Error |
| OAI | OpenAirInterface |
| PoC | Proof of Concept |
| QoE | Quality of Experience |

© ADROIT6G, 2023

| | |
|---|---|
| RAN | Radio Access Network |
| REST | Representational State Transfer |
| RL | Reinforcement Learning |
| RTT | Round Trip Time |
| SLAs | Service Level Agreements |
| UE | User Equipment |
| UE-VBS | User Equipment as Virtual Base Station |

© ADROIT6G, 2023

## Executive Summary

ADROIT6G Deliverable D5.5 presents the final evaluation of the project's Proof-of-Concepts (PoCs) and associated technologies during Testing Cycle 2. The assessment, performed jointly by PoC owners and technology partners, focused on network-level Key Performance Indicators (KPIs) for the ADROIT6G innovative components and algorithms to demonstrate how ADROIT6G capabilities can support diverse application scenarios.

This deliverable provides the impact analysis and monitoring outcomes, comparing observed performance against the reference network-level targets. It also outlines the final set of replication guidelines based on Testing Cycle 2, targeting to support internal knowledge sharing within the consortium, capacity building, and external dissemination, laying the groundwork for extension towards future activities.

In more detail, deliverable D5.6 provides the evaluation results for Testing Cycle 2 of all 4 ADROIT6G PoCs: Chapter 2 for PoC1, Chapter 3 for PoC2a, Chapter 4 for PoC2b and Chapter 5 for PoC3. Multiple experiments have been defined for each PoC. For each one of these experiments, the deliverable reports the objectives, the test description and the corresponding evaluation results. At the end of each chapter, replication guidelines are provided, targeting to provide the roadmap for other testbeds to reproduce the PoC execution, measure comparable KPIs, and evaluate portability. These guidelines describe how the testbed and the corresponding environment were setup, provide the configuration and integration instructions, present the considered KPI measurement methodology, and provide portability and independence considerations, targeting to explain how other testbeds can reproduce the PoC without requiring the exact same setup as the ADROIT-6G Testing Cycles.

© ADROIT6G, 2023

# 1 Introduction

This ADROIT6G deliverable reports the final set of the WP5 activities for the performance evaluation of the ADROIT6G components and algorithms that have been reported in ADROIT6G's WP3 and WP4 deliverables. Specifically, this deliverable provides the evaluation results of the second testing cycle for all ADROIT6G Proof-of-Concepts (PoCs) against the reference network-level target Key Performance Indicators (KPIs). For each PoC, multiple experiments have been defined and evaluated. For each experiment, the objectives, the test description and the corresponding performance evaluation results are presented. Finally, for each PoC, replication guidelines are presented, targeting to enable other testbeds to reproduce the PoC execution, measure comparable KPIs, and evaluate portability without being locked to the specifics of this Testing Cycle.

## 1.1 Mapping ADROIT6G Outputs

This section presents the ADROIT6G's Grant Agreement (GA) commitments, as extracted from the formal deliverable and task description, with respect to their outputs and work to be performed, as described in Table 1.1.

Table 1-1. Adherence to ADROIT6G's GA deliverable & tasks descriptions

| ADROIT6G GA Component Title | ADROIT6G GA Component Outline | Respective Document Chapter(s) | Justification |
|---|---|---|---|
| **DELIVERABLE** | | | |
| D5.6: Performance evaluation and replication guidelines – Final<br>Report containing the evaluation process under ADROIT6G's Testing Cycle 1 | | | |
| **TASKS** | | | |
| Task 5.3 Performance evaluation and replication guidelines | The purpose of this task is three-fold: Firstly, the performance of each PoC will be evaluated by all PoC and technology partners, particularly from the network and application-level KPI and KVI perspective to demonstrate how ADROIT6G capabilities can be leveraged for different applications. Secondly, according to the results received from each PoC in every agile-based iteration, requirements and suggestions will be provided to further improve both functional and non-functional capabilities of the system. Thirdly, each PoC will be further evaluated by both the PoC and all | Chapter 2<br>Chapter 3<br>Chapter 4<br>Chapter 5 | Chapter 2 reports the evaluation activities under the framework of ADROIT6G's PoC1. Four experiments have been defined, while the objectives, the test description and the evaluation results are presented. At the end, replication guidelines are presented, designed to ensure that the PoC results can be replicated in other testbeds.<br>Chapter 3 reports the evaluation activities under the framework of ADROIT6G's PoC2a. Two experiments have been defined, while the objectives, the test description and the evaluation results are presented. At the end, |

| | | | |
|---|---|---|---|
| | technological partners to reflect the latest improvements/upgrades of the underlying infrastructure and application perspective. This task will also provide and establish systematic feedback loops to WP2-WP4, for continuous refinement. Results will be analysed both quantitively and qualitatively. Conclusions and recommendations will be drawn including guidance for further testing validations. Throughout the testing period, knowledge and findings will be documented in deliverable D5.3 together with impact analysis and monitoring (i.e., impact assessment) that will be applied to the PoCs to collect, analyse data, and present the results against the reference network-level target KPIs and KVIs. The replication guidelines will be used for internal dissemination among the consortium, capacity building and external dissemination. | | replication guidelines are presented, designed to ensure that the PoC results can be replicated in other testbeds.<br><br>Chapter 4 reports the evaluation activities under the framework of ADROIT6G's PoC2b. Three experiments have been defined, while the objectives, the test description and the evaluation results are presented. At the end, replication guidelines are presented, designed to ensure that the PoC results can be replicated in other testbeds.<br><br>Chapter 5 reports the evaluation activities under the framework of ADROIT6G's PoC3. Two experiments have been defined, while the objectives, the test description and the evaluation results are presented. At the end, replication guidelines are presented, designed to ensure that the PoC results can be replicated in other testbeds. |

## 1.2 Deliverable Overview and Report Structure

Table 1-2. Links to other project's outputs/ work

| WP # | Task # | Deliverable No related | Content |
|---|---|---|---|
| WP2 | Task 2.1 – Task 2.4 | D2.3 | D2.3 describes the final ADROIT6G architecture design and the PoCs definition, considered in D5.3. |
| WP3 | Task 3.1 – Task 3.4 | D3.2 | Technologies evaluated in D5.6 originate from WP3. |
| WP4 | Task 4.1 – Task 4.4 | D4.1 | Technologies evaluated in D5.6 originate from WP4. |
| WP5 | Task 5.1 | D5.1 | D5.1 performs and reports the integration and deployment status of testbeds, preparing the PoCs and enabling the functional block evaluation performed in D5.6. |

© ADROIT6G, 2023

| | | | |
|---|---|---|---|
| WP5 | Task 5.2 | D5.4 | D5.4 is the final version of technology validation. D5.6 provides the KPI-based performance evaluation of these technologies as well as the PoCs' replication guidelines. |
| WP5 | Task 5.3 | D5.5 | D5.6 is built on D5.5 which provided the performance evaluation results of ADROIT6G's testing cycle 1. |
| WP6 | Task 6.1 – Task 6.3 | D6.3 | D5.6 evaluation results can contribute to dissemination and communication activities and may support contributions to standardisation. |

# 2 Performance Evaluation of PoC1

## 2.1 Introduction

PoC1 evaluates a 6G-enabled holographic teaching scenario with eEMBB characteristics, where a large number of student-devices simultaneously receive individually rendered views of a shared XR learning environment. The objective of the PoC is to assess whether the ADROIT6G platform can support immersive, scalable, and responsive educational services while maintaining the quality of experience required by XR applications.

The performance evaluation of PoC1 follows up on the validation methodology defined in D5.4. The experiments were conducted in the ISI/ATH testbed and involved the Service Orchestrator and AI/ML platform provided by NXW, the slice orchestration framework from IQU, CYENS BDIx and UE-VBS components, Prometheus/Grafana monitoring tools, and the Open5GS core network. The evaluation is structured around three experiments:

- **Experiment #1**: Infrastructure Setup Phase.
- **Experiment #2**: Resource Setup Phase for the Holographic Teaching session.
- **Experiment #3**: Real-time Immersive Holographic Teaching

These experiments evaluate the ability of the ADROIT6G platform to deploy and orchestrate the required infrastructure and services, allocate and manage resources for XR workloads, and sustain real-time holographic teaching sessions under increasing user demand. The results confirm the successful execution of all experiments and the correct interaction among the involved ADROIT6G components. Furthermore, the evaluation demonstrates that the platform can effectively support immersive XR services while achieving improvements in latency, reliability, scalability, and resource efficiency. Overall, the PoC1 results provide evidence that the ADROIT6G architecture can enable advanced holographic teaching applications in future 6G networks.

## 2.2 Evaluation Results

### 2.2.1 Experiment 1: Infrastructure Setup Phase

**Objectives**: To showcase the infrastructure setup capability of automatically onboarding and configuring the UEs to be UE-VBS-capable.

**Test description:**

- Initialize the topology module and the UE-VBS-BDIx agent devices.
- Retrieve the device-specific policies and rules from the Closed-Loop Catalogue
- Confirm the connectivity between the components.

**Results:**

The deployed agent first connects to the 6G network and having established connectivity, it initiates communication with the topology module. Afterwards, it retrieves the BDIx agent desires from the Closed-Loop Catalogue, as shown in Figure 2-1: the provided desires contain information about connectivity, transmission and energy consumption.

© ADROIT6G, 2023

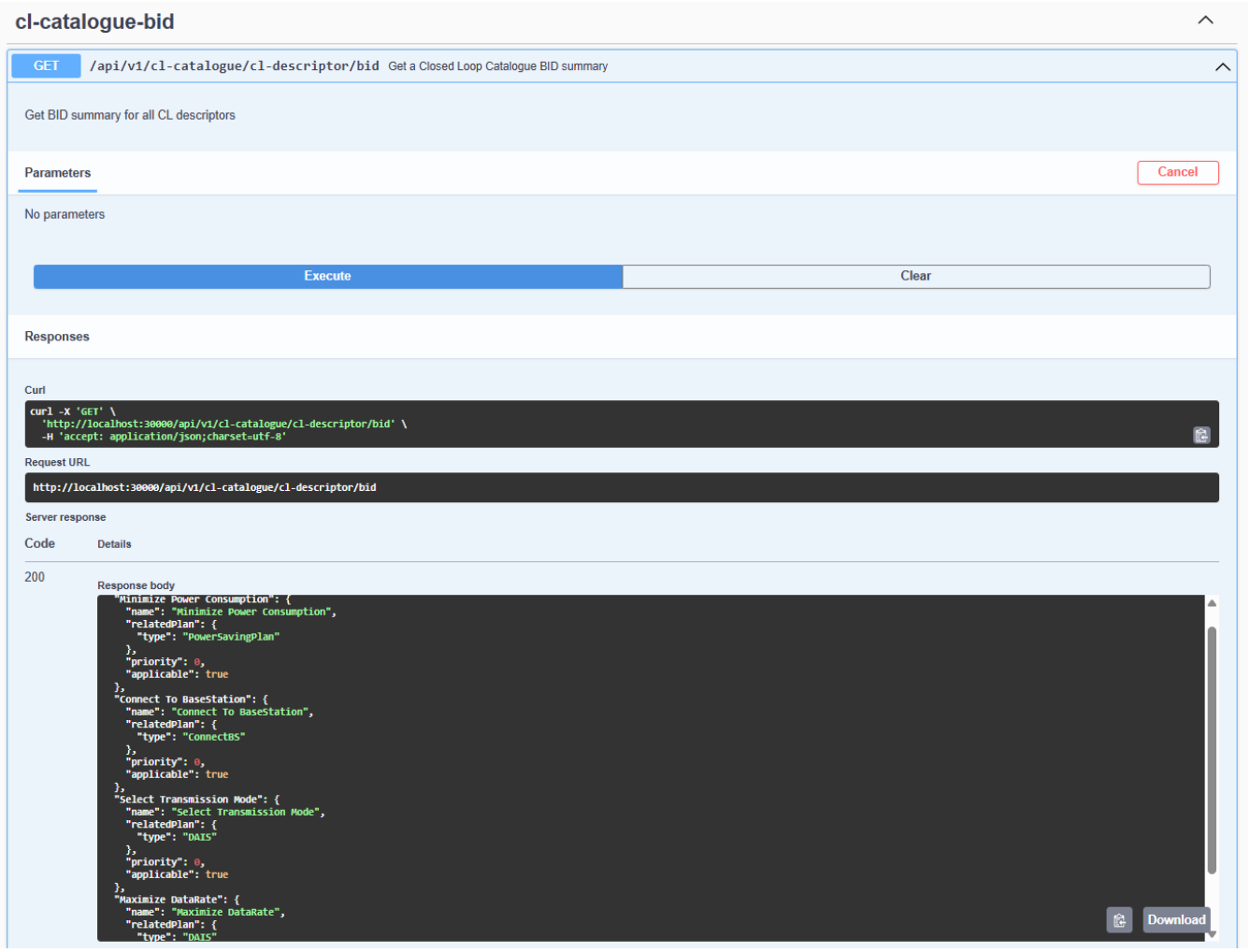


Figure 2-1 PoC1 Experiment #1 BDIx Agent desires onboarded in the Closed-Loop Catalogue.

### 2.2.2 Experiment 2: Resources Setup Phase for Holographic Teaching session

**Objectives**: To showcase the automated orchestration and intelligent resource setup process for a holographic teaching session.

**Test description:**

- Instantiate the Holographic Teaching service through the Service Orchestrator and deploy the XR application on Kubernetes.
- Start an EMBB session through the Slice Orchestrator.
- Collect the application results for KPI measurements.
- Terminate the service.
- Collect the service provisioning/decommissioning time results for KPI measurements.

**Results:** At startup, the holographic teaching service blueprint is onboarded in the Service Catalogue, as shown in Figure 2-2: it comprises three components, i.e. source, orchestrator and web, and defines the EMBB Slice Profile that interconnects the UE-VBS on students' side with the XR application on teacher side.

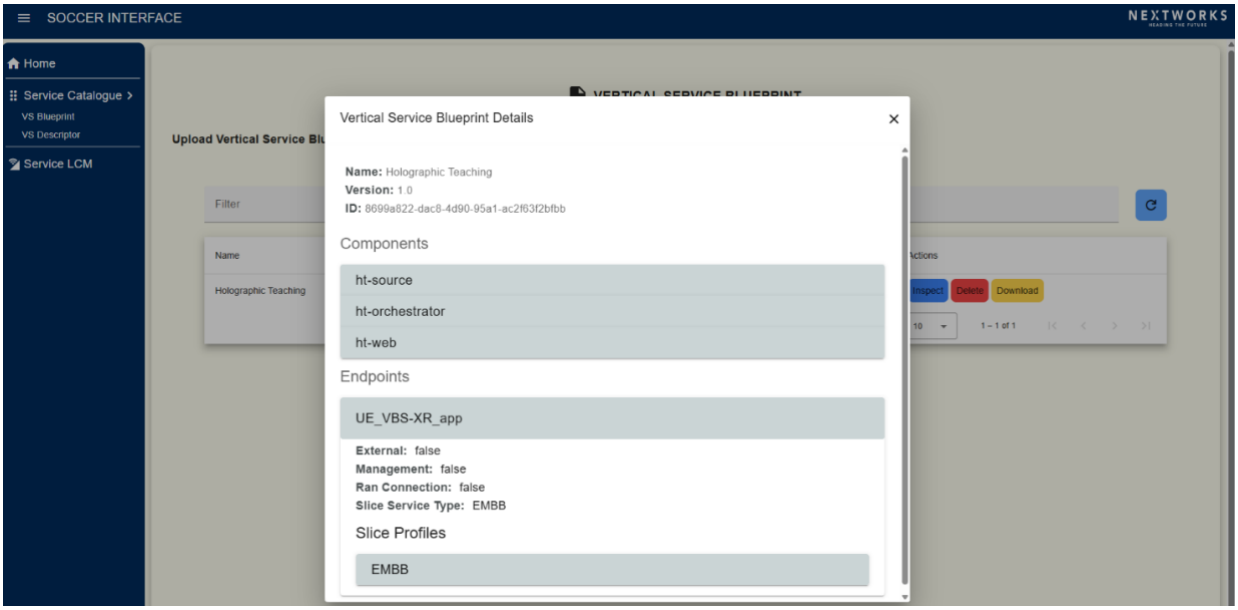


Figure 2-2 PoC1 Experiment #2 Holographic Teaching Service Blueprint onboarded in the Service Catalogue.

Once the instantiation of the service, created by providing the expected consumers as input parameter, is requested, the XR application is deployed in the Kubernetes cluster and a dedicated EMBB slice for the user traffic is created, as shown in Figure 2-3 and Figure 2-4, respectively.

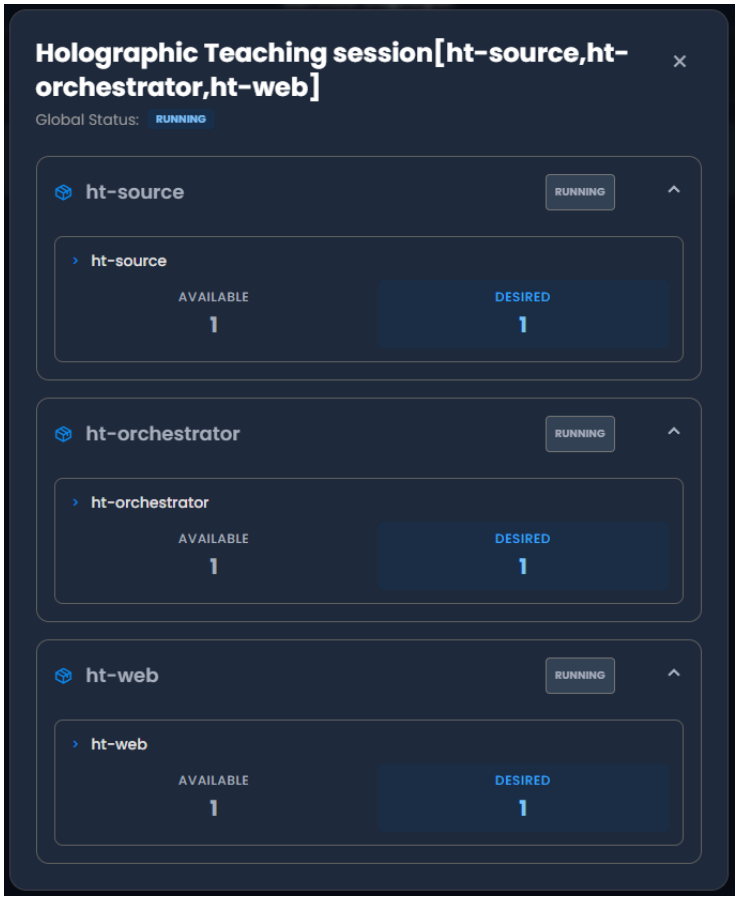


Figure 2-3 PoC1 Experiment #2 Service components deployed on Kubernetes cluster.

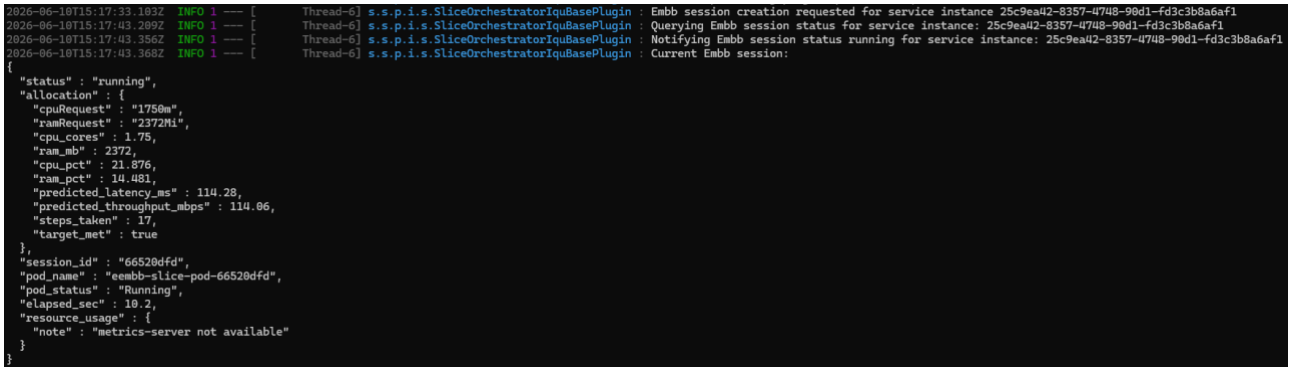


Figure 2-4 PoC1 Experiment #2 EMBB session.

Table 2-1 PoC1 Experiment #2 Performance of Service Orchestrator.

| Instance | Instantiation time (s) |
|---|---|
| Vertical service | 48.141 |
| Resource Allocation | 1.24 |
| XR application | 36.215 |
| EMBB network slice | 10.685 |

The instantiation process was executed 10 times to measure the time needed for the XR application to be deployed and the time needed for the EMBB network slice to be created. The average times from these tests are reported in Table 2-1. The achieved mean total instantiation time of 48.141 seconds is substantially lower than the 10-minute requirement, thereby fulfilling the targeted KPI for service provisioning time.

### 2.2.3 Experiment 3: Real-time Immersive Holographic Teaching Communication

**Objectives**: To validate the end-to-end real-time holographic teaching experience enabled by applications running on top of the ADROIT6G infrastructure.

**Test description:**

- The teacher initiates data streaming from the primary setup through the allocated infrastructure resources.
- Student devices are progressively connected to the active session.

System performance, resource utilization, and stream stability are monitored in real time.

**Results:** Following the successful completion of Experiment 2, the session has been created, and students are able to connect to it, and stream the holographic teacher's view.

To measure the resources required per stream a teaching session was initialized and scaled incrementally. The resource usage of the rendering pod was measured throughout, in timed intervals and averaged to produce an accurate reading. The measurements taken are shown in the following Table 2-2.

Table 2-2. Results of experiment 3 of PoC1.

| # Streams | CPU | RAM | CPU Delta | RAM Delta |
|---|---|---|---|---|
| 1 | 765m | 222Mi | - | - |
| 2 | 1282m | 336Mi | 517m | 114Mi |
| 3 | 1389m | 473Mi | 109m | 137Mi |
| 4 | 1474m | 586Mi | 85m | 113Mi |
| 5 | 1783m | 705Mi | 309m | 119Mi |
| Per stream | 357.4m | 141Mi | 255m | 120.75Mi |

By omitting the static base resource usage of the rendering application, which does not scale with users, each additional stream rendered requires approximately 0.018 TFLOPS of CPU and 148MB of RAM. Note that these measurements reflect a pure CPU workload without GPU hardware acceleration, which limits the performance but at the same time makes the application more platform agnostic.

To test the reliability of the application, the session was scaled to include 10 students/consumers and then maintained for 24 hours. The streaming didn't lose any quality, and the rendering components had stable performance without any loss of service. The deployment was also able to scale to 50 students and more, experiencing a small transient uptick in CPU load that stabilizes within 1 minute (Figure 2-5).

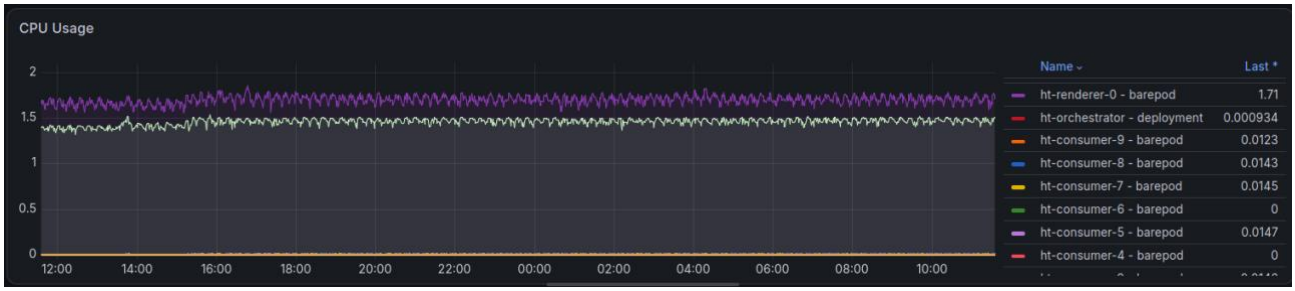


Figure 2-5. CPU usage in experiment 3 of PoC1.

Resilience and recovery capabilities were tested by forcefully scaling the source deployment down to 0 and back up to 1. Across 10 sequential test iterations, the rendering pod achieved an average recovery and readiness time of 5 seconds.

The network performance was measured by using a test pod, that connected through the 5g network and run UDP iperf3 tests from the UE to the rendering pod. For the throughput, reliability and jitter test, a simulated RAN was used to achieve the goal throughput, as the equipment of the testbed cannot support 1Gbit/s connection. This removes the impact of radio conditions on the connection but introduces a large CPU overhead to encapsulate and de-encapsulate the packages.

### 2.2.4 Experiment 3.1: Regeneration of a lost camera stream

**Objectives**: The objective of this experiment is to assess whether a lost camera stream in a calibrated multi-view capture can be regenerated from the remaining streams, providing a

resilience mechanism for networked multi-view video in which a temporal segment of one view is interrupted (e.g. packet loss or a transient link failure). Concretely, we evaluate the MVLD cross-view reconstruction method on its ability to synthesise the full image sequence of a missing camera from the two surviving, pose-calibrated views, and we quantify the fidelity of the reconstruction against the original (ground-truth) stream. The experiment also characterises the conditions under which the approach is reliable, informing how such a recovery mechanism could be integrated into a multi-view delivery pipeline

**Test description:** The test uses the PoC1 Experiment 3 recordings: a single subject captured by three synchronised Azure Kinect DK devices (1080p colour, 30 fps), with the inter-camera geometry given by the per-camera camera-to-master extrinsic transforms. Camera 0 is treated as the lost stream and is removed for a one-second window (30 consecutive frames); cameras 1 and 2 are retained as the available inputs. For each frame, MVLD — a pose-conditioned, two-pass multi-view diffusion reconstruction method — is conditioned on the two surviving views together with the known relative camera poses and generates the image at camera 0's viewpoint. The reconstruction is performed independently for every frame in the window and compared, frame by frame, to the withheld ground-truth camera-0 image. Reconstruction quality is reported using three complementary full-reference metrics — PSNR and SSIM (pixel and structural fidelity, higher is better) and LPIPS (learned perceptual distance, lower is better) — computed at a common 256×256 evaluation resolution and averaged over the 30 frames.

**Results:** MVLD successfully reconstructs the missing camera-0 stream over the full window, recovering both the room geometry and the subject at the correct viewpoint. Averaged over the 30 reconstructed frames, the method achieves the following fidelity against the ground-truth stream (Table 2-3):

Table 2-3. Results of the PoC1 Experiment 3.

| Frames | PSNR (dB) | SSIM | LPIPS |
|---|---|---|---|
| 30 | 19.77 ± 0.32 | 0.702 ± 0.007 | 0.308 ± 0.009 |

The low per-frame variance indicates a temporally stable reconstruction across the window rather than isolated good frames. Qualitatively, the synthesised views preserve the scene layout (walls, furniture, the standing subject) and the camera-0 perspective, with residual softness concentrated on fine, high-frequency detail of the subject — consistent with the generative nature of the method and its operating resolution. These results demonstrate that, for a favourably positioned viewpoint, a dropped multi-view stream can be regenerated from the surviving calibrated views with good structural and perceptual fidelity, supporting the feasibility of MVLD-based recovery as a resilience component for multi-view video transport.

## 2.3 KPI/KVI evaluation

The evaluation of ADROIT6G PoC1 “Holographic Teaching” is focused on the capability of the ADROIT6G architecture to satisfy stringent network and service Key Performance Indicators (KPIs), while simultaneously demonstrating positive societal and ecosystem-oriented Key Value Indicators (KVIs). The ADROIT6G framework follows the holistic KPIs and KVIs validation methodology for measuring performance and value in the provided services, respectively. The complete set of KPIs and KVIs, along with the descriptions of the indicators, is well-defined in the deliverables D2.3 and

© ADROIT6G, 2023

D5.4 of the project and is designed to be completely aligned and follow the IMT-2030 vision, SNS JU Key Value definitions, and the methodological frameworks of relevant standards and SDOs, being fully aligned with the relevant 3GPP and ETSI standards for 6G performance evaluation

From a performance evaluation perspective, PoC1 is an Extreme eMBB service class, due to the very high throughput, immersive visualization requirements, AI-assisted orchestration, and strict synchronization demands required for holographic interaction. The service requires simultaneous support for ultra-high bandwidth 3D streams, low end-to-end latency, low jitter, high reliability, and adaptive orchestration mechanisms capable of dynamically allocating network and computational resources according to the user demand and environmental conditions.

The KPI/KVI evaluation methodology combines quantitative monitoring with qualitative assessment mechanisms. Technical KPIs are collected through the monitoring infrastructure integrated into the ADROIT6G testbeds, leveraging real-time monitoring components, orchestration logs, network analytics, and Grafana-based visualization dashboards.

This section outlines the KPIs and KVIs associated with PoC1, which are utilized to guarantee that the methodological consistency and alignment across the ADROIT6G project, the selected KPIs and KVIs are fully harmonized with the framework and definitions established and comprehensively described in Deliverables D2.1 and D2.3. Moreover, for each KPI and KVI, D5.4 described the corresponding measurement methodology, including the monitoring tools, data sources, collection mechanisms, and assessment procedures that will be employed within the testbed environment to calculate, monitor, and validate the reported performance results.

Within this context, the KPI/KVI validation framework evaluates both the technological performance of the systems and infrastructure and the perceived value generated for the educational ecosystem. This multidimensional framework enables a holistic assessment of the PoC beyond purely technological benchmarking, capturing both quantitative system performance and qualitative human-centric outcomes. More specifically, the evaluation framework for PoC1 is structured around three complementary dimensions: Network-level KPIs, Service-level KPIs, and Value-oriented KVIs.

- Network-level KPIs assess the capability of the underlying 6G architecture to deliver the required communication and computing performance.
- Service-level KPIs evaluate the operational behavior and user-centric characteristics of the holographic teaching service itself.
- Value-oriented KVIs assess the broader impact of the solution with respect to innovation, democracy, sustainability, trustworthiness, and inclusion.

Table 2-4. Service KPIs for PoC1.

| KPI | Description | Target Value | Measured in | PoC1 Measurement |
|---|---|---|---|---|
| SR-HT-01 | End-to-End Latency | < 350 ms | Exp. 3 | 1080ms (due to heavy computational overhead) |

| | | | | |
|---|---|---|---|---|
| SR-HT-03 | Reliability | > 95% uptime | Exp. 2 and/or 3 | >99.9% |
| SR-HT-04 | System Scalability | Up to 50 devices | Exp. 3 | 50 |
| SR-HT-05 | AI Model Selection Latency | < 100 ms | Exp. 1 | 360.6 ms |
| SR-HT-06 | Orchestration Efficiency | < 10 min[1] | Exp. 2 | 48.141 sec |
| SR-HT-07 | Service Recovery Time | < 1 minute | Exp. 2 and/or 3 | 5s |
| SR-HT-08 | Computing power needed per stream rendering (aggregate processing power in CSP) | ~30 TFLOPS | Exp. 1 and/or 2 | 0.018 TFLOPs |
| SR-HT-09 | Ram needed per stream rendering | <32 GB | Exp. 1 and/or 2 | 147.8 MB |

Table 2-5. Network KPIs for PoC1.

| KPI | Description | Target Value | Measured in | PoC1 Measurement |
|---|---|---|---|---|
| NR-HT-01 | Peak Throughput | > 1 Gbps | Exp. 2 and/or 3 | 0.984 Gbps |
| NR-HT-02 | Latency | < 150 ms | Exp. 2 and/or 3 | 40.196ms |
| NR-HT-03 | Jitter | < 5 ms | Exp. 3 | 0.031 ms |
| NR-HT-04 | Reliability | > 95% | Exp. 2 and/or 3 | 99.47% |
| NR-HT-05 | Orchestration Latency | < 250 ms | Exp. 2 | Does not apply to the Open5Gs core, it is pre-configured, and it's not being managed by ETSI MANO. |
| NR-HT-06 | Resource Utilization Efficiency | > 90% utilization | Exp 1 and 2 | There are no limits being applied. |
| NR-HT-07 | Slice Adaptation Time | < 10 s. | Exp 2 | 10.685 s. |

[1] For clarity, the KPI value reported in D5.4 has been revised. The previously reported value does not refer to the Service Provisioning Time KPI. The applicable target for this KPI is less than 10 minutes, and this deliverable evaluates the corresponding experiments against this reference value.

© ADROIT6G, 2023

Table 2-6. Service KVIs for PoC1.

| KVI | Description | Values/Scaling | PoC1 Measurement |
|---|---|---|---|
| KVI-SR-HT-I1 | Safety | User-Centric perception Low / Medium / High | High |
| KVI-SR-HT-I2 | Security | User-Centric perception Low / Medium / High | High |
| KVI-SR-HT-I3 | Regulation | User-Centric perception Low / Medium / High | High |
| KVI-SR-HT-I4 | Responsibility | 0 user manipulation incidents | 0 users manipulation incidents were noticed during routine inspection of administrative access logs to rule out unauthorized changes |
| KVI-SR-HT-D1 | Privacy | User-Centric perception Low / Medium / High | High |
| KVI-SR-HT-D2 | Fairness | User-Centric perception Low / Medium / High | High |
| KVI-SR-HT-E1 | Sustainability | Mean Opinion Score (MOS) of > 4 of the users expressing positive evaluation regarding the sustainability of the devices, network, developed applications and services. | Average value of using Likert scale 4,5 among the users answered the relevant questionnaire |

Table 2-7. Network KVIs for PoC1.

| KVI | Description | Values | PoC1 Measurement |
|---|---|---|---|
| KVI-NR-HT-I1 | Energy efficiency | 10% improved energy efficiency compared to current status | Testbed simulation and emulation (in the testbed lab) managed to achieve and energy efficiency >10% |

| | | | |
|---|---|---|---|
| KVI-NR-HT-D1 | Digital Inclusion | 50 students with connected devices and 99.9% communication service reliability | Reliability indicator reached a range between 99.47% to 99.99% |
| KVI-NR-HT-D2 | Trustworthiness | 100% consistent coverage of the network | >99.9% of coverage was achieved |
| KVI-NR-HT-E1 | Open collaboration | 50 connected devices able to interact | Users per service: 50 tested by the testbed |

The KPIs evaluation demonstrates that PoC1 has successfully validated the capabilities of the ADROIT6G architecture to support immersive holographic teaching services requiring extremely high bandwidth, ultra-low latency, AI-assisted orchestration, and dynamic resource management. More specifically, the experimental results showed that the platform managed to address or exceed almost all the predefined performance objectives. From a service perspective, the holographic teaching application achieved more than 99.9% service uptime, substantially exceeding the target reliability of 95%, while successfully scaling to 50 concurrent users, fully satisfying the scalability objective. The measured service recovery time of only 5 seconds, compared with the target of less than one minute, demonstrates the effectiveness of the orchestration framework in rapidly restoring services after failures. Furthermore, the computational efficiency of the rendering platform proved particularly encouraging, requiring only approximately 0.018 TFLOPS and 148 MB RAM per rendered stream, significantly below the allocated resource budget, indicating that immersive holographic applications can be deployed efficiently even without GPU acceleration. Although the AI model selection latency of 360.6 ms exceeded the desired target of 100 ms, this measurement primarily reflects the current implementation of model retrieval and initialization rather than a limitation of the underlying communication infrastructure, providing a clear direction for future optimisation of AI lifecycle management.

The network-level KPIs further confirmed the maturity and robustness of the ADROIT6G platform for supporting next-generation immersive applications. The achieved end-to-end network latency of 40.2 ms remained comfortably below the required 150 ms threshold, while the measured jitter of only 0.031 ms indicates exceptionally stable packet delivery, an essential requirement for maintaining synchronized holographic interactions and high-quality user experiences. Network reliability reached 99.47%, again significantly surpassing the 95% target, while throughput of 0.984 Gbps approached the ambitious 1 Gbps objective despite limitations imposed by the available experimental infrastructure. The measured network slice provisioning time of 10.685 sec was also very close to the target value, demonstrating the orchestration framework's ability to dynamically instantiate and configure network resources for immersive services. Taken together, these measurements confirm that the integration of cloud-native orchestration, Kubernetes-based deployment, AI-assisted resource management, and slice orchestration provides a highly responsive and scalable infrastructure capable of supporting extreme eMBB applications with predictable performance and efficient resource utilisation.

© ADROIT6G, 2023

Beyond the technical performance, the KVI evaluation demonstrates that PoC1 generates significant educational, societal, and ecosystem value through the deployment of immersive 6G services. The platform successfully supports the key value dimensions of safety, security, privacy, fairness, responsibility, sustainability, digital inclusion, trustworthiness, and open collaboration, illustrating the holistic design philosophy of ADROIT6G. The ability to reliably support 50 simultaneously connected devices, together with high communication reliability and stable service continuity, directly contributes to digital inclusion by enabling equitable access to immersive education. In addition, the qualitative KVIs based on the UX (including safety, security, privacy, fairness, responsibility, sustainability, digital inclusion, trustworthiness) were reinforced through the highly reliable network performance, deterministic communication behaviour, and resilient service orchestration, while open collaboration is facilitated through the interoperability of cloud-native components, network slicing, and AI-driven resource management. Sustainability is further supported by the remarkably low computational resource requirements per holographic stream, indicating that large-scale deployment of immersive educational services can be achieved with relatively modest computing resources and improved infrastructure efficiency.

Collectively, the KPI and KVI results demonstrate that PoC1 extends beyond validating a high-performance 6G networking platform by illustrating how advanced AI-enabled orchestration, edge-cloud infrastructures, and immersive communication technologies can generate measurable societal benefits. The experiments confirm that ADROIT6G successfully combines technical excellence with human-centric value creation, providing a robust foundation for future 6G-enabled educational ecosystems that are scalable, resilient, sustainable, and capable of delivering high-quality immersive learning experiences.

## 2.4 Replication Guidelines

The replication of the PoC1 Holographic Teaching experiments requires deployment of the full ADROIT6G testbed within a Kubernetes cluster. The core components include the Service Orchestrator, Closed Loop Governance engine, ML-Platform, Slice Orchestrator, UE-VBS BDIx Agents, and the Topology Module. In addition, a Prometheus–Grafana monitoring stack should be deployed for infrastructure and application observability. The Service Orchestrator must be onboarded with the holographic teaching service blueprint, and all container images for the holographic teaching application components must be preloaded and made available on all Kubernetes worker nodes to ensure deterministic deployment performance.

First, for experiment initialization, the UE-VBS BDIx agents are deployed and connected through the 5G core network to the Topology Module. The agents retrieve their corresponding models, desires, and beliefs from the ML-Platform and use this information to establish and configure the system topology. Once the topology is initialized, the Closed Loop Governance process is activated, enabling continuous optimization of resource allocation and communication behavior based on system objectives. During this phase, the AI Model Selection Latency is measured, defined as the time required for UE agents to retrieve and instantiate the appropriate models from the ML-Platform and integrate them into the operational control loop.

Second, for Experiment #2, the Service Orchestrator is accessed via the SOCCER web interface to instantiate a new holographic teaching session. This triggers automated deployment of the XR-based teaching application and the corresponding service components within the Kubernetes cluster. In parallel, the Slice Orchestrator establishes the required network slice to support the session traffic. During this experiment, key orchestration KPIs are collected, including Service

© ADROIT6G, 2023

Orchestration Efficiency and Service Recovery Time, measured across multiple instantiation and failure-recovery cycles.

Third, for Experiment #3, student devices connect to the initialized teaching session and begin streaming the holographic rendered view generated by the XR application. The system is evaluated under increasing numbers of concurrent users to assess scalability and stability. In this phase, both application-level and infrastructure-level metrics are collected, including CPU and RAM consumption per stream, end-to-end streaming latency, system throughput, and service reliability under load.

Finally, KPI and KVI measurements are collected using an integrated Prometheus–Grafana monitoring stack deployed within the same Kubernetes cluster, providing visibility into compute, memory, and service-level metrics. Network performance evaluation is conducted using UE-VBS BDIx agents connected through the 5G core, which execute iperf3-based measurements between UE endpoints and the rendering pods. These measurements capture throughput, latency, jitter, and reliability under controlled emulated network conditions.

The minimum set of evidence required for replication includes Kubernetes deployment logs for all PoC components, Slice Orchestrator session creation logs, Prometheus and Grafana dashboards, application streaming performance metrics, and iperf3 network test results. These artefacts collectively enable full reproduction of the PoC1 evaluation and KPI assessment.

© ADROIT6G, 2023

# 3 Performance Evaluation of PoC2a

## 3.1 Introduction

PoC2a evaluates a terrestrial 6G Industrial IoT (IIoT) factory scenario with extreme mMTC characteristics, where many industrial devices, sensors, actuators, mobile robots, and edge nodes generate both periodic and event-driven traffic. The purpose of the PoC is to assess whether the ADROIT6G platform can support safe and responsive robot teleoperation, predictive analytics for production management, and adaptive end-to-end orchestration under dynamic load conditions.

The performance evaluation of PoC2a builds directly on the validation workflow defined in D5.4. The experiments were executed using the CAFA Robot application patterns, based on the Gazebo robot simulator, together with the EURE cloud-native testbed, OAI-based 5G/6G emulation, Prometheus/Grafana monitoring, OULU federated learning and quantization mechanisms, and CYENS BDIx/UE-VBS components. The evaluation focuses on two experiments:

- **Experiment #1**: Federated Predictive Analytics with CAFA, evaluating the application and AI/ML level.
- **Experiment #2**: Integrated Teleoperation and Federated Analytics with BDIx/UE-VBS, evaluating the complete end-to-end integration level.

The results confirm that the PoC2a experiments executed successfully and that the ADROIT6G components interacted as expected. The platform was able to deploy and sustain the analytics workflow, support distributed federated learning with reduced communication overhead, enable low-latency teleoperation, and preserve service stability under increasing device density and dynamic load. Overall, the PoC2a evaluation demonstrates that the ADROIT6G architecture can provide measurable improvements in latency, reliability, scalability, and energy-aware operation compared with the baseline configuration.

## 3.2 Evaluation Results

### 3.2.1 Experiment 1: Federated Predictive Analytics with CAFA

**Objectives:** From the application point of view, the objective of Experiment #1 was to validate the use of AI-driven predictive analytics over the ADROIT6G platform. The experiment focused on the deployment of the CAFA analytics service together with federated learning components, where distributed clients process local IIoT and robot telemetry and send quantized model updates to the aggregator. The purpose was to demonstrate that predictive maintenance and process optimization insights can be generated without requiring the centralized collection of all raw data. From the testbed and component point of view, the objective was to demonstrate that the orchestration stack can deploy and sustain the analytics workflow and that quantized federated learning reduces the communication and energy cost of the AI/ML workflow.

**Test description:** The ADROIT6G components were configured and operated as follows:

- Instantiated the CAFA analytics service and the federated learning components through the Service Orchestrator and allocated the required compute and networking resources through the Resource Orchestrator in the EURE Kubernetes-based testbed.
- Ingested telemetry generated by the robot and IIoT traffic patterns into the CAFA backend; the federated learning clients trained locally and exchanged quantized updates with the aggregator, after which the global model was updated and redistributed.

© ADROIT6G, 2023

- Collected service-level and infrastructure-level metrics through the Prometheus/Grafana monitoring stack, and allowed the Closed-Loop Governance component to trigger corrective actions (e.g. scaling or redeployment) when thresholds are approached.
- Collected the application and AI/ML results for KPI measurements.

The Robot application traffic was generated from the CAFA Robot application dashboard. The operator selected a scenario in which the robots move and stream telemetry, optionally entered an experiment name and notes, and started the run; the scenario executed automatically and the generated traffic fed the analytics workflow.

**Results:** The experiment was executed successfully. The Service Orchestrator instantiated the CAFA analytics service and the federated learning components, while the Resource Orchestrator allocated the required compute and networking resources in the EURE Kubernetes-based testbed. The CAFA backend ingested telemetry generated by the robot and IIoT traffic patterns, while the federated learning clients trained locally and exchanged quantized updates with the aggregator, while the training process was realized with stable Normalized Root Mean Squared Error (NRMSE), as depicted in Figure 3-1. The global model was updated and redistributed to the clients.

The monitoring stack, based on Prometheus and Grafana, successfully collected service-level and infrastructure-level metrics during the experiment. The Closed-Loop Governance component monitored the analytics service and was able to trigger corrective actions when thresholds were approached, such as scaling or redeployment actions in response to resource pressure or ingestion lag. No blocking deployment failures were observed during the final experiment runs.

The use of quantized federated learning updates reduced the communication overhead of the AI/ML workflow. Compared with the non-quantized baseline exchange of model updates, the quantized update mechanism reduced the amount of transmitted learning-related data by approximately 30%, while preserving the usefulness of the predictive analytics workflow. This reduction contributed directly to improved energy efficiency and lower network load, which is particularly important in the extreme mMTC context of PoC2a.

The main observations from Experiment #1 are summarized as follows:

- The CAFA analytics service and patterns was successfully deployed and maintained through the ADROIT6G orchestration stack.
- The federated learning workflow operated correctly with distributed clients and a central aggregator.
- Quantized model updates reduced the communication overhead of the federated learning process.
- The monitoring stack provided the required observability for service performance, resource usage, and closed-loop actions.
- The experiment validated the Production Manager user story by providing near-real-time analytics and predictive maintenance information.

Overall, Experiment #1 validated the application and AI/ML layer of PoC2a. The results confirm that the ADROIT6G platform can support predictive analytics for IIoT environments while reducing communication and energy costs through federated learning quantization.

© ADROIT6G, 2023

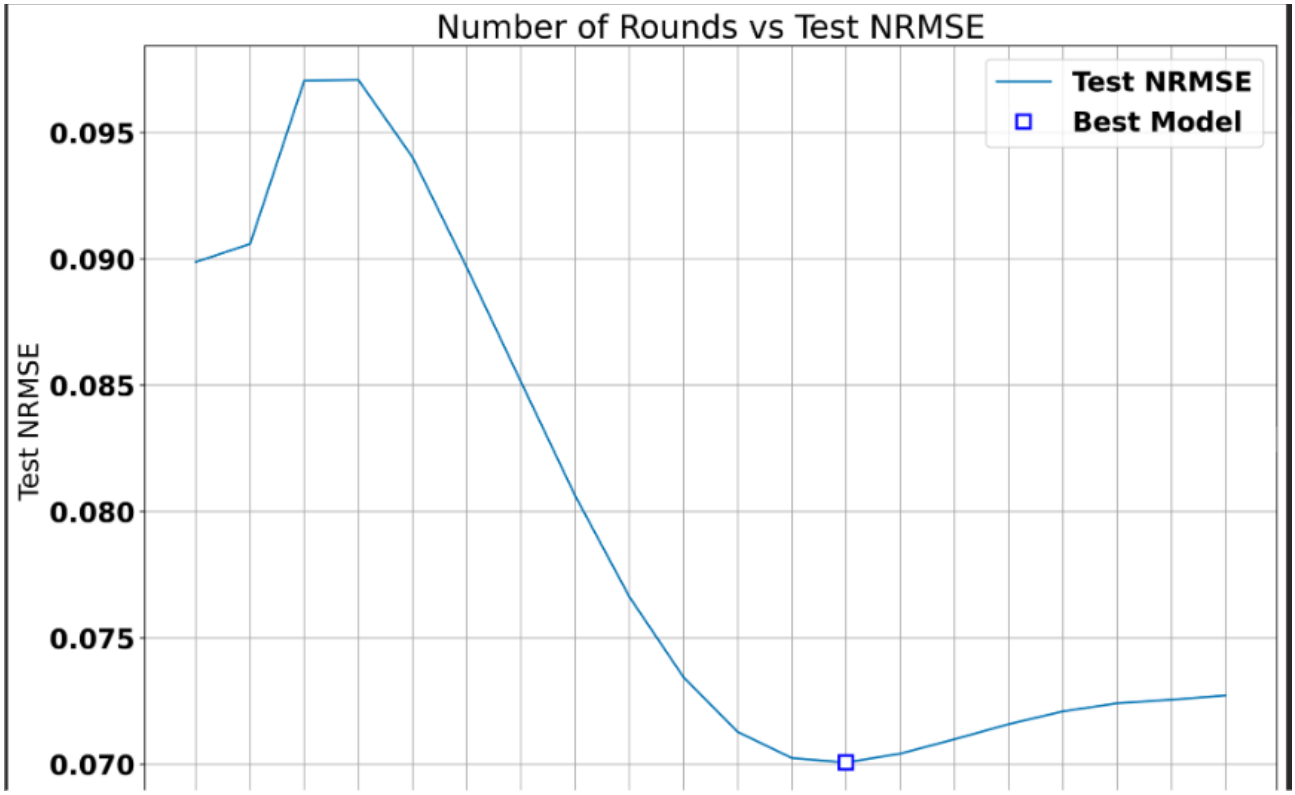


Figure 3-1. Federated learning convergence during PoC2a Experiment #1, showing stable NRMSE evolution across training rounds.

### 3.2.2 Experiment 2: Integrated Teleoperation and Federated Analytics with BDIx/UE-VBS

**Objectives:** The objective of Experiment #2 was to validate the complete end-to-end integration of robot teleoperation, federated predictive analytics, distributed resource discovery, and adaptive infrastructure control. This experiment combined the CAFA robot teleoperation and analytics workflow with the ADROIT6G orchestration stack, BDIx agents, UE-VBS control, slice monitoring, and closed-loop control mechanisms.

**Test description:** The integrated workflow was configured and operated as follows (EURE/CYENS/OULU):

- Deployed the CAFA robot teleoperation and analytics workflow together with the ADROIT6G orchestration stack, BDIx agents, UE-VBS control, and the slice-monitoring and lifecycle-management xApp.
- Operated multiple IIoT devices and mobile robot traffic sources concurrently, generating command-and-control traffic together with telemetry and sensor streams.
- Allowed the BDIx agents to discover topology and resource information, evaluate communication options, and interact with the UE-VBS control component; allow the slice-monitoring xApp and Closed-Loop Governance to observe slice KPIs and support closed-loop control through policy updates.
- Executed the planned scale-up scenario, including traffic spikes, dynamic workload changes, and emulated failures, and collect the results for KPI measurements.

The Robot application teleoperation and traffic were driven from the CAFA Robot application dashboard, using scripted scenarios in which the robots move along assigned routes while streaming video and telemetry, so that the generated traffic was repeatable across runs.

**Results:** The experiment was executed successfully. Multiple IIoT devices and mobile robot traffic sources operated concurrently, generating command-and-control traffic as well as telemetry and sensor streams. The BDIx agents discovered topology and resource information, evaluated available communication options, and interacted with the UE-VBS control component to support resilient D2D and device-to-control-station communication paths. In parallel, the slice-monitoring and lifecycle-management xApp observed slice-level KPIs and supported closed-loop control through policy updates.

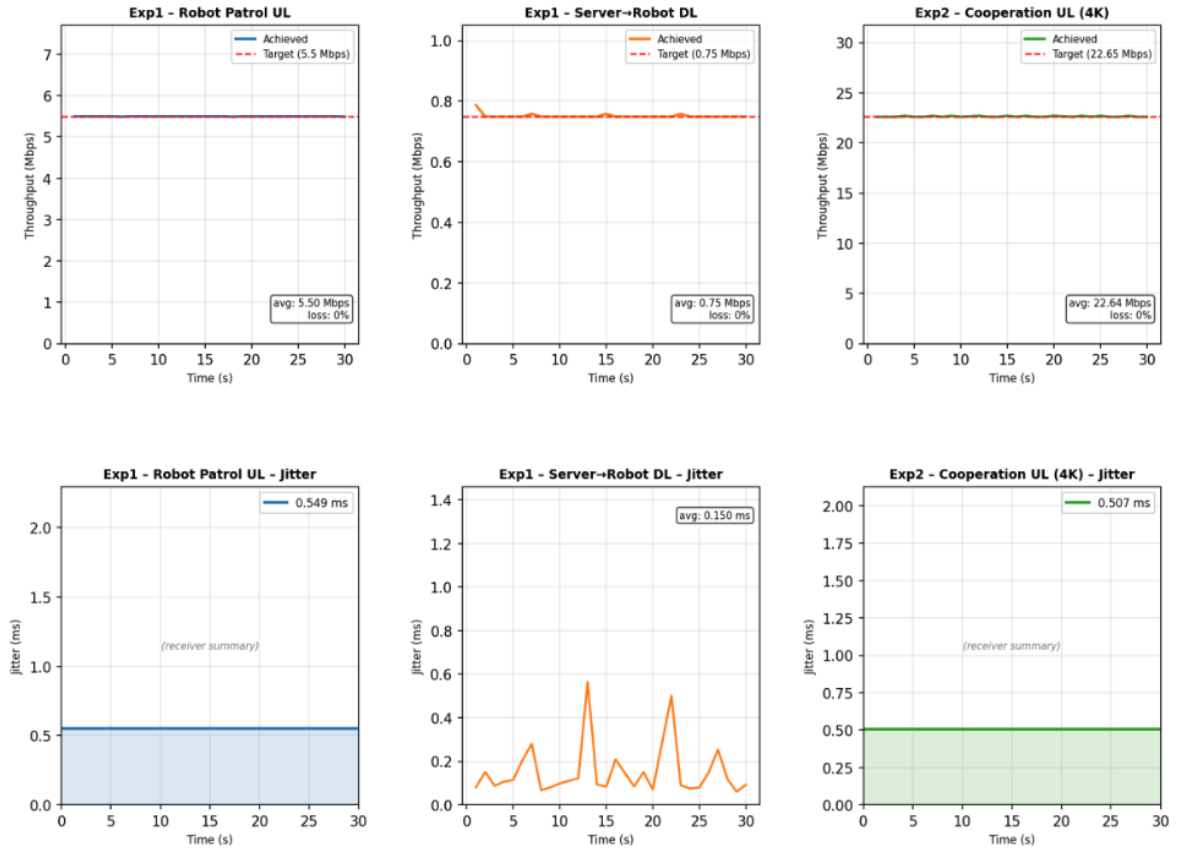


Figure 3-2. PoC2a traffic-pattern results showing throughput and jitter for robot patrol, server-to-robot downlink, and 4K cooperation uplink flows.

The experiment confirmed that the integrated PoC2a workflow can maintain stable teleoperation and analytics under dynamic load. During the final experiment runs, the system supported the planned scale-up scenario, including traffic spikes, dynamic workload changes, and emulated failures. The ADROIT6G-enabled configuration maintained the target communication delay below 10 ms for the teleoperation control path, with measured average end-to-end latency of approximately 8ms. Jitter remained below the target threshold of 1 ms, with an observed average jitter of approximately 0.6 ms (Figure 3-2). The synchronized timing accuracy remained within the target bound, with observed synchronization error below 500 microseconds.

© ADROIT6G, 2023

Reliability was also maintained during repeated runs. The successful session completion ratio and service continuity observations resulted in an achieved service reliability of approximately 99.99%, exceeding the PoC2a target of 99.95%. The platform also showed bounded performance degradation as the number of active traffic sources and emulated devices increased, validating the scalability target of slow degradation under higher device density.

The BDIx/UE-VBS functionality contributed to resilience by enabling distributed communication decisions and alternative relay paths. When the system experienced increased load or emulated degradation, the distributed control logic supported communication path adaptation without interrupting the teleoperation workflow. The xApp and Closed-Loop Governance mechanisms complemented this behaviour by monitoring slice and service KPIs and triggering orchestration-level corrective actions when required.

The main observations from Experiment #2 are summarized as follows:

- The end-to-end PoC2a workflow was successfully deployed and executed.
- Robot teleoperation traffic remained stable under concurrent analytics traffic.
- BDIx agents and UE-VBS components supported distributed resource discovery and resilient communication control.
- The slice-monitoring xApp and Closed-Loop Governance mechanisms enabled runtime observability and adaptation.
- The system maintained latency, jitter, reliability, throughput, and scalability targets during the final successful experiment runs.
- The experiment validated the Robot Operator, Production Manager, and Integrator user stories jointly.

Overall, Experiment #2 validated the complete end-to-end PoC2a integration. The results demonstrate that the ADROIT6G platform can support low-latency and reliable robot teleoperation while concurrently running predictive analytics and adaptive orchestration in an extreme mMTC IIoT environment.

## 3.3 KPI/KVI evaluation

The following tables summarize the PoC2a KPI and KVI evaluation results. The targets are taken from the PoC2a KPI/KVI definitions established in D5.4, while the achievement descriptions reflect the final successful experiment runs.

Table 3-1. Service KPIs for PoC2a.

| KPI | Description | Target Value | Measured / Observed Result | Achievement description |
|---|---|---|---|---|
| SR-Poc2a-01 | Latency | Communication delay < 10 ms | Average: 8 ms | Achieved. The ADROIT6G-enabled teleoperation workflow |

| | | | | |
|---|---|---|---|---|
| | | | | maintained end-to-end communication delay below the 10 ms target during the final Experiment #2 runs. |
| SR-Poc2a-02 | Reliability | SLA 99.95% | 99.99% | Achieved. Repeated runs completed successfully without loss of service continuity or unsafe interruption of the teleoperation workflow. |
| SR-Poc2a-03 | Scalability | Slow degradation for higher numbers, e.g., >10 devices/m² | Bounded degradation under increased emulated density; final scenario executed with multiple traffic sources, robots, UEs, and edge nodes | Achieved. Performance degradation remained gradual and controlled as the number of active traffic sources and emulated devices increased. |
| SR-Poc2a-04 | Energy Efficiency | >30% reduction compared with legacy/baseline operation | Approximately 30% estimated reduction | Achieved. Quantized federated learning updates reduced communication overhead and contributed to energy-aware operation compared with the baseline non-quantized workflow. |

Table 3-2. Network KPIs for PoC2a.

| KPI | Description | Target Value | Measured / Observed Result | Achievement description |
|---|---|---|---|---|
| NR-Poc2a-01 | Jitter | < 1 ms | Average: 0.6 ms | Achieved. Jitter remained below the target threshold during teleoperation and concurrent analytics traffic. |
| NR-Poc2a-02 | Synchronized timing accuracy | < 500 microseconds | Approximately 400 microseconds | Achieved. Timestamp comparison across involved components showed synchronization accuracy within the required bound. |
| NR-Poc2a-03 | Upload and download speeds | ≥10 Mbps per device | Upload: 11 Mbps per device; Download: 20 Mbps per device | Achieved. Device-level throughput remained above the minimum target under nominal and loaded conditions. |

Table 3-3. KVIs for PoC2a.

| KVI | Description | Target / Value | Achievement description |
|---|---|---|---|
| KVI-NR-Poc2a-I1 | Safety | Safe operation maintained; accidents avoided | Achieved. The robot teleoperation scenarios completed without unsafe behaviour, loss of control, or failed control sequence during the final experiment runs. |
| KVI-NR-Poc2a-I2 | Security | Ability to detect anomalies, SLA violations, or abnormal behaviour | Achieved at validation level. The monitoring and orchestration framework detected service and resource conditions relevant to SLA degradation and enabled corrective closed-loop actions. |

© ADROIT6G, 2023

| | | | |
|---|---|---|---|
| KVI-NR-Poc2a-I3 | Regulation | Number of standards, interfaces, or standard-aligned components involved | Achieved. The PoC used standard-aligned interfaces and components, including O-RAN-aligned xApp interaction, Kubernetes-based cloud-native deployment, Prometheus/Grafana monitoring, ROS2/Gazebo robot integration, and OAI-based 5G/6G emulation. |
| KVI-NR-Poc2a-D3 | Digital Inclusion | Lower latency than current procedures; communication delay <10 ms | Achieved. The measured teleoperation latency remained below 10 ms, supporting more responsive access to remote industrial operation. |
| KVI-NR-Poc2a-D4 | Trustworthiness | SLA 99.95% and latency comparison versus baseline | Achieved. The ADROIT6G-enabled configuration achieved 99.99% reliability and maintained latency below the target threshold, improving trust in stable remote operation. |
| KVI-NR-Poc2a-E1 | Sustainability | Energy-aware operation linked to SR-Poc2a-04 | Achieved. Quantized federated learning reduced transmitted model-update data and contributed to an estimated energy-efficiency improvement above the 30% target. |
| KVI-NR-Poc2a-E2 | Business Value | Scalability improvement under higher density linked to SR-Poc2a-03 | Achieved. The system maintained bounded degradation as device density and traffic load increased, supporting scalable |

© ADROIT6G, 2023

| | | | |
|---|---|---|---|
| | | | industrial deployment. |
| KVI-NR-Poc2a-E3 | Economic Growth | Productivity increase or operational cost decrease | Achieved qualitatively. The combination of predictive analytics, remote teleoperation, and closed-loop orchestration can reduce manual intervention, support earlier maintenance decisions, and improve operational efficiency. |

## 3.4 Replication Guidelines

The PoC2a experiments can be replicated by following the same workflow and testbed configuration defined during the validation phase.

First, the EURE Kubernetes-based testbed should be prepared. The required components include the Service Orchestrator, Resource Orchestrator, Closed-Loop Governance engine, monitoring stack, OAI core and emulated RAN components, federated learning clients and aggregator, and the CAFA analytics and robot teleoperation application or traffic patterns. The recommended namespaces are adroit-system for orchestration, closed-loop governance, BDIx and federated learning components; trirematics for OAI network functions; and monitoring for Prometheus, Alertmanager, and Grafana.

Second, for Experiment #1, the CAFA analytics service and federated learning workflow should be deployed through the Service Orchestrator. The Resource Orchestrator should allocate the required compute and networking resources, while the monitoring stack collects service and infrastructure metrics. IIoT and robot telemetry traffic should be generated using both periodic and event-driven traffic patterns. Federated learning clients should train locally and transmit quantized model updates to the aggregator. The experiment should record model-update overhead, analytics service latency, resource usage, and dashboard availability.

Third, for Experiment #2, the BDIx agents, UE-VBS control components, and slice-monitoring xApp should be enabled in addition to the components used in Experiment #1. The experiment should generate concurrent teleoperation and analytics traffic while progressively increasing the number of active devices, traffic sources, and emulated edge nodes. Dynamic load spikes and failure conditions should be introduced to verify closed-loop adaptation. The monitoring stack should collect latency, jitter, reliability, synchronization accuracy, throughput, and service continuity measurements.

Fourth, results should be compared against the PoC2a KPI and KVI targets. In particular, the replication should verify that communication latency remains below 10 ms, jitter remains below 1 ms, synchronized timing accuracy remains below 500 microseconds, throughput remains above 10 Mbps per device, reliability remains above 99.95%, and energy-efficiency improvement exceeds 30% compared with the baseline. The KVI assessment should also verify that safe teleoperation is

© ADROIT6G, 2023

maintained, monitoring and orchestration provide trustworthy operation, scalability degradation remains bounded, and the analytics workflow supports production-management value.

The minimum evidence to be collected during replication includes CAFA application traffic logs, robot command latency logs, Prometheus and Grafana monitoring traces, orchestration logs, closed-loop governance decisions, federated learning update logs, quantization overhead measurements, and any xApp or BDIx/UE-VBS control logs. These artefacts are sufficient to reproduce the KPI/KVI assessment and compare the replicated execution against the results reported in this deliverable.

© ADROIT6G, 2023

# 4 Performance Evaluation of PoC2b

## 4.1 Introduction

PoC2b is focused on Railway Operations, namely an exemplary *Interlocking Service*. Therefore, the PoC application consists of emulated track-element devices (i.e., the RaSTA SCI-Points) and the Interlocking Server (i.e., RaSTA server). The PoC application leverages NTN primary and/or backup connection, as already showcased in the Testing Cycle 1. An overview is depicted in Figure 4-1.

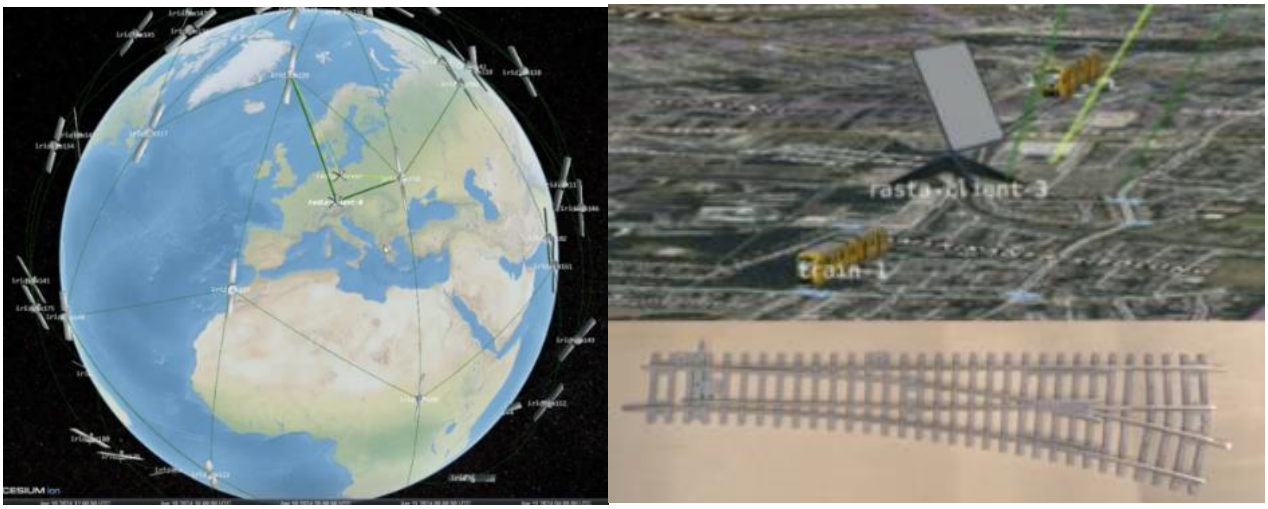


Figure 4-1: PoC2b Performance Evaluation (left: NTN emulation; top-right: moving trains and location of SCI-Points; bottom-right: scale-size SCI-Point connected to the setup).

Testing Cycle 2 advances the performance evaluation carried out in Testing Cycle 1 and showcases large parts of the interlocking service enhanced with the ADROIT6G stack and NTN connectivity (comprising both communication and computing capabilities). In specific, closed loops automate railway service migration (Experiment #1), BDIx agents handle UE connection improvement in the background (Experiment #2) and SLA monitoring, crucial for safety critical processes, is performed by FL (Experiment #3).

Figure 4-2 shows a typical trace from a single running PoC application pair (i.e., SCI-Point and Interlocking Service) when connected via an NTN bent-pipe channel. As a result, the monitoring application interface allows live tracking of PoC application KPIs.

## 4.2 Evaluation Results

This section iterates through the 3 experiments performed in Testing Cycle #2, created as a complement of the experiments already performed in the previous testing cycle: (i) Closed-Loop Automation for Service Migration, (ii) UE connection enhancement with BDIx agents; and (iii) Automated selective SLA monitoring. Impressions of the experiment are outlined, the KPIs measured are listed along with overall replication guidelines for the experiments carried out.

© ADROIT6G, 2023

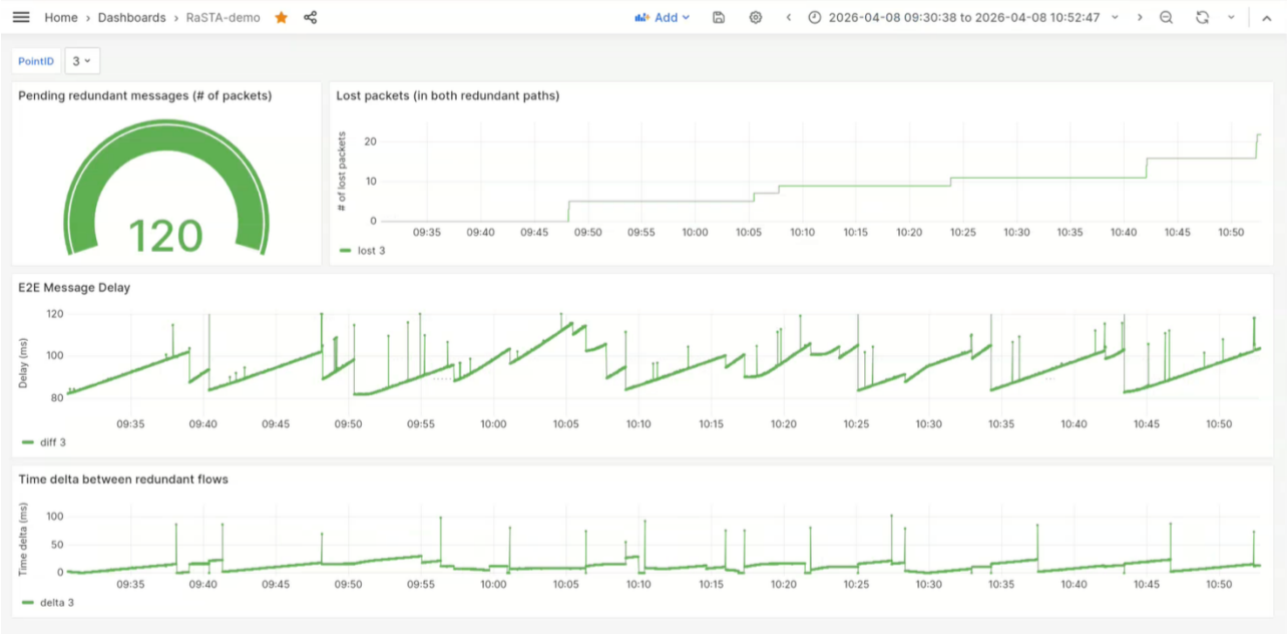


Figure 4-2. PoC application monitoring interface.

### 4.2.1 Experiment 1: Closed-Loop Operations for Service Migration

**Objectives**: showcase automation of service migration by the Closed-loop governance, including the closed-loop operations automation.

**Test description:**

- Initialize the service with via Service Descriptor triggered by the Service Orchestrator.
- Deployment of the virtualized application components, and instantiation of the closed-loop functions via Closed-Loop Governance.
- Monitor application performance throughout experiment, including the service migration whenever a closer satellite is detected within the coverage area.
- Stop the emulation.
- Collect the results for KPI measurements.

The closed-loop functions are expected to monitor for *(i)* degradation of the service KPIs and *(ii)* incoming satellites within the connectivity range of SCI Points. Based on these two parameters, the closed-loop decides when and to which satellite the virtualized application components need to be migrated, as well as when to trigger such service migration.

**Results:** At startup, the service blueprint for Railway Interlocking Service and the closed-loop descriptor are onboarded in their respective catalogues, as shown in Figure 4-3 and Figure 4-4. The service blueprint contains information about the RaSTA application to be started and the endpoints for the communication between the Interlocking Server and the track-element devices. The descriptor of the closed-loop, and of the related functions, highlights the aim of reconfiguring the service upon a satellites handover while assuring the KPIs.

© ADROIT6G, 2023

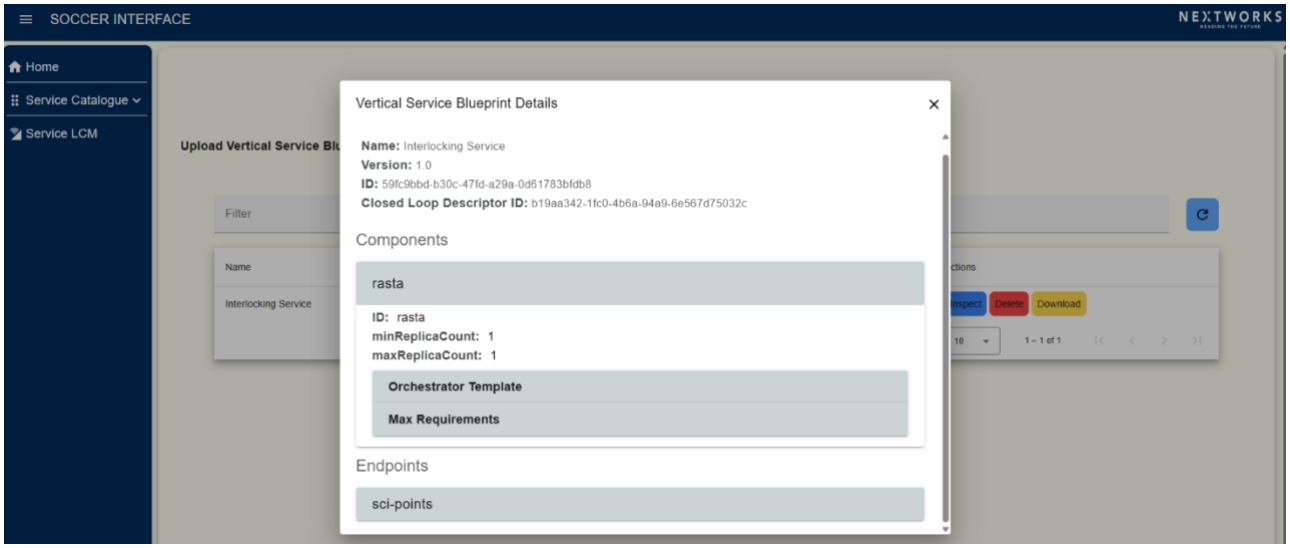


Figure 4-3 PoC2b Experiment #1 Interlocking Service VSB onboarded in the Service Catalogue.

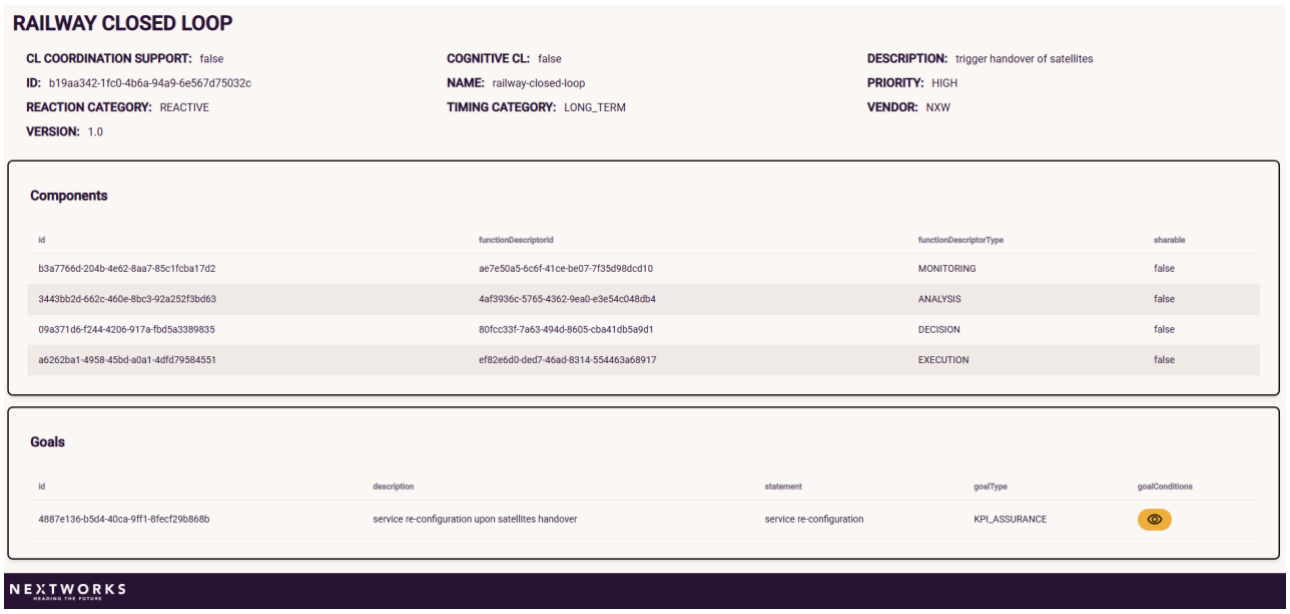


Figure 4-4 PoC2b Experiment #1 Closed-Loop descriptor onboarded in the Closed-Loop Catalogue.

Once the instantiation of the service, created by providing the MQTT topics for the closed/loop functions and the IP of the NTN Emulation System as input parameters, is requested, the test application is started (Figure 4-5) and the closed-loop functions are deployed in the Kubernetes cluster (Figure 4-6). At this point, the closed-loop starts monitoring the satellite position and whenever it is no more visible, it triggers a handover to migrate the service while keeping the SLA.

The instantiation process was executed 10 times to measure the time needed for the test application to be started and the time needed for closed-loop to be allocated. The average times from these tests are reported in Table 4-1. The total instantiation time for a closed-loop instance accounts for the complete setup of every closed-loop function, which specifically includes:

- Deploying the function within the Kubernetes cluster.
- Subscribing to Message Queuing Telemetry Transport (MQTT).
- Sending a notification to the Service Orchestrator.

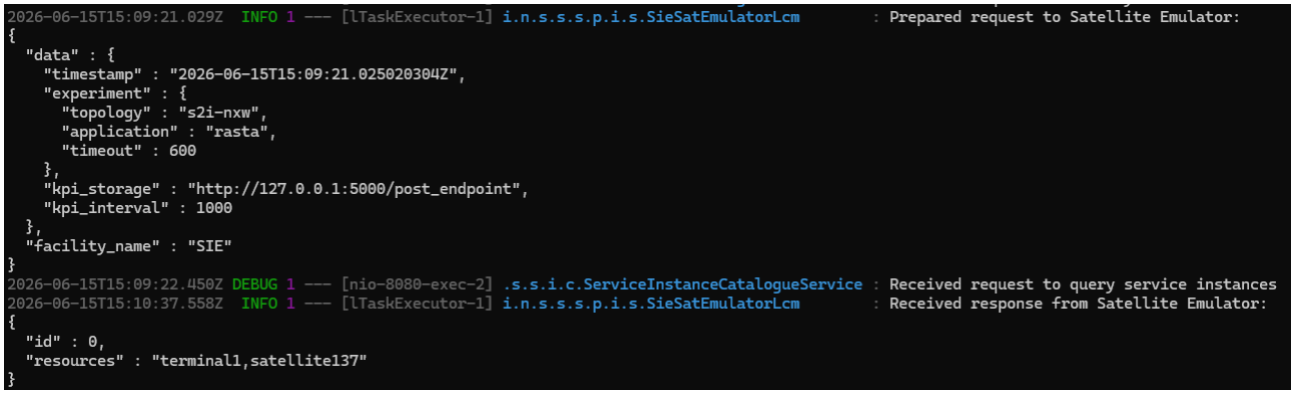


Figure 4-5 PoC2b Experiment #1 Test application.

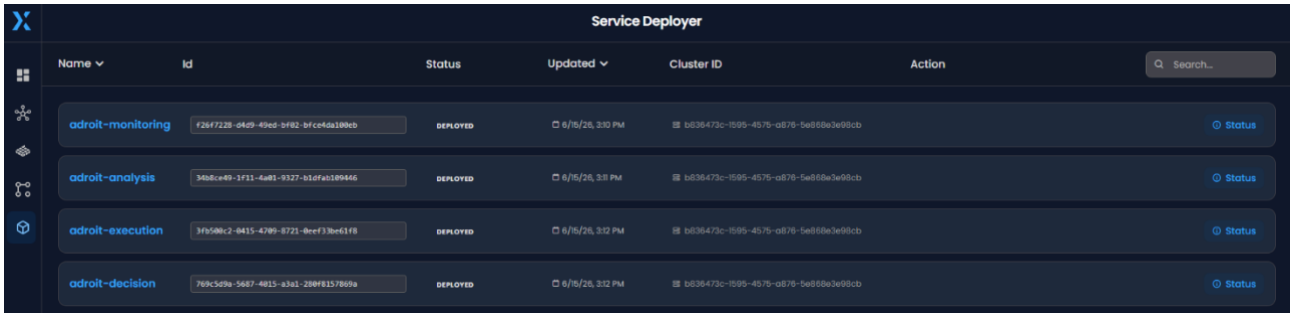


Figure 4-6: PoC2b Experiment #1 Closed-Loop functions deployed through the Resource Orchestrator.

The achieved mean total instantiation time of 211.412 seconds is substantially lower than the 10-minute requirement, thereby fulfilling the targeted KPI for service provisioning time.

Table 4-1 PoC2b Experiment #1 Performance of Service Orchestrator and Closed-Loop Governance.

| Instance | Instantiation time (s) |
|---|---|
| Vertical service | 211.412 |
| Test application | 29.143 |
| Closed-loop instance | 178.185 |
| Closed-loop monitoring function | 44.84 |
| Closed-loop analysis function | 43.931 |
| Closed-loop decision function | 45.131 |
| Closed-loop execution function | 44.23 |

Table 4-2 summarizes the prediction model inference results to be part of the analysis function. The model achieved an average top-1 accuracy of 48.70% and an average top-3 accuracy of 84.52%, indicating strong candidate ranking performance despite moderate first-choice accuracy. Performance was highly consistent across runs, with a standard deviation of only 0.35% for top-1

accuracy and 0.22% for top-3 accuracy. Top-1 accuracy ranged from 48.27% to 49.10%, while top-3 accuracy ranged from 84.12% to 84.81%. The evaluation was conducted on a combined total of 59,998 samples. In terms of efficiency, the model required an average initialization and retrieval time of 50.94 ms, an average full inference batch time of 702.28 ms, and an average per-sample latency of 0.0585 ms.

Table 4-2: Summary of prediction model inference performance, data scope, and latency metrics.

| Category | Metric | Value |
|---|---|---|
| **Model accuracy** | Average top-1 accuracy | 48.70% |
| | Average top-3 accuracy | 84.52% |
| | Standard deviation of top-1 accuracy | 0.35% |
| | Standard deviation of top-3 accuracy | 0.22% |
| | Minimum and maximum client top-1 accuracy | 48.27% - 49.10 % |
| | Minimum and maximum client top-3 accuracy | 84.12% - 84.81% |
| **Data scope** | Total combined evaluated samples | 59,998 |
| **Inference latency** | Average model retrieval (init) time | 50.94 ms |
| | Average full inference batch time | 702.28 ms |
| | Average per-sample latency | 0.0585 ms |

Given that a best-case evaluation would require higher levels of predictive accuracy than those achieved by the model, we decided not to use the model in the subsequent results. Instead, we implemented a deterministic computation of the available satellites and their locations, which is then used within the closed-loop framework.

The results in Figure 4-7 show that the E2E message delay varies significantly over time and differs markedly across the three evaluated cases. The *Bent-Pipe* scenario (green line) represents the case where the system relies exclusively on satellite bent-pipe communication, providing a baseline for comparison without hosting the application onboard the satellite segment. In contrast, the *Interlocking @ Satellite (with ISL, no migration)* scenario (yellow line) leverages communication between the terminal and a virtual application running on the same LEO satellite, but without relocating the application as satellite positions change over time. This case exhibits the highest latency levels and the largest fluctuations, frequently approaching and often exceeding the indicated latency threshold, with several pronounced peaks throughout the observation window. Finally, the *Interlocking @ Satellite (with migration via closed-loop)* scenario (blue line) extends the previous satellite-hosted application case by continuously migrating the virtual application to the closest available satellite via the ADROIT6G closed-loops. As a result, it consistently achieves the lowest latency and smallest variability, showing that the closed-loop functions can effectively

identify the nearest satellite and dynamically relocate the interlocking service accordingly. Overall, these results indicate that this latter case can leverage the ADROIT6G capabilities to provide the most reliable delay performance.

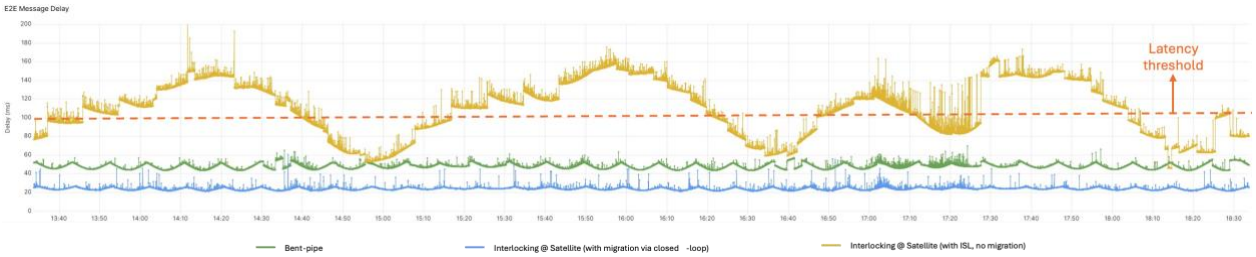


Figure 4-7: E2E communication delay across different scenarios.

Figure 4-8 shows the cumulative packet loss observed over time for the three evaluated cases, highlighting clear differences in reliability. The *Bent-Pipe* scenario (green line), which relies solely on satellite bent-pipe communication, experiences a limited number of packet losses and remains relatively stable throughout the observation period. The *Interlocking @ Satellite (with ISL, no migration)* scenario (yellow line), where the terminal communicates with a virtual application running on the same LEO satellite without migration, exhibits by far the highest packet loss, with the cumulative number of lost packets increasing steadily and reaching the largest final value among all cases. In contrast, the *Interlocking @ Satellite (with migration via closed-loop)* scenario (blue line), which extends the previous case by continuously migrating the virtual application to the closest available satellite, achieves the lowest packet loss and shows the most stable behaviour over time. Overall, these results confirm that the closed-loop migration mechanism not only improves latency performance but also significantly enhances communication reliability, allowing the ADROIT6G-enabled solution to maintain a more robust end-to-end service.

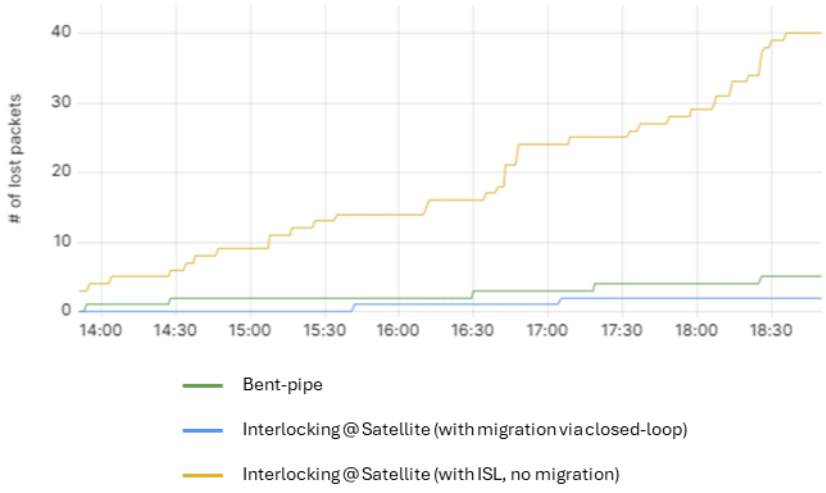


Figure 4-8: Packet loss across different scenarios.

### 4.2.2 Experiment 2: UE connection enhancement

**Objectives**: To demonstrate 6G-enabled UE behaviour in the context of dynamically changing network environment.

**Test description:**

- Initialize the constellation with BDIx agents and the PoC application

- Start testbed adaptor monitoring DAI decisions and executing link changes
- Start the emulation, BDIx agents, DAI algorithm and PoC application
- Observe link creation and deletion until stabilization of a full UE connection tree
- Monitor application performance throughout the link change
- Stop the emulation, BDIx agents and DAI algorithm
- Collect the results for KPI measurements

UE transformation decisions are expected to correlate with LOS for ground devices, and a converged connectivity tree is expected to lower end-to-end delay for the PoC application due to fast D2D links

**Results:** BDIx agents collaboratively achieve minimal service delay by forming a tree among themselves, successfully using Emulation information (e.g., latitude, longitude, LOS).

Figure 4-9 shows the initial topology (left), where no high-speed connectivity is initialized between the UEs. In under 2 minutes, the UEs form high speed D2D links, resulting in a tree structure connecting all UEs with a PoC client deployed to the PoC server in space (Figure 4-9, right).

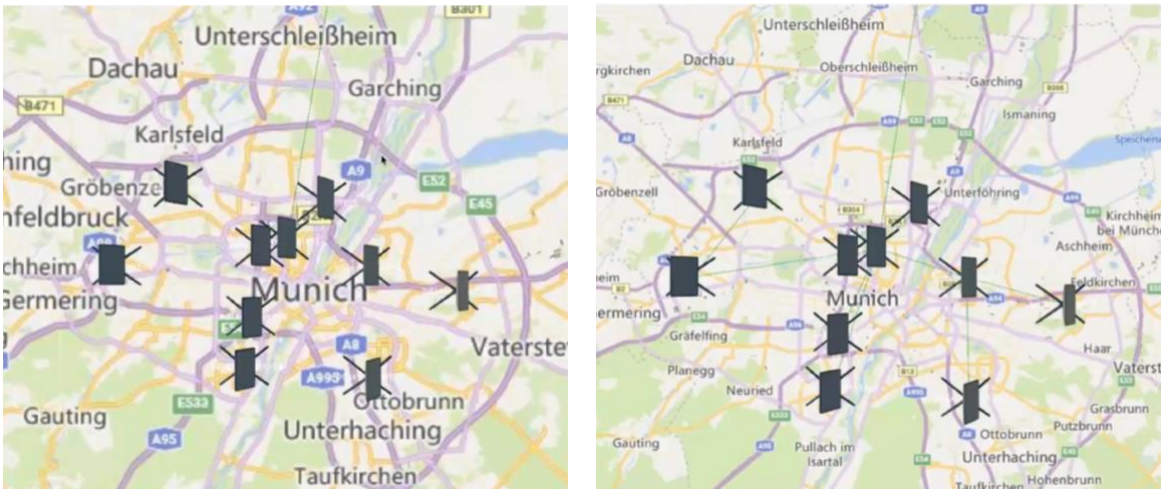


Figure 4-9. Experiment #2 topology visualization: start of the experiment with no inter-UE connection (left) and end of the experiment with a fully connected UE swarm (right).

Logically, prior to D2D link formation connection via management links is used for PoC traffic. As the tree forms, e2e delay of the PoC process drops for all PoC clients, as visible on Figure 4-10 at time 14:37:20. *Availability* is also improved, as no sharp spikes are visible in time delta between redundant flows after the UE tree is formed. As no messages are dropped, even in longer runs, *reliability* in this test also persists.

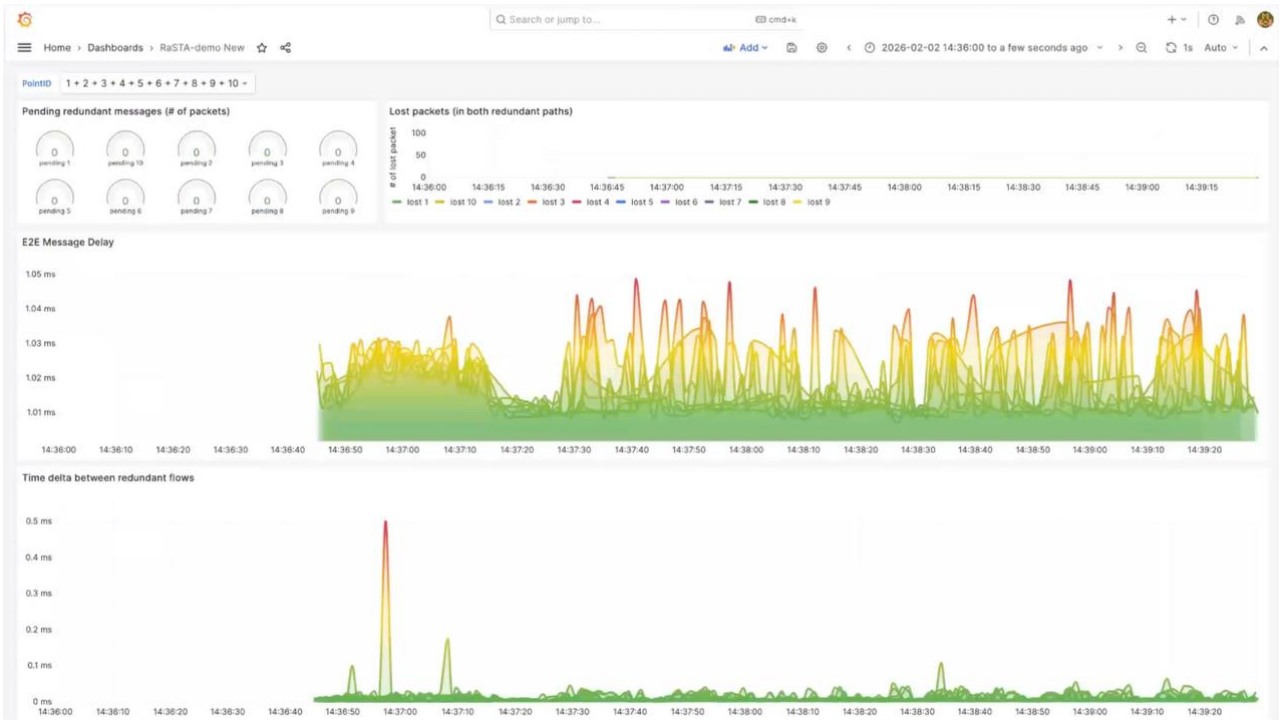


Figure 4-10. Experiment #2 PoC application traffic: UE topology converged at 14:37:15.

As the UEs were setup with emulated location, *scalability* is also achieved as the emulation and UE-internal software can handle 10 and more UEs running concurrently sharing the same emulated location.

### 4.2.3 Experiment 3: Automated selective SLA monitoring

**Objectives:** To present AI-operations working alongside safety-critical railway mechanisms, without interference or resource starvation.

**Test description:**

- Initialize the constellation with PoC application, FL clients and server for Predictive Monitoring and SLA Offering Solutions
- Start the orchestration script, consisting of emulation start, training initialization and inference call upon training completion
- Observe training and inference impacting PoC application metrics
- View inference results, confirming SLA violations during training phase
- Stop the orchestration script, consisting of emulation termination, FL clients and server training termination and DB cleanup

Training and inference are expected not to degrade the PoC application performance, moreover, SLA monitoring is projected to assist further process automation by confirming or denying violations.

**Results:** Federated Learning agents perform training and execute inference with no PoC service degradation. Moreover, agents monitor end-to-end delay SLA by reporting inference results back to Testbed Orchestration.

The topology in this experiment is similar to the one in the previous experiment, yet all terminals have a direct uplink to the satellite (see Figure 4-11). Terrestrial clients run not only Federated Learning clients, but also PoC clients, as well as the satellite executing both FL server and PoC server.

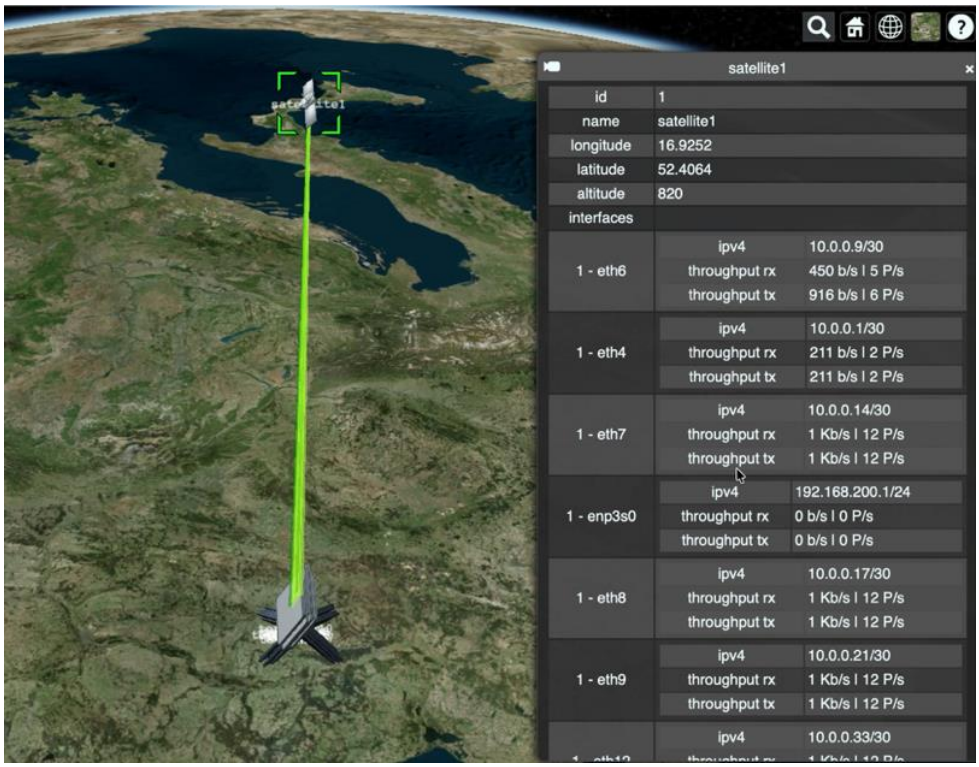


Figure 4-11. Experiment #3 visualization tool.

Figure 4-11 and Figure 4-12 allow to visually compare PoC performance prior to training and during training. Clearly, no PoC service degradation can be observed, with similar end-to-end delay and no lost messages hence *reliability* is achieved. As spikes are visible in time delta metric, *availability* of the service cannot be claimed to persist 99.999% short term, however, the training is completed rather quick and long-term high availability can still be achieved. Moreover, averaged over multiple training experiments availability shows to be very high, over 99%.

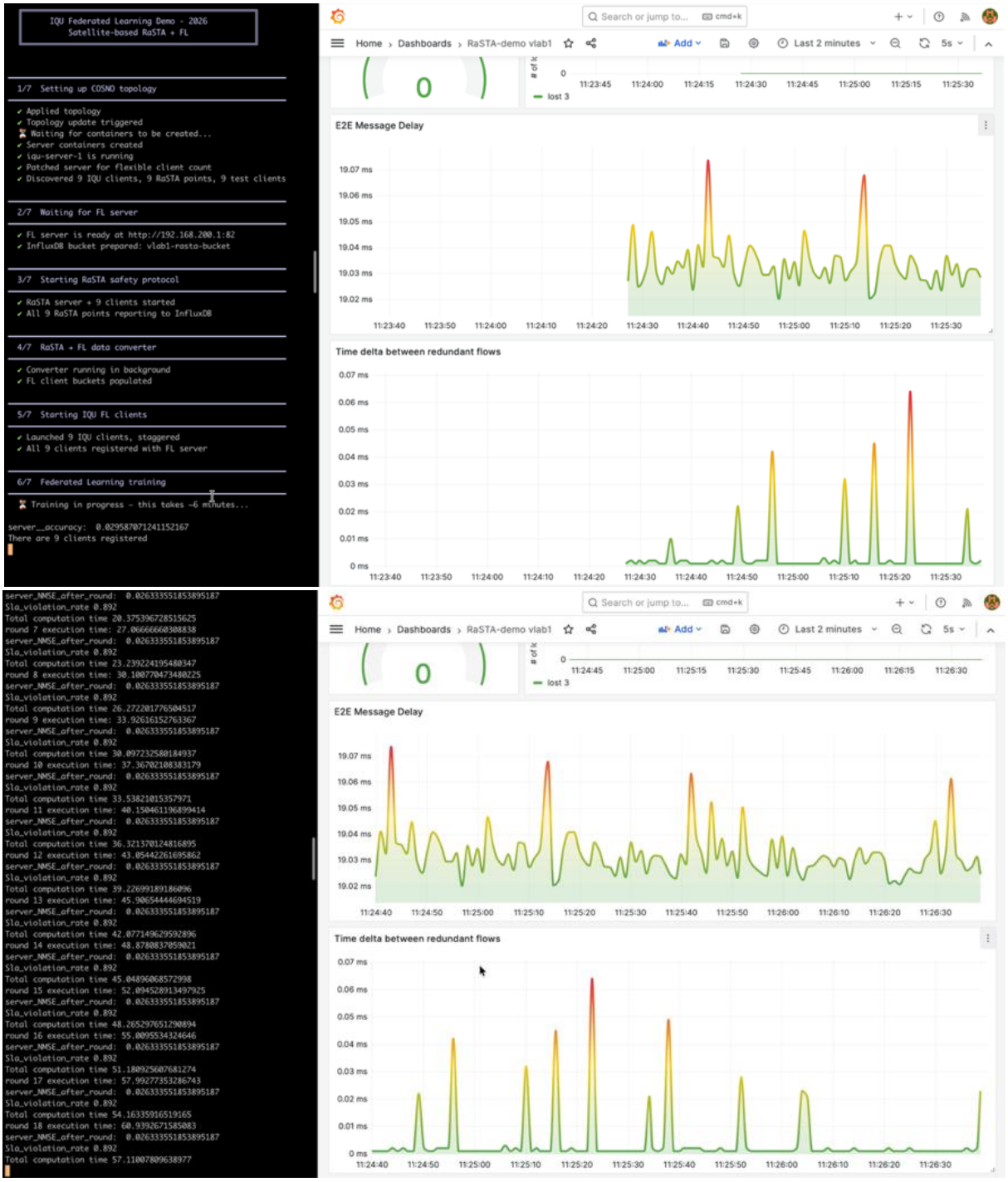


Figure 4-12. Experiment #3 training phase monitoring.

In this test as the training stage is completed, the model is stored and then dispatched to all clients for the inference phase. Figure 4-13 shows all clients reporting inference results on the latest PoC client metrics. “SAFE” in this context means that no SLA violations have been observed during the training phase. *Scalability* is achieved here as more than 10 clients are established both for training and inference.

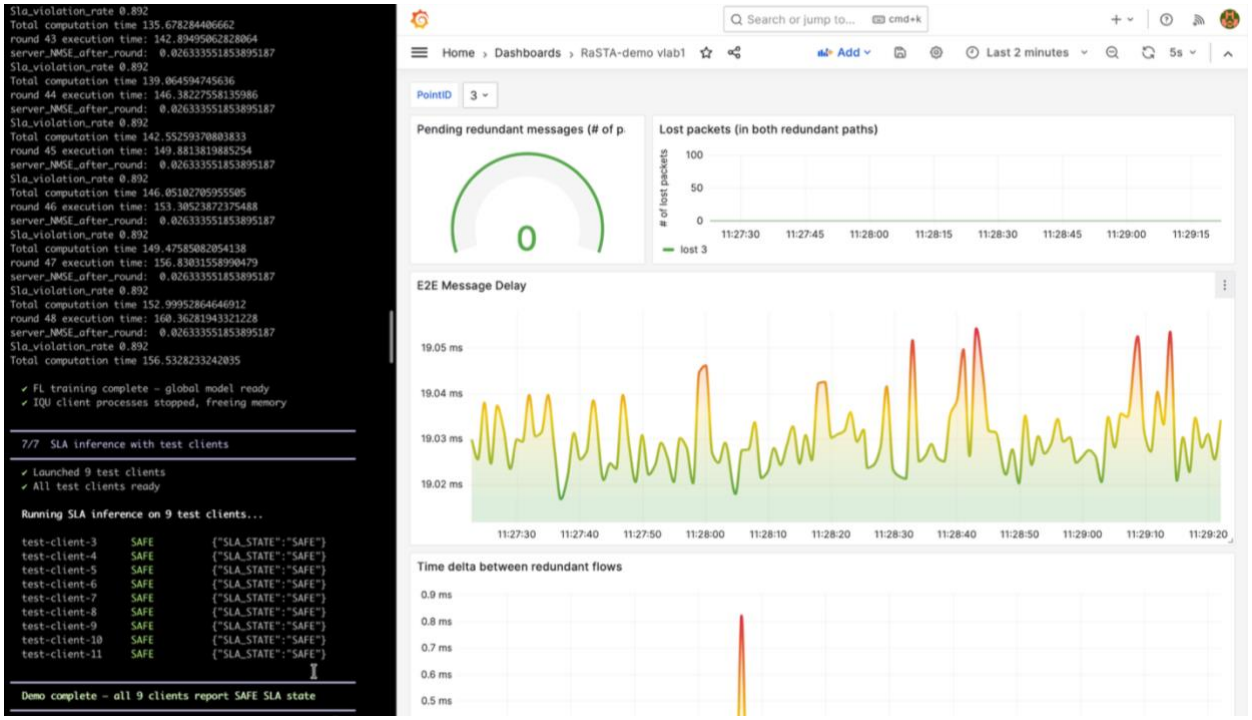


Figure 4-13. Experiment #3 inference phase monitoring.

### 4.2.4 KPI & KVI summary

To summarize, the following KPIs have been achieved:

Table 4-3: Service KPIs for PoC2b.

| KPI | Description | Target Value | Achievement description |
|---|---|---|---|
| SR-NTN-01 | Service availability | > 99.999% over longer time frames | Achieved in all experiments by continuous monitoring a stable E2E connection between SCI-Points and the Interlocking Service. In experiment #1 longer result windows were measured. |
| SR-NTN-02 | Reliability | Message loss on application layer $<10^{-5}$ | Achieved in all experiments by continuous monitoring in testbed monitoring. In experiment #1 longer result windows were measured. |
| SR-NTN-03 | Scalability | Slow degradation for even higher numbers (e.g., >10 devices/$m^2$) | Achieved in experiments #2 and #3, as the more than 10 agents set up carried out the experiment with no degradation with a shared emulated location. |
| SR-NTN-04 | Energy Efficiency | Reduction of energy consumption by >50% compared to legacy NTNs | Achieved in Testing Cycle 1 by demonstrated energy savings with |

| | | | |
|---|---|---|---|
| | | | Federated Learning and Quantization applied. |
| SR-NTN-05 | Service Provisioning Time | < 10 minutes, evidencing >90% improvement compared to 5G | Achieved both in Testing Cycle 1 and Testing Cycle 2 by measuring the service provisioning time over 10 tests. |

Table 4-4: Network KPIs for PoC2b.

| KPI | Description | Target Value | Achievement description |
|---|---|---|---|
| NR-NTN-01 | Latency / Jitter | <100 milliseconds / <50 milliseconds | Achieved in all experiments. More details in experiment #1. |
| NR-NTN-02 | Device density | <10 per $m^2$ | Achieved in experiments #2 and #3, as the more than 10 agents set up carried out the experiment with no degradation with a shared emulated location. |
| NR-NTN-03 | Packet loss rate | $5 \cdot 10^{-6}$ | Achieved in all experiments by network setup with reliable and high throughput NTN-segment. |
| NR-NTN-04 | QoE | >4.3 | Achieved in Baseline & Experiment #1<br>A questionnaire was made among the team members that used the ADROIT6G interfaces, having ranked 5 (out of 5). |

Table 4-5: Service KVIs for PoC2b.

| KVI | Description | Values | Achievement description |
|---|---|---|---|
| **KVI-NR-Poc2b-I1** | Safety | Monitoring resources from different providers to identify anomalies and intrusion detection. PoC of anomaly detection in production-grade systems, dealing with the orchestration policies of resources across multiple domains, while maintaining service quality guarantees. | SLA violations are successfully prevented through predictive SLA monitoring combined with automated closed-loop prevention mechanisms. By continuously analysing service behaviour, resource conditions, and cross-domain orchestration data, the system can detect early signs |

| | | | |
|---|---|---|---|
| | | Target value > 0.85 The KVI is related to KPIs on Anomaly detection recall and sensitivity | of degradation or anomaly before they impact service quality. This enables timely corrective actions, helping maintain service guarantees. |
| **KVI-NR-Poc2b-I3** | Regulation | # of standards involved in the PoC Execution | RaSTA standard applied, including EULYNX semantics.<br>Number of standards: 2 |
| **KVI-NR-Poc2b-D3** | Digital Inclusion | Lower latency than current procedures | NTN enables connectivity in remote and underserved areas where terrestrial networks are impractical or too costly to deploy, such as rural regions and mountainous terrain. By extending coverage beyond the limits of ground-based infrastructure, NTN makes digital services accessible to communities and operations that previously had little or no reliable connection. The latency KPI was used for the measurement of the Key Value and the target of <100 milliseconds were achieved. |
| **KVI-NR-Poc2b-D4** | Trustworthiness | >99.999% over longer time frames | For the measurement of the Key Value, the outcome of availability and reliability utilised. > 99.999% over longer time frames and message loss on application layer $<10^{-5}$ |
| **KVI-NR-Poc2b-E1** | Sustainability | SR- Poc2b -04 Reduction of energy consumption by >50% compared to legacy NTNs | Outcome of energy efficiency |
| **KVI-NR-Poc2b-E2** | Business Value | Slow degradation for even higher numbers (e.g., >10 devices/$m^2$) and <10 per $m^2$ | Outcome of scalability |
| **KVI-NR-Poc2b-E3** | Economic Growth | % decrease in operational costs | The deployment of interlocking services in remote areas via NTN, supported predictive SLA |

© ADROIT6G, 2023

| | | | |
|---|---|---|---|
| | | | monitoring combined with automated closed-loop prevention mechanisms, contributes to a percentage decrease in operational costs by reducing the need for extensive terrestrial infrastructure and simplifying service provision. |

## 4.3 Replication Guidelines

This section provides an overview of the software and hardware requirements, as well as KPI/KVI measuring methodology needed to replicate PoC2b experimentation in Testing Cycle 2.

All tests have been carried out using a powerful server (within Siemens testbed) with VM management and docker/podman container. A powerful server is a hard requirement for all of the test cases above. As for software prerequisites, similarly to Testing Cycle 1, docker/podman engine, Kubernetes tools v1.33, net-tools and other component-specific packages are required. Influx DB and Grafana are recommended to be used for monitoring. For component specific packages, please refer ADROIT6G's WP4 deliverables, relevant component list can be found in D5.4 section 3.4. PoC application must also be installed.

Upon successful software installation (e.g., docker images loading), each component must be configured to be compatible with the testbed/hardware setup. For example, influx DB tokens must be generated and propagated to all components.

Once all the components are up and running, as well as interconnected, the KPIs can be collected. In Testing Cycle 2, KPIs are automatically collected by the PoC application, as this stage only measures overall influence of ADROIT6G stack on the use case. To collect the KPIs, please open the respective PoC Grafana dashboard, where most of application and network KPIs are reported. Device density KPI can be achieved with high density emulation setup, as well as energy efficiency can be verified in the logs of quantization component energy measurements. QoS KPI can be judged personally. When it comes to KVIs, some are directly measured by the KPIs reported in the dashboard, whereas others are justified in the chapters above. The experimenter can personally judge the remainder of KVI, e.g., safety KVI.

All of the collected values can be compared to initially declared values for the ADROIT6G project, and values reported in Testing Cycle 1. For convenience, one may also use this document as a baseline.

As all integration interfaces are described in the previous ADROIT6G deliverables (D5.1/D5.2/D2.3), one may also use a custom component instead of any ADROIT6G component. All the components used (except for the NTN emulation framework) in the test cases are containerized and can be integrated with substitutes.

© ADROIT6G, 2023

# 5 Performance Evaluation of PoC3

## 5.1 Introduction

PoC3 evaluates a terrestrial 6G Collaborative Construction Robots scenario with extreme URLLC requirements, in which mobile ground robots and aerial drones cooperate on a construction site to move material, stream sensor data, and coordinate tasks such as lifting and synchronized loading. The purpose of the PoC is to assess whether the ADROIT6G platform can maintain reliable, low-latency communication for a fleet of collaborating robots under changing network load, and adapt network resources automatically as conditions change.

As in PoC2a, the experiments were executed using the CAFA Robot application as a controlled and repeatable model of the construction-robot scenario rather than a physical deployment. A fleet of camera- and lidar-equipped robots is simulated in Gazebo on a Control Station, and each robot's sensor and command traffic is relayed through Virtual Robot (VRo) nodes running on separate hosts in the ORO testbed's 5G standalone (SA) network. This generates representative and reproducible 5G traffic, allowing the behaviour of the application and of the ADROIT6G components acting on it to be measured through the application's latency and traffic logs.

The CAFA Robot application consists of two parts: a Control Station (CS) and several Virtual Robot (VRo) relay nodes. The CS runs the Gazebo simulation of the robot fleet (three robots in the reference configuration), a ROS 2 / Gazebo bridge, a stream-switch component that enables or disables each robot's video and lidar streams, and a Flask web dashboard (Figure 5-1). Each robot's camera stream is relayed out through its VRo node and back to the dashboard, while movement commands are issued from scripted scenarios. The CS timestamps each command when it is sent and computes the round-trip time (RTT) when the relay's acknowledgment returns; for video, it computes the receive latency from the frame's publish time to its arrival at the dashboard. All measurements are logged to CSV. The scenarios are time-scripted, with each robot receiving its movement and configuration commands at fixed offsets from the start of the run, which ensures that the generated traffic is repeatable across runs.

The application uses ROS 2 for messaging. Within each host, the ROS 2 nodes communicate over Eclipse CycloneDDS confined to the loopback interface, so that no DDS traffic leaves the machine; all traffic between hosts is carried by Eclipse Zenoh bridges. On every host, a Zenoh bridge container runs alongside the application container — the CS container on the Control Station host, and a relay-switch (VRo) container on each relay host — as shown in Figure 5-2. The bridges are interconnected over TCP in a full mesh, with peer endpoints configured explicitly and topic forwarding constrained by an allow-list with per-robot namespacing. The router configuration of the bridges differs between the two experiments and is described in each experiment below.

© ADROIT6G, 2023

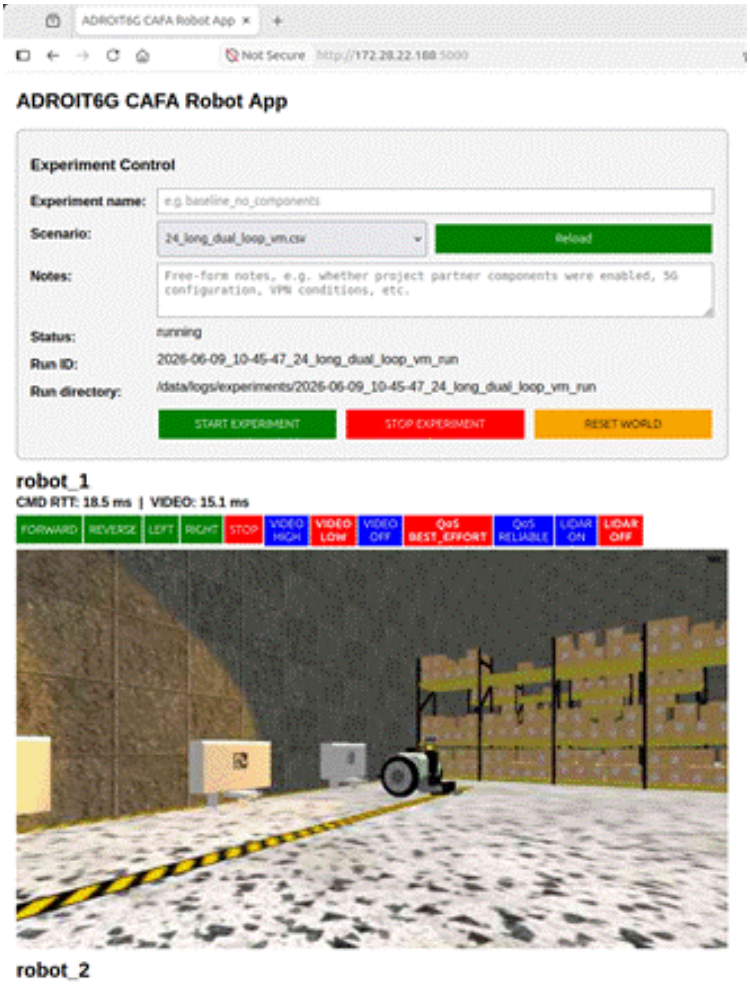


Figure 5-1. CAFA Robot application dashboard, showing the experiment control panel and a robot view with control buttons and live command/video latency.

```
cafa@tu-video-vm:~/cafa_robot_app/A6G-CAFA-Robot-app_v0.4$ sudo docker ps
CONTAINER ID   IMAGE

b6169d221a4a   eclipse/zenoh-bridge-ros2dds:latest

9fb4aa06bd89   cafa_robot_cs:cyclone_gpu_2
```

```
ubuntu@ubuntuserver:~$ sudo docker ps
[sudo] password for ubuntu:
CONTAINER ID   IMAGE
fd034660d1ef   jazzy-robot-relay-switch:cyclone
7357ff281f64   eclipse/zenoh-bridge-ros2dds:latest
```

```
ana@ana:~$ sudo docker ps
[sudo] password for ana:
CONTAINER ID   IMAGE
0c845e1a39ac   eclipse/zenoh-bridge-ros2dds:latest
cdfb52da410b   jazzy-robot-relay-switch:cyclone
```

Figure 5-2 Running Robot app containers on three hosts: the Control Station host (top) runs the CS container (cafa_robot_cs:cyclone_gpu_2) and a Zenoh bridge; each relay host (middle, bottom) runs a relay-switch (VRo) container and a Zenoh bridge.

Video is streamed using MJPEG, in which each frame is compressed independently as a JPEG image. This encoding is well suited to a robot teleoperation application: it is simple to implement, has low encoding and decoding latency because frames carry no inter-frame dependencies, and degrades gracefully under constrained bandwidth, since a dropped or delayed frame does not affect the decoding of subsequent frames and produces no tearing or propagated artefacts. The trade-off, examined in the results below, is a higher and more variable bitrate than inter-frame codecs such as H.264, because the per-frame payload depends on scene content.

The evaluation covers the two experiments defined for PoC3 in D5.4:

- Experiment #1: Autonomous Transport with Dynamic Re-Routing. Robots follow scripted routes and continuously upload video and, optionally, lidar data, producing sustained network traffic. The experiment exercises the application together with the ADROIT6G Service Orchestration & Resource Management, Closed-Loop Governance (CLG), and AI/ML Pipeline Manager & Catalogue components contributed by ORO and NXW, which monitor the application traffic and adjust the network-slice resources accordingly.
- Experiment #2: Drone–AGV Collaborative working with D2D. The application's inter-node transport is reconfigured so that robot nodes can relay traffic for one another, modelling the device-to-device mesh in which devices act as relays to maintain connectivity.

## 5.2 Evaluation Results

### 5.2.1 Experiment 1: Autonomous Transport with Dynamic Re-Routing Loading

**Objectives**: From the application point of view, the objective was to demonstrate that the robot fleet can autonomously execute its transport routes while communication remains reliable and low-latency under a continuous sensor-streaming load. From the testbed and component point of view, the objective was to showcase network-slice automation and performance optimization for collaborative robots through an AI-enabled predictive closed-loop, reconfiguring the network slice when high traffic load is predicted.

Showcase network slice automation and performance optimization for collaborative robots through AI-enabled predictive closed-loop.

**Test description:**

- Instantiate the service through the Service Orchestrator and deploy the Robot Application components through the Resource Orchestrator.
- Configure the Network Slice profile through the 5G Slice Management System.
- Instantiate the closed-loop through the Closed-Loop Governance and deploy the related closed-loop functions through the Resource Orchestrator along with the AI/ML models through the AI/ML Framework. During this deployment, MERGE dynamically assembles the resource-optimized Random Forest model.
- Start a scenario for the Robot Application where the robots move along some assigned routes and continuously upload video.
- Monitor Robot Application performance throughout experiment.
- Whenever the AI/ML enabled closed-loop functions detect a network congestion, the network slice is re-configured through the 5G Slice Management System in order to upgrade the slice profile.

© ADROIT6G, 2023

- Collect the application results for KPI measurements.
- Terminate the service.
- Collect the service provisioning/decommissioning and closed-loop operations time results for KPI measurements.

The Robot application run is started from the dashboard. In Experiment #1 the Zenoh bridges are configured with only the Control Station bridge acting as a router, so traffic between each VRo host and the CS is carried directly over that bridge. Before each run, RESET WORLD returns the robots to their initial positions and default sensor state. The operator enters an experiment name and optional notes, selects a scenario file in which the robots move along assigned routes while continuously uploading video, and presses START EXPERIMENT. The scenario then executes automatically, the command and video latencies are displayed live and logged, and the run stops when the scenario finishes.

**Results**: The service blueprint for collaborative construction robots and the descriptor of the AI-enabled predictive closed-loop are shown in Figure 5-3 and Figure 5-4, respectively. The service blueprint contains information about the robot application components, i.e. three virtual robots and a control station, and the endpoints for their connectivity. The descriptor of the closed-loop, and of the related functions, highlights the aim of reconfiguring the network slice upon high traffic load prediction.

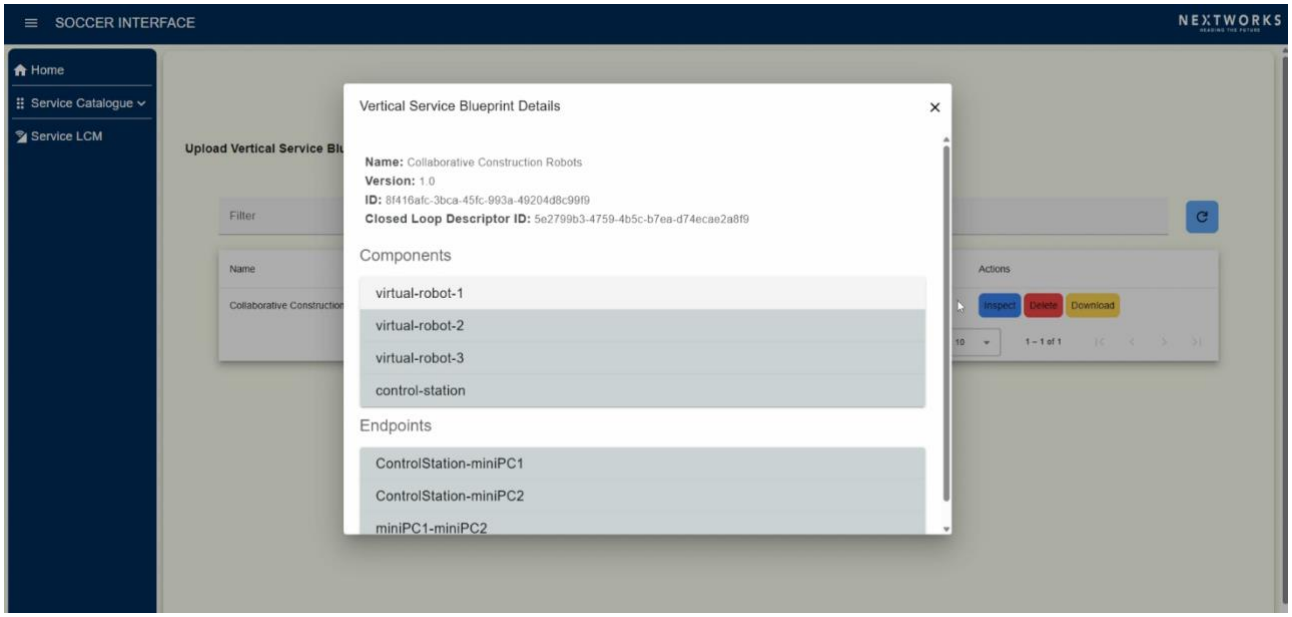


Figure 5-3. PoC3 Experiment #1 Collaborative Construction Robots Service Blueprint onboarded in the Service Catalogue.

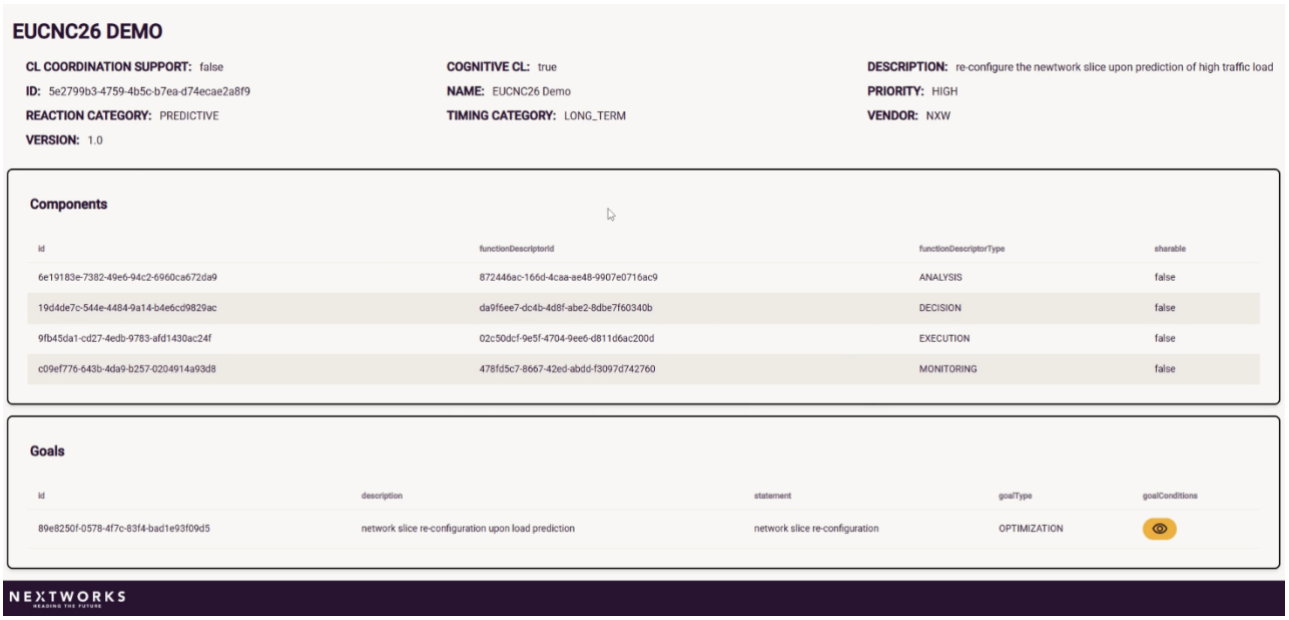


Figure 5-4. PoC3 Experiment #1 Closed-Loop descriptor onboarded in the Closed-Loop Catalogue.

Once the instantiation of the service, created by providing the MQTT topics for the closed/loop functions, the network slice ID, the AI models thresholds and sample size, and the endpoint for slice re-configuration as input parameters, is requested, the network slice profile is created and both application components and the closed-loop functions (Figure 5-5) are deployed in the Kubernetes cluster. Two AI models have been deployed in Kubernetes, too (Figure 5-6): an ANFIS, deployed by the analysis closed-loop function, to predict the network slice performance score and a Random Forest, deployed by the decision closed-loop function, to classify ANFIS output in order to identify network slice congestion. Their average inference time has been computed over 10 iterations with a result of 0.0174 seconds for the ANFIS model and a result of 0.0155 seconds for the RF model.

Hence, the Robot application starts running and a scenario is uploaded to make robots moving and streaming video, as shown in Figure 5-5.

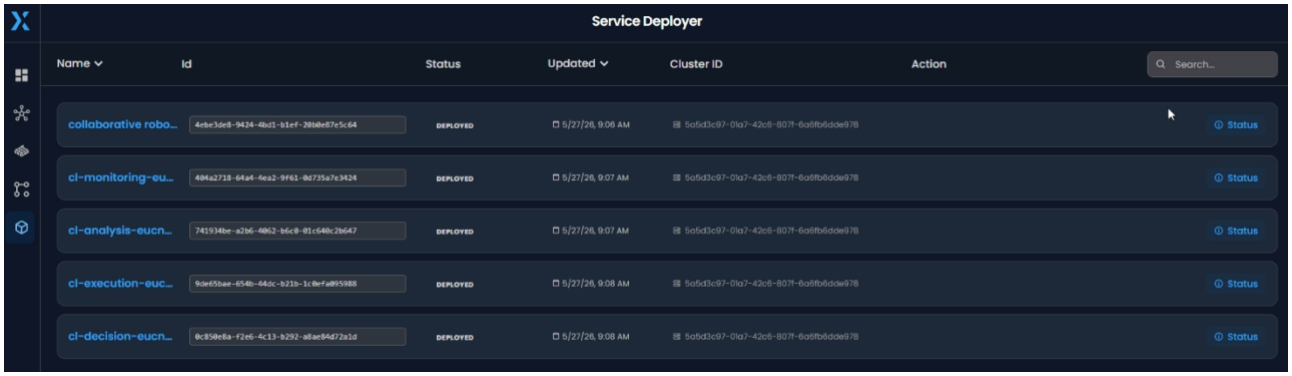


Figure 5-5. PoC3 Experiment #1 Application components and Closed-Loop functions deployed through the Resource Orchestrator.

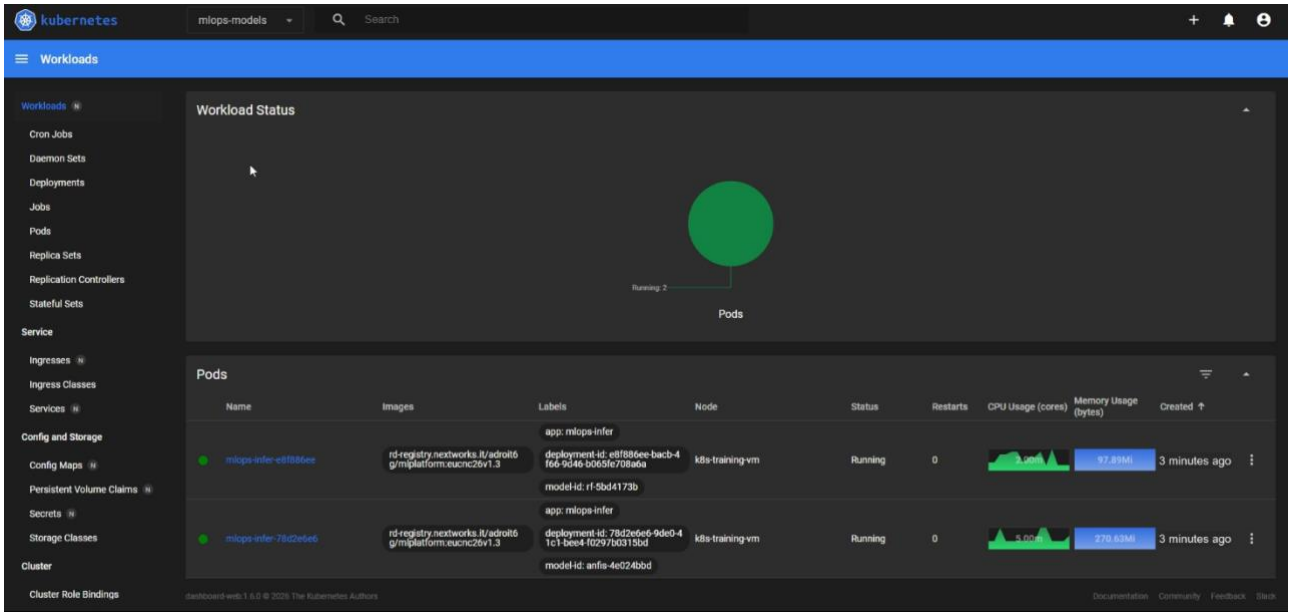


Figure 5-6. PoC3 Experiment #1 AI models deployed in the Kubernetes cluster.

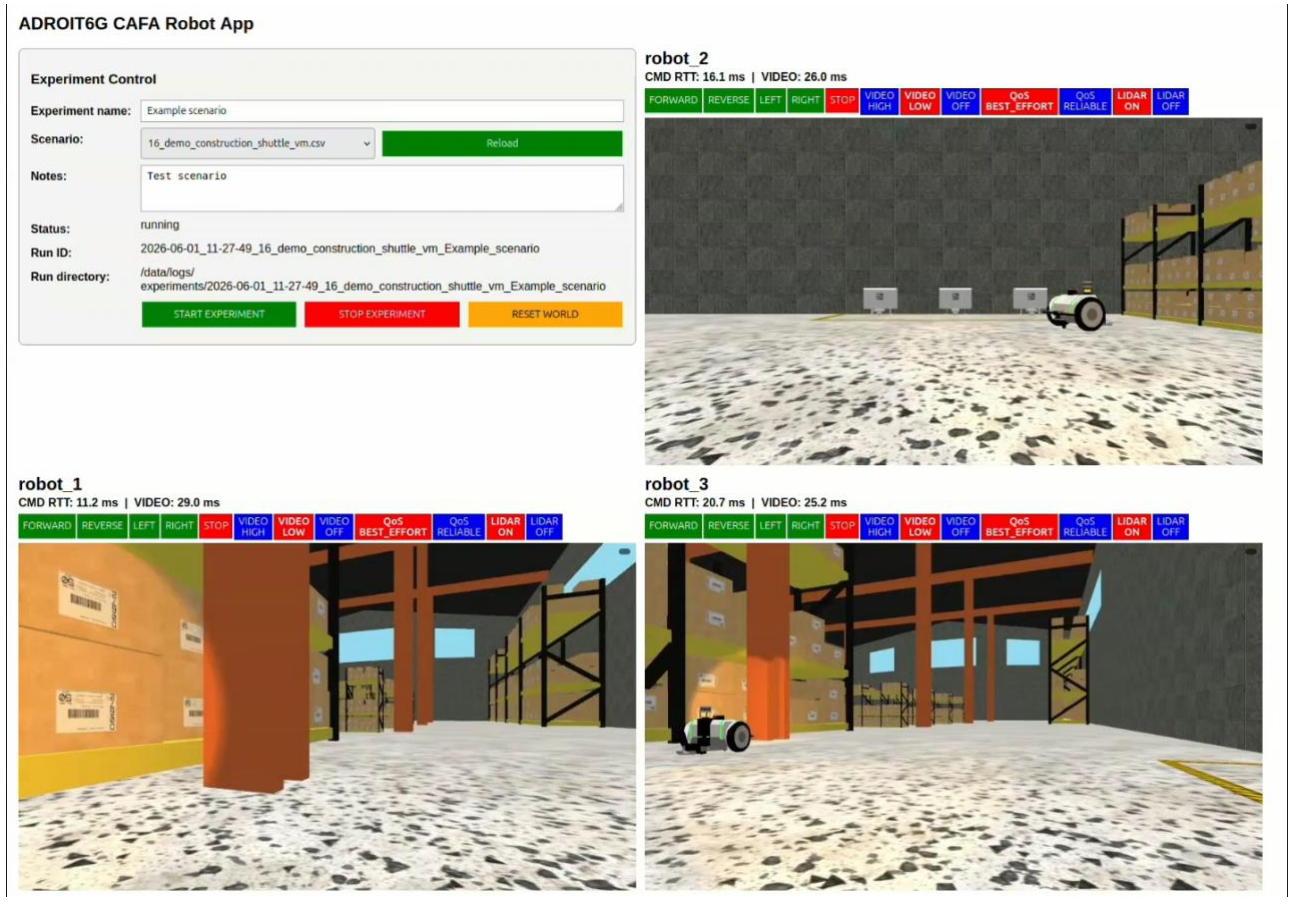


Figure 5-7. PoC3 Experiment #1 Robot application scenario.

The deployed Random Forest is a pre-assembled, MERGE-driven aggregated model retrieved directly from the catalogue. Prior to deployment, this model was constructed by selecting and combining optimal decision trees based on the target node's computational constraints. To achieve this, MERGE deconstructs pre-trained RF models of varying complexities into a catalogue of individual trees, annotating each with metadata for quality and a hardware-independent computational cost metric (average path nodes). These baseline models span five distinct

configurations, scaling in complexity from a minimal "tiny" model (1 estimator, max depth 1) up to a complex "xlarge" model (200 estimators, unlimited depth). To preserve ensemble diversity, the system selects trees strictly based on their validation F1 score (a balanced metric combining precision and recall to provide a more rigorous measure of model accuracy) aggregating them until the target budget is reached. Because the baseline models already exceed 99% accuracy, performance was measured via relative F1 error reduction rather than flat percentages, yielding highly successful benchmarks. Ultimately, the MERGE ensemble reduced the F1 error by up to 5.9% compared to the best available models under the same strict budget, as shown in Table 5-1, while simultaneously matching the quality of standard medium-complexity models and slashing computational costs by up to 89.6%, as shown in Table 5-2.

Table 5-1 Performance comparison of the MERGE-aggregated model versus the best baseline model under the same computational budget.

| Budget | Baseline | Δ Accuracy | Δ Precision | Δ Recall | Δ F1 |
|---|---|---|---|---|---|
| 10 | rf_tiny | +5.88% | +0.06% | +100.00% | +5.91% |
| 20 | rf_small | +5.88% | +0.06% | +100.00% | +5.91% |
| 25 | rf_small | +5.88% | +0.06% | +100.00% | +5.91% |
| 50 | rf_small | +5.88% | +0.06% | +100.00% | +5.91% |
| 100 | rf_medium | 0.00% | 0.00% | 0.00% | 0.00% |

Table 5-2 Computational cost reduction of the MERGE-aggregated model compared to a baseline model of equivalent quality.

| Budget | Baseline | Δ F1 | Δ Compute Cost |
|---|---|---|---|
| 10 | rf_medium | ≈ 0% | -89.6% |
| 20 | rf_medium | ≈ 0% | -80.0% |
| 25 | rf_medium | ≈ 0% | -74.1% |
| 50 | rf_medium | ≈ 0% | -47.4% |
| 100 | rf_medium | ≈ 0% | +6.4% |

At this point, the environment and the performance of the application is monitored: the monitoring closed-loop function continuously retrieves slice metrics and make them available to the closed-loop. The analysis function invokes the ANFIS model to predict future state of metrics. The decision function invokes the classifier ML model and once the traffic gets higher, i.e., a network congestion is identified, it triggers a scaling operation, as shown in Figure 5-8. Thus, the execution function requests the re-configuration of the network slice, as shown in Figure 5-9.

© ADROIT6G, 2023

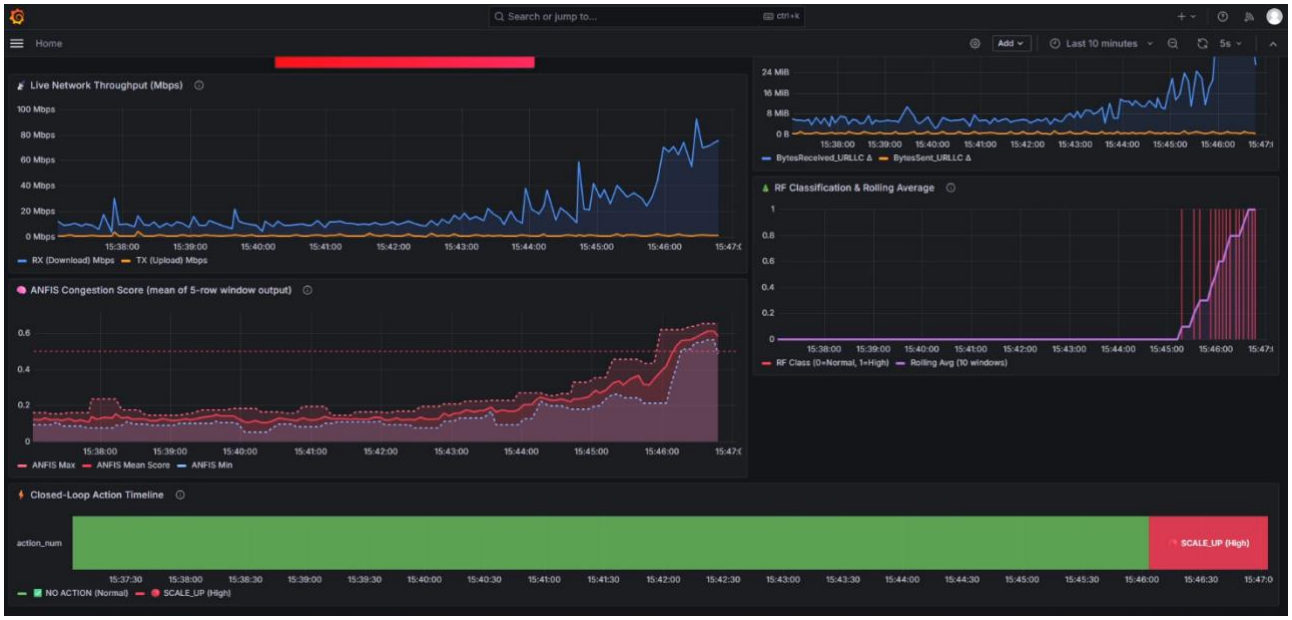


Figure 5-8. PoC3 Experiment #1 Grafana dashboard monitoring the traffic and showing the current congestion score.

Figure 5-9. PoC3 Experiment #1 Closed-Loop functions logs showing network slice re-configuration request.

The instantiation process was executed 10 times to measure the time needed for the Network Slice profile to be created and the time needed for the deployment of both Robot Application components and closed-loop. The average times from these tests are reported in Table 5-3. The total instantiation time for a closed-loop instance accounts for the complete setup of every closed-loop function, which specifically includes:

- Deploying the function within the Kubernetes cluster (and the ML model for the analysis and decision functions).
- Subscribing to Message Queuing Telemetry Transport (MQTT).
- Sending a notification to the Service Orchestrator.

The achieved mean total instantiation time of 214.138 seconds is substantially lower than the 10-minute requirement, thereby fulfilling the targeted KPI for service provisioning time.

Table 5-3. PoC3 Experiment #1 Performance of Service Orchestrator and Closed-Loop Governance.

| Instance | Instantiation time (s) |
|---|---|
| Vertical service | 214.138 |
| Resource Allocation | 1.245 |
| Network Slice | 0.092 |
| Robot application | 61.153 |
| Closed-loop instance | 145.415 |
| Closed-loop monitoring function | 36.084 |
| Closed-loop analysis function | 36.694 |
| Closed-loop decision function | 36.683 |
| Closed-loop execution function | 35.82 |

During initial deployment the application used FastDDS over a WireGuard VPN between hosts, in which configuration the inter-node communication was unreliable. It was migrated to Zenoh bridges with CycloneDDS, which is better suited to a routed 5G network, and this resolved the problem. On an identical high-load scenario, the migration reduced the median command RTT from approximately 184 ms to approximately 21 ms and the median video latency from approximately 146 ms to approximately 75 ms (Figure 5-10), and removed a tail of multi-second latencies present in the earlier configuration. This illustrates how the application's transport layer can be adapted to operate efficiently over a 5G network; the transport selection is internal to the application and is not part of the ADROIT6G evaluation, so the FastDDS/WireGuard runs are treated as a non-functional early configuration rather than as a performance baseline, and the measurements below are taken from the working Zenoh configuration.

© ADROIT6G, 2023

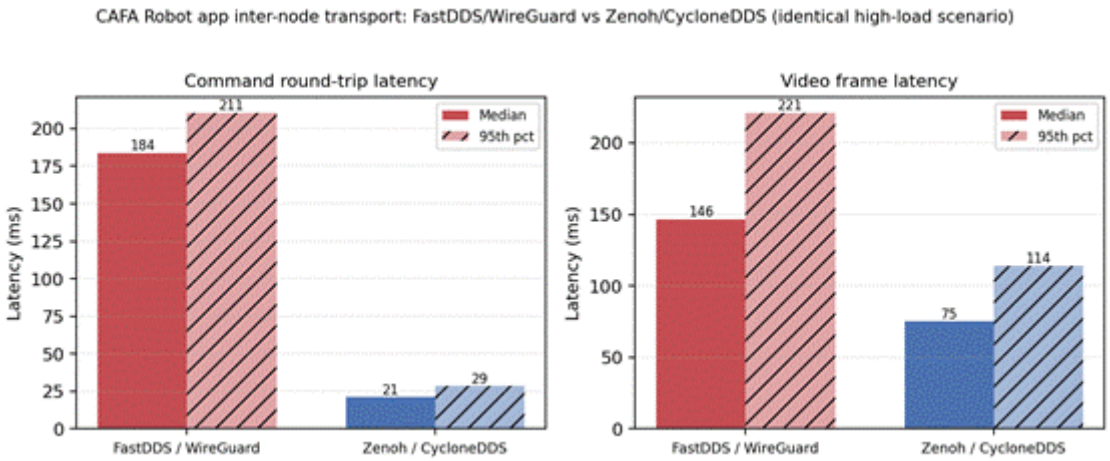


Figure 5-10. CAFA Robot application inter-node transport — FastDDS/WireGuard compared with Zenoh/CycloneDDS on an identical high-load scenario (median and 95th-percentile latency).

Because the application streams video using MJPEG, the per-frame payload depends on scene content and the bitrate is variable. In the high-quality video runs the per-second application payload varied substantially, with peaks of approximately 15 Mbps against a mean of approximately 9 Mbps. Figure 5-11 illustrates this for a high-quality run, in which the video latency closely tracks the instantaneous bitrate and settles only as the bitrate decreases.

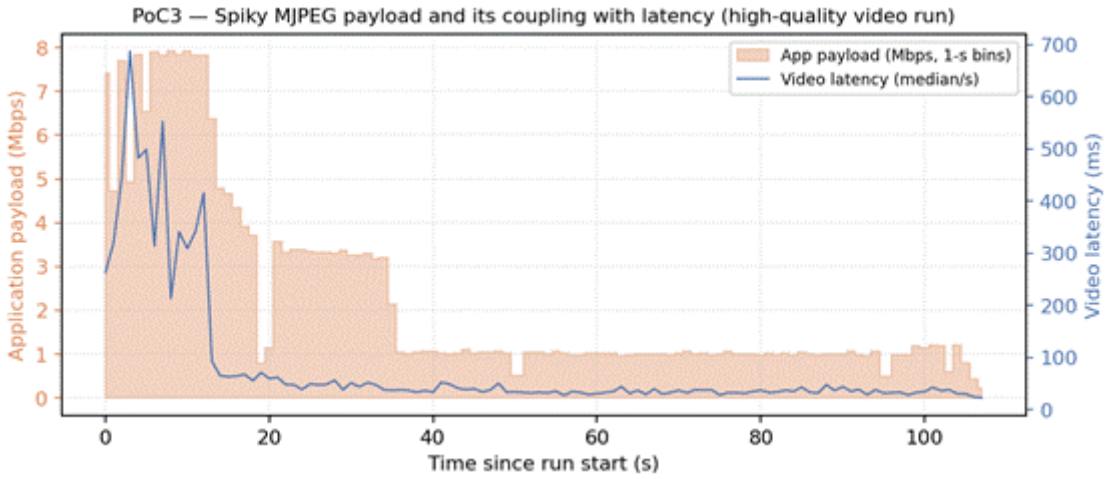


Figure 5-11. MJPEG application payload (per-second bins) and its relationship with video latency during a high-quality video run.

The NXW AI/ML components predict the application traffic and allocate slice resources to match it. The expected bitrate at which the components operated without scaling up had been set conservatively, and was exceeded during normal operation of the application, which triggered the components to allocate a higher resource state. To keep the application traffic within the expected envelope, the video quality — resolution, frame rate, and JPEG quality — was reduced over the course of the campaign. Figure 5-12 shows the effect: from high-quality video at several Mbps per robot, to approximately 1 Mbps for the original low-quality profile, and to approximately 0.5 Mbps once that profile was reduced further. This relationship is necessary context for interpreting the latency results, since latency and payload are coupled: in some runs the latency decreased mainly as a result of the reduced payload, while in others it increased because the payload exceeded the capacity available at that moment.

© ADROIT6G, 2023

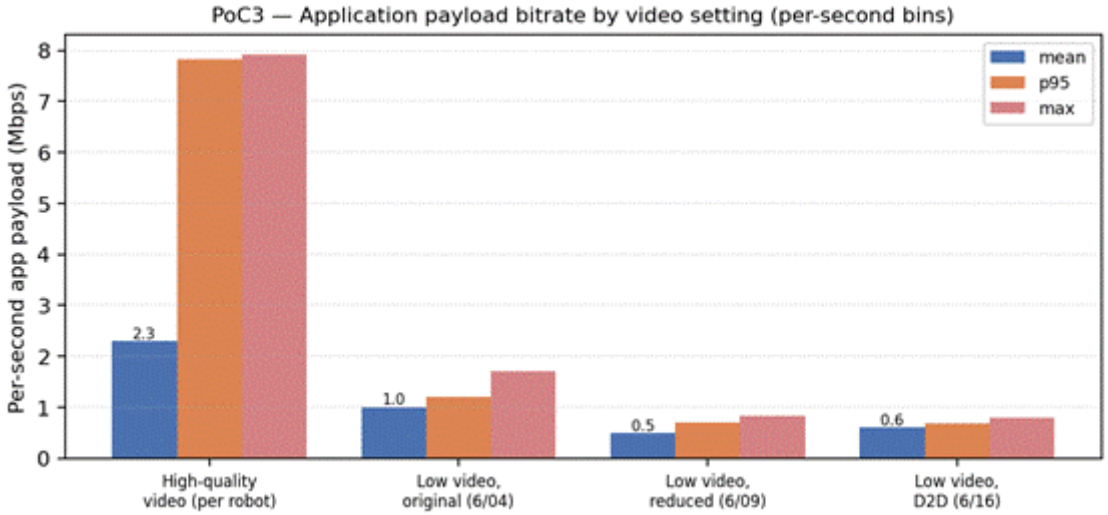


Figure 5-12. Application payload bitrate by video setting (per-second bins): mean, 95th percentile, and maximum.

In the working Zenoh configuration at the reduced video setting, the application was stable and responsive, with a median video latency of approximately 21 ms (95th percentile approximately 35 ms) and a median command RTT of approximately 20 ms. Runs of ten minutes or longer completed without dropped sessions and with only occasional brief gaps in frame delivery. The main observations from the application measurements are:

- The CAFA Robot application operated as a repeatable traffic model of autonomous transport over the ORO 5G SA testbed.
- The MJPEG video stream produces a variable bitrate, and the video latency tracks the instantaneous bitrate closely.
- The NXW AI/ML components allocated network-slice resources in response to the application traffic, scaling to a higher resource state when the traffic exceeded the expected envelope.
- The video quality was tuned to remain within the expected bitrate envelope; because latency and payload are coupled, the two must be interpreted together.

### 5.2.2 Experiment 2: Drone–AGV Collaborative working with D2D

**Objectives**: From the application point of view, the objective was to model low-latency D2D coordination between collaborating devices, in which devices form a relay mesh to maintain connectivity rather than each relying solely on its own link to the Control Station. From the testbed and component point of view, the objective was to showcase precise, low-latency D2D coordination between devices with safe fallback in case of link degradation, leveraging the distributed AI and orchestration components (BDIx Agents, UE-VBS, Topology Module, CLG, and the AI/ML Pipeline Manager & Catalogue).

**Test description:** In the Robot application, the D2D configuration is enabled by configuring all Zenoh bridges as routers, rather than only the Control Station bridge as in Experiment #1. With every bridge routing, the bridge at each robot's VRo node can carry traffic on behalf of other nodes, so that if the direct link between a robot node and the CS is unavailable, that robot's traffic can be carried through another robot node acting as a relay. The run is started from the dashboard in the same way as Experiment #1, after applying the all-routers configuration; the experiment notes field records the configuration in use (Figure 5-13). The distributed AI and orchestration components are

configured and operated by ORO/NXW as defined for this experiment in ADROIT6G's deliverable D5.4.

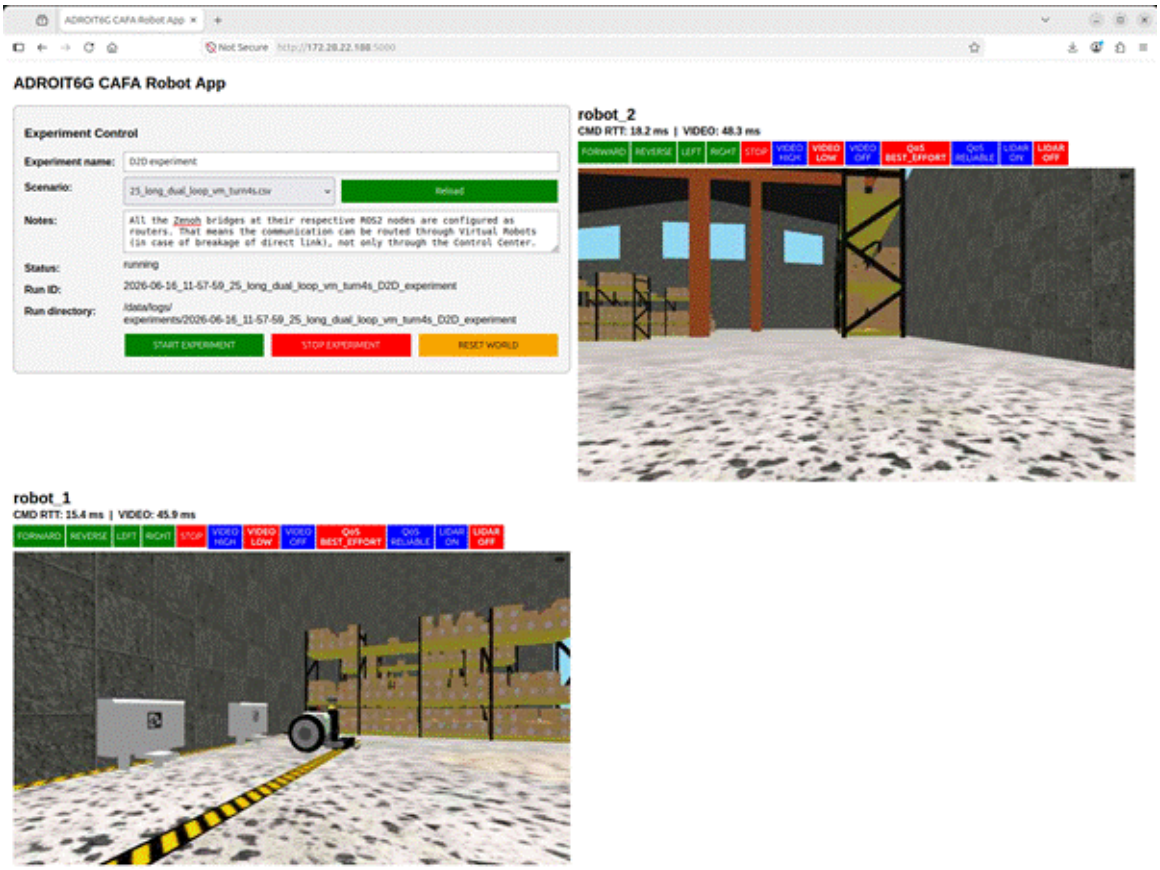


Figure 5-13. CAFA Robot application dashboard during the Experiment #2 (D2D) run, with two robots active and all Zenoh bridges configured as routers (see the experiment notes field).

**Results**: A representative ten-minute, two-robot run in the all-routers configuration remained stable throughout, with a median video latency of approximately 28 ms (95th percentile approximately 62 ms) and a median command RTT of approximately 33 ms (95th percentile approximately 67 ms). Frames were delivered continuously at the expected half-second cadence, with only a small number of brief gaps over the full run and no dropped sessions. Figure 5-14 shows the latency for both robots across the run; the absence of upward drift or sustained gaps indicates that introducing the relay capability did not degrade responsiveness or continuity. The main observations are:

- Configuring every Zenoh bridge as a router allows robot nodes to relay traffic for one another, modelling D2D mesh relaying at the transport level.
- Application latency and continuity remained within the same range as the direct-path configuration, confirming that the relay capability did not degrade the service.
- The experiment validates the D2D collaborative user story at the application and transport level, complementing the BDIx/UE-VBS components.

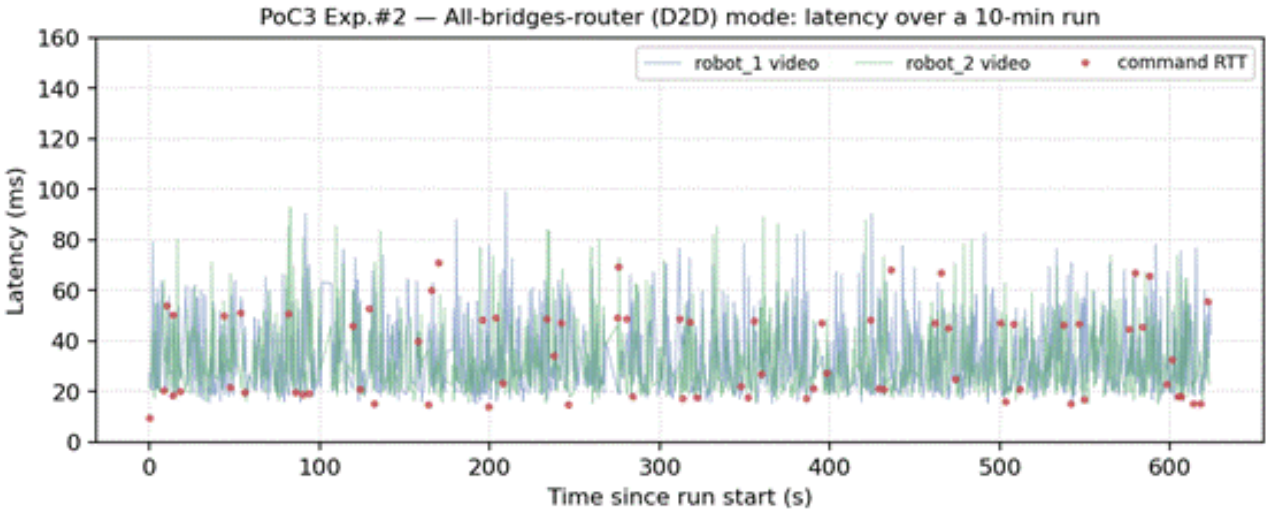


Figure 5-14. All-bridges-router (D2D) configuration: per-robot video latency and command RTT over a 10-minute run.

## 5.3 KPI/KVI evaluation

The following tables summarize the PoC3 KPI and KVI evaluation results. The targets are taken from the PoC2a KPI/KVI definitions established in D5.4, while the achievement descriptions reflect the final successful experiment runs.

Table 5-4. Service KPIs for PoC3.

| KPI | Description | Target Value | Achievement description |
|---|---|---|---|
| SR-Poc3-01 | Latency | Communication delay <10 ms | Achieved in both experiments by continuous monitoring a stable E2E connection between Cloud/Edge servers and robots. |
| SR-Poc3-02 | Reliability | SLA 99.95% | Achieved in both experiments by continuous monitoring in testbed during all tests. |
| SR-Poc3-03 | Energy Efficiency | Reduction of energy consumption by >30% compared to legacy | Achieved in both experiments by measuring the consumption and comparing with legacy values. |

Table 5-5. Network KPIs for PoC3.

| KPI | Description | Target Value | Achievement description |
|---|---|---|---|
| NR-Poc3-01 | Jitter of Terrestrial Network | <1ms | Achieved in indoor (industrial site) communication for both experiments |

| | | | |
|---|---|---|---|
| | | | in special network configuration (radio and core network part) |
| NR-Poc3-02 | Localisation accuracy using 5G/ 6G network | <3 m | Achieved in indoor (industrial site) localization, <1m for both experiments |
| NR-Poc3-03 | Upload and download speeds | ≥10 Mbps per device | Achieved in both experiments by measuring in the testbed during experiments run. |

Table 5-6. KVIs for PoC3.

| KVI | Description | Values/Scaling | Achievement description |
|---|---|---|---|
| KVI-R-PoC3-I1 | Safety | User-Centric perception Low / Medium / High | Achieved. All components are developed to have sensors and specifics procedure to stop anything in case of issue. All experiment has been conducted in lab, in safety scenarios. |
| KVI-R-PoC3-I2 | Security | User-Centric perception Low / Medium / High | Achieved. All network components and devices are using specific authentication methods and protocols to ensure that entire communication is safe and encrypted. |
| KVI-R-PoC3-I3 | Regulation | # Standards used for definitions, requirements etc | Achieved. The PoC used standard-aligned interfaces and components, including RAN and Core/Edge network part, Kubernetes-based cloud-native deployment, Prometheus/Grafana monitoring, ROS2/Gazebo robot integration, and OAI-based 5G/6G emulation. |
| KVI-R-PoC3-I4 | Responsibility | 0 users manipulation incidences | Achieved. Having sensors and scenarios that can prevent manipulation issue, no incident occurs. |

| | | | |
|---|---|---|---|
| KVI-R-PoC3-I5 | Energy efficiency | Reduction of energy consumption by >30% compared to legacy | Achieved. Energy consumption reduces with minimum 30% based on new technologies and protocols that can efficiently allocate the power need and consumption. |
| KVI-R-PoC3-D1 | Privacy | User-Centric perception Low / Medium / High | Achieved. For both scenarios and entire PoC, all of manoeuvres are made inside a secure facility and private network components are in place. |
| KVI-R-PoC3-D2 | Fairness | User-Centric perception Low / Medium / High | Achieved. Orchestration, slicing, and communication policies are applied uniformly across all connected robots, with no per-user prioritization. |
| KVI-R-PoC3-D3 | Digital Inclusion | 99.9% communication service reliability | Achieved. The ADROIT6G-enabled configuration achieved 99.99% reliability of service. |
| KVI-R-PoC3-D4 | Trustworthiness | 100% consistent coverage of the network | Achieved. The ADROIT6G-enabled configuration achieved 99.99% reliability of network and compute components and maintained 100% network coverage in site. |
| KVI-R-PoC3-E1 | Sustainability | Reduction of energy consumption by >30% compared to legacy | Achieved. Energy consumption reduces with minimum 30% based on new technologies and protocols that can efficiently allocate the power need and consumption. |
| KVI-R-PoC3-E2 | Business Value | User-Centric perception Low / Medium / High | Achieved. The combination of hardware and software components, network and software that can be installed and configured to work and hard environments for people, can add many value |

© ADROIT6G, 2023

| | | | |
|---|---|---|---|
| | | | to these types of use cases. (e.g. industrial) |
| KVI-R-PoC3-E3 | Economic Growth | User-Centric perception Low / Medium / High | Achieved. The combination of predictive analytics, remote teleoperation, and closed-loop orchestration can reduce manual intervention, support earlier maintenance decisions, and improve operational efficiency. |
| KVI-R-PoC3-E4 | Open collaboration | Number of Devices connected | Achieved. Four devices (one Control Station and three Virtual Robots) connected and interoperated concurrently over standard, open components. |
| KVI-R-PoC3-E5 | New Value Chain | Number of Devices connected | Achieved. Reliable low-latency coordination of four connected devices (one Control Station and three Virtual Robots) demonstrates the basis for collaborative-robotics value chains on shared 6G infrastructure. |

## 5.4 Replication Guidelines

This section describes the software and hardware required to replicate the PoC3 experimentation, together with the procedure for executing the experiments and collecting the KPI/KVI evidence.

The experiments are replicated on the ORO 5G standalone testbed. The hardware requirements follow those defined in D5.4 (Section 5.3): a 5G NR gNodeB (FR1, sub-6 GHz) supporting network slicing and edge computing; an edge computing server (multi-core CPU, at least 32 GB RAM, a GPU recommended for the Gazebo simulation, and Kubernetes/Docker support) hosting the Control Station and the ADROIT6G components; and one host per Virtual Robot relay, each connected to the 5G network. In the reported campaign the Control Station ran on a GPU-equipped virtual machine and the relays on mini-PCs; any host meeting these requirements and able to reach the 5G network is suitable.

Each host requires a Docker engine and connectivity to the 5G SA network. The CAFA Robot application is deployed as containers: on the Control Station host, the CS container (Gazebo, ROS 2, the ROS 2 / Gazebo bridge, the stream-switch component, and the Flask dashboard) together with a Zenoh bridge container (eclipse/zenoh-bridge-ros2dds); and on each relay host, the relay-switch (VRo) container together with its own Zenoh bridge container. Within every host the ROS 2 nodes communicate over Eclipse CycloneDDS confined to the loopback interface, using an identical configuration on every container so that no DDS traffic leaves the machine, while all inter-host

© ADROIT6G, 2023

traffic is carried by the Zenoh bridges. The bridges are interconnected over TCP in a full mesh, with multicast scouting disabled and peer endpoints configured explicitly, and with topic forwarding constrained by an allow-list and per-robot namespacing. For Experiment #1, only the Control Station bridge is configured as a router; for Experiment #2, every bridge is configured as a router so that relay hosts can carry traffic on behalf of one another.

The ADROIT6G components used in PoC3 — Service Orchestration & Resource Management, Closed-Loop Governance, and the AI/ML Pipeline Manager & Catalogue — are deployed and configured by ORO and NXW on the edge server, as described in the WP4 deliverables and in the PoC3 component list in D5.4 (Section 5.1). These components monitor the application traffic and reconfigure the network slice accordingly; their installation and configuration is the responsibility of the respective component owners. Because the integration interfaces are documented in the earlier ADROIT6G deliverables (D5.1, D5.2, D2.3), individual ADROIT6G components may be substituted by compatible alternatives.

Once all components are running and interconnected, experiments are started from the CAFA Robot application dashboard. Before each run, RESET WORLD returns the robots to their initial positions and default sensor state. The operator enters an experiment name and optional notes (for example, the transport configuration and 5G conditions), selects a scenario file, and presses START EXPERIMENT; the scenario then executes automatically, latency values are displayed live, and the run stops when the scenario finishes. The same scenario files should be used across configurations so that the generated traffic is repeatable and comparable, in particular when comparing operation with and without the ADROIT6G components.

Each run creates a directory under /data/logs/experiments containing five files: cmd_latency.csv (command round-trip latency), video_latency.csv (video-frame receive latency and per-frame JPEG payload size), events.csv (scenario and run events), scenario.csv (the executed scenario), and meta.json (run metadata). These logs constitute the minimum evidence required to reproduce the application-level KPI assessment: the command and video latency series support the latency and reliability KPIs, the per-frame payload field supports the throughput assessment, and the events log allows the latency to be related to the scenario phases.

The replicated results can be compared against the PoC3 KPI/KVI targets defined in D5.4 and against the values reported in this deliverable. When interpreting the results, two properties of the logging should be kept in mind: the logged video latency is the application-side receive latency rather than glass-to-glass or pure wire latency, and the logged throughput is the application JPEG payload rather than the true wire rate. Because the MJPEG bitrate is variable and coupled with latency, runs should be compared at the same video setting, and the bitrate envelope expected by the NXW components should be considered. Localization accuracy (NR-Poc3-02) cannot be reproduced with the current application, as positioning is not implemented, and the energy-efficiency KPI (SR-Poc3-03) is measured by ORO/NXW rather than by the application.

© ADROIT6G, 2023

# 6 Conclusion

This deliverable has presented the final performance evaluation results of the ADROIT6G components and algorithms developed throughout the project. Building upon the implementations described in the WP3 and WP4 deliverables, the second testing cycle assessed all ADROIT6G PoCs against the reference network-level KPIs and corresponding KVIs, providing a comprehensive validation of their functionality and performance.

For each PoC, a set of representative experiments was conducted, documenting the evaluation objectives, testing methodology, and measured performance results. The reported outcomes demonstrate the capabilities of the proposed solutions and provide quantitative evidence of their effectiveness under the considered experimental conditions. Furthermore, the inclusion of detailed replication guidelines enhances the reproducibility of the presented work, enabling other testbeds to reproduce the experiments, validate the reported results, and assess the portability of the developed solutions in different environments.

Overall, the activities reported in this deliverable conclude the WP5 performance evaluation effort, providing the final validation of the ADROIT6G innovations and supporting the project's objective of delivering experimentally verified, reproducible, and portable Beyond 5G/6G technologies.

.

© ADROIT6G, 2023