This project has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No. 101069732

aerOS
EUROPEAN IOT-EDGE-CLOUD

D5.6 – Technical evaluation, validation and assessment report (2)

Deliverable No.	**D5.6**	**Due Date**	**31-OCT-2025**
Type	**Report**	**Dissemination Level**	**Public** **(PU)**
Version	**1.0**	**WP**	**WP5**
Description	*Document summarising the validation approach and methodology adopted by the aerOS consortium for assessing the performance of the integrated aerOS ecosystem, including the evaluation of technical, pilot, and impact KPIs, as well as the verification of system requirements and Key Value Indicators across all pilots.*		

Copyright

The aerOS consortium consists of the following 27 partners:

UNIVERSITAT POLITÈCNICA DE VALÈNCIA	ES
NATIONAL CENTER FOR SCIENTIFIC RESEARCH "DEMOKRITOS"	EL
ASOCIACION DE EMPRESAS TECNOLOGICAS INNOVALIA	ES
TTCONTROL GMBH	AT
TTTECH COMPUTERTECHNIK AG (*third linked party*)	AT
SIEMENS AKTIENGELSCHAFT	DE
FIWARE FOUNDATION EV	DE
TELEFONICA INVESTIGACION Y DESARROLLO SA	ES
COSMOTE KINITES TILEPIKOINONIES AE	EL
EIGHT BELLS LTD	CY
INQBIT INNOVATIONS SRL	RO
FOGUS INNOVATIONS & SERVICES P.C.	EL
L.M. ERICSSON LIMITED	IE
SYSTEMS RESEARCH INSTITUTE OF THE POLISH ACADEMY OF SCIENCES IBS PAN	PL
ICTFICIAL OY	FI
INFOLYSIS P.C.	EL
PRODEVELOP SL	ES
EUROGATE CONTAINER TERMINAL LIMASSOL LIMITED	CY
TECHNOLOGIKO PANEPISTIMIO KYPROU	CY
DS TECH SRL	IT
GRUPO S 21SEC GESTION SA	ES
JOHN DEERE GMBH & CO. KG*JD	DE
CLOUDFERRO S.A.	PL
TRUM SP ZOO	PL
POLITECNICO DI MILANO	IT
MADE SCARL	IT
NAVARRA DE SERVICIOS Y TECNOLOGIAS SA	ES
SWITZERLAND INNOVATION PARK BIEL/BIENNE AG	CH

Disclaimer

This document contains material, which is the copyright of certain aerOS consortium parties, and may not be reproduced or copied without permission. This deliverable contains original unpublished work except where clearly indicated otherwise.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.

The information contained in this document is the proprietary confidential information of the aerOS Consortium (including the Commission Services) and may not be disclosed except in accordance with the Consortium Agreement. The commercial use of any information contained in this document may require a license from the proprietor of that information.

Neither the Project Consortium as a whole nor a certain party of the Consortium warrant that the information contained in this document is capable of use, nor that use of the information is free from risk and accepts no liability for loss or damage suffered by any person using this information.

The information in this document is subject to change without notice.

The content of this report reflects only the authors' view. The Directorate-General for Communications Networks, Content and Technology, Resources and Support, Administration and Finance (DG-CONNECT) is not responsible for any use that may be made of the information it contains.

Authors

Name	Partner	e-mail
Carlos E. Palau	P01 UPV	cpalau@dcom.upv.es
Ignacio Lacalle	P01 UPV	iglaub@upv.es
Rafael Vañó	P01 UPV	ravagar2@upv.es
Raúl San Julián	P01 UPV	rausanga@upv.es
Salvador Cuñat	P01 UPV	salcuane@upv.es
Jorge Jiménez	P01 UPV	jjimgar1@upv.es
Fernando Boronat	P01 UPV	fboronat@dcom.upv.es
Andreas Sakellaropoulos	P02 NCSRD	asakellaropoulos@iit.demokritos.gr
Harilaos Koumaras	P02 NCSRD	koumaras@iit.demokritos.gr
Roger Briz	P03 INNO	rbriz@innovalia.org
Miguel Angel Urrutia	P03.1 CBT	maurrutia@cbt.es
Francesca Flamigni	P04 TTC	Francesca.flamigni@tttech.com
Florian Gramss	P05 SIEMENS	florian.gramss@siemens.com
José Ignacio Carretero Guarde	P06 FIWARE	joseignacio.carretero@fiware.org
Lucía Cabanillas Rodríguez	P07 TID	lucia.cabanillasrodriguez@telefonica.com
Ignacio Domínguez Martínez-Casanueva	P07 TID	ignacio.dominguezmartinez@telefonica.com
David Martínez García	P07.1 UPM	david.martinezgarci@upm.es
Fofy Setaki	P08 OTE	fsetaki@ote.gr
George Lyberopoulos	P08 OTE	glimperop@ote.gr
Elina Theodoropoulou	P08 OTE	etheodorop@ote.gr
Nikolaos Zompakis	P09 EIGHT BELLS LTD	nikolaos.zombakis@8bellsresearch.com
Ioannis Makropodis	P10 IQB	giannis.makropodis@inqbit.io
Ioannis Chouchoulis	P10 IQB	giannis.chouchoulis@inqbit.io
Vasiliki Sampazioti	P10 IQB	vasiliki.maria.sampazioti@inqbit.io
Nikos Passas	P11 FOGUS	passas@fogus.gr
Katerina Giannopoulou	P11 FOGUS	kgiannopoulou@fogus.gr
Wiesław Pawłowski	P14 SRIPAS	wieslaw.pawlowski@ibspan.waw.pl
Paweł Szmeja	P14 SRIPAS	Pawel.Szmeja@ibspan.waw.pl
Vaios Koumaras	P15 INFOLYSIS	vkoumaras@infolysis.gr
Vasilis Mavrikakis	P15 INFOLYSIS	vmavrikakis@infolysis.gr
Eduardo Garro	P16 PRODEVELOP	egarro@prodevelop.es
Álvaro Martínez-Romero	P16 PRODEVELOP	amromero@prodevelop.es

Juan Antonio Pavón	P16 PRODEVELOP	jpavon@prodevelop.es
Kyriacos Orphanide	P17 EGCTL	kyriacos.orphanides@eurogate-limassol.com
Alessandro Cassera	P17 EGCTL	alessandro.cassera@eurogate-limassol.com
Michalis Michaelides	P18 CUT	michalis.michaelides@cut.ac.cy
Herodotos Herodotou	P18 CUT	herodotos.herodotou@cut.ac.cy
Oscar López Pérez	P20 S21SEC GES	olopez@s21sec.com
Jon Egaña Zubia	P20 S21SEC GES	jegana@s21sec.com
Marcus Reutemann	P21 JD	ReutemannMarcus@JohnDeere.com
Beata Kroszka	P22 CF	bkroszka@cloudferro.com
Piotr Kłos	P22 CF	pklos@cloudferro.com
Karol Klimaszewski	P22 CF	kklimaszewski@cloudferro.com
Tomasz Mrowiec	P23 ELECTRUM	tmrowiec@electrum.pl
Bartłomiej Arendarski	P23 ELECTRUM	barendarski@electrum.pl
Danish Abbad Syed	P24 POLIMI	danishabbas.syed@polimi.it
Giulio Centi	P25 MADE	giulio.centi@made-cc.eu
Lucie Stutz	P27 SIPBB	Lucie.stutz@sipbb.ch
Sven Wallau	P27 SIPBB	Sven.wallau@sipbb.ch

History

Date	Version	Change
13-JUN-2025	0.1	ToC created and a draft version of D5.6
2-JUL-2025	0.2	First round of contributions
23-SEP-2025	0.3	Second round of contributions
10-OCT-2025	0.4	Final round of contributions and internal review process
23-OCT-2025	0.5	Incorporating feedback from the internal review and implementing improvements
31-OCT-2025	1.0	Document ready for submission

Key Data

Keywords	KPI, evaluation, validation activities, requirements, KeVIs
Lead Editor	P12 – FOGUS
Internal Reviewer(s)	UPV, TID

Table of contents

Table of contents 5
List of tables 6
List of figures 9
List of acronyms 13
Executive Summary 16
1. About this document 17
  1.1. Deliverable context 17
  1.2. The rationale behind the structure 17
  1.3. Deviation and corrective actions 18
2. Final Report on Results of Activities 19
3. Technical KPIs for aerOS 25
4. aerOS pilot KPIs 32
5. aerOS impact KPIs 38
6. Requirements coverage assessment 41
7. Final KeVI analysis 50
8. Conclusions 78
A. Appendix A. Validation Activities 80
B. Appendix B. Technical KPIs 175
  aerOS network and compute fabric 175
  aerOS Data Fabric 195
  aerOS service fabric 219
  aerOS cybersecurity and trust components 235
  aerOS self-* and monitoring 267
  aerOS decentralised AI 292
  aerOS common API 311
  aerOS management framework 324
  aerOS embedded analytics 342
  Stakeholder user satisfaction 344
C. Appendix C. Pilot and Overall KPIs 350
  Pilot 1 Data-driven cognitive production lines 350
  Pilot 2 Containerised edge computing near renewable energy sources 372
  Pilot 3 High performance computing platform for connected and cooperative mobile machinery 382
  Pilot 4 Smart edge services for the port continuum 388
  Pilot 5 Energy efficient, health safe and sustainable smart buildings 403
  Overall pilots engagement 411
D. Appendix D. Requirements Coverage 415

List of tables

Table 1: Deliverable context 17
Table 2: Activities carried out by Pilot 1 21
Table 3: Activities carried out by Pilot 2 22
Table 4: Activities carried out by Pilot 3 23
Table 5: Activities carried out by Pilot 4 23
Table 6: Activities carried out by Pilot 5 24
Table 7: Master technical KPIs table. Green coloured achieved. Red coloured not fulfilled 27
Table 8: Summary of Pilot and Overall KPIs 32
Table 9: List of aerOS Impact KPIs 38
Table 11: Summary of aerOS technical requirements per type. 41
Table 10: Summary of aerOS technical requirements per classification. 41
Table 12: Summary of aerOS technical requirements per priority. 41
Table 13: Summary of aerOS user and system requirements per type. 45
Table 14: Summary of aerOS user and system requirements per classification. 45
Table 15:Summary of aerOS user and system requirements per priority. 45
Table 16: Pilot 1.4 Reduction directly impact the CO2 production. 97
Table 17: Results connectivity test 112
Table 18: Pilot 3 Benchmark 1 114
Table 19: Pilot 3 Benchmark 2 115
Table 20: Pilot 3 Benchmark 3 116
Table 21: Confusion Matrix and classification metrics for the hydraulic anomaly detection model. 130
Table 22: Performance analysis in terms of accuracy, precision, recall, and F1-score. 131
Table 23: KPI 1.1.1 Response time for the orchestration of IoT applications (KV-1.1) 175
Table 24: KPI 1.1.2 Open-source components for aerOS to deploy and manage applications spanning the continuum (KVI-1.2) 178
Table 25: KPI 1.1.3 Usage of 5G native APIs (3GPP NEF and SEAL) (KVI-1.3) 179
Table 26: KPI 1.1.4 Usage of TSN (KVI-1.4) 181
Table 27:KPI 1.1.5 Number of old equipment units turned on actionable aerOS nodes 184
Table 28: KPI 1.1.6 Consistency of deployment compared to app blueprints 186
Table 29: KPI 1.2.1 Full support for data pipelines in all use cases, identified through requirements elicitation (KVI-5.1) 195
Table 30: KPI 1.2.2 Semantic and syntactic interoperability between all data producers and consumers in all use cases (KVI-5.2) 198
Table 31: KPI 1.2.3 Reference implementation for a data infrastructure supporting full user-control in the definition of data sources, consumers and flows (KVI-5.3) 199
Table 32: KPI 1.2.4 # of data sovereignty initiatives 202
Table 33: KPI 1.2.5 aerOS data models in open markets 203
Table 34: KPI 1.2.6 Semantic annotation support for commonly used data format 206
Table 35: KPI 1.2.7 % data sources from aerOS scenarios to be semantically annotated and exposed via Data Fabric 208
Table 36: KPI 1.2.8 Support for multiple types of data sources 209
Table 37: KPI 1.2.9 Data pipeline latency for data integration 212
Table 38: KPI 1.2.10 Simultaneous data pipeline execution 217
Table 39: KPI 1.3.1 Number of VNF/NetApps to improve performance and self-* network reconfiguration (KVI-2.3) 219
Table 40: KPI 1.3.2 Total services delivered by aerOS 223
Table 41: KPI 1.3.3 # of successful CI/CD pipelines implemented in the project 225
Table 42: KPI 1.3.4 Number of different service components running in different domains that form functional services thanks to aerOS network components 227
Table 43: KPI 1.3.5 Different types of networks managed by aerOS in pilot deployment 234

Table 44: KPI 1.4.1 Delivery of dedicated aerOS components as Open-Source SW for cybersecurity, privacy and trust (KVI-3.1) 235
Table 45: KPI 1.4.2 # scenarios with security, privacy and trust by design deployed (KVI-3.2) 237
Table 46: KPI 1.4.3 Delivery of a DevPrivSecOps cookbook and good practices manual (KVI-3.3)........ 239
Table 47: KPI 1.4.4 % of users/device/services properly authenticated 240
Table 48: KPI 1.4.5 # of parallel successfully authenticated user/devices/services........ 242
Table 49: KPI 1.4.6 % of users/device/services properly authorized........ 245
Table 50: KPI 1.4.7 # of petitions handled by the API Gateway per second 246
Table 51: KPI 1.4.8 % trusted scenarios that make use of IOTA's DLT........ 250
Table 52: KPI 1.4.9 Network overload limit due to the usage of IOTA and Tangle........ 256
Table 53: KPI 1.4.10 Trust Score Recalculation and Resource Balance........ 261
Table 54: KPI 1.5.1 Average overload time of IEs 267
Table 55: KPI 1.5.2 Number of different topologies and hardware/software combinations of IEs supported 270
Table 56: KPI 1.5.3 Number of metrics monitored from IEs........ 275
Table 57: KPI 1.5.4 Number of avoided service downgrade experience cases........ 277
Table 58: KPI 1.5.5 % of reorchestration requests issued by decentralized IE........ 282
Table 59: KPI 1.5.6 # of IoT healing scenarios covered 284
Table 60: KPI 1.5.7 % of intrusion detected by the self-security........ 286
Table 61: KPI 1.6.1 Realising decentralized AI/ML with scalability comparable to centralized approach (KVI-4.1)........ 292
Table 62: KPI 1.6.2 Energy consumption reduction due to moving AI from cloud to the edge (KVI-4.2).... 299
Table 63: KPI 1.6.3 Validation of comprehensive support, by aerOS, for distributed frugal AI components with explainability (KVI-4.3)........ 304
Table 64: KPI 1.6.4 Delivery of cookbook/good practices manual for explainable frugal AI near the edge (KVI-4.4)........ 307
Table 65: KPI 1.6.5 Decentralized frugal AI techniques available 308
Table 66: KPI 1.6.6 AI explainability techniques available........ 309
Table 67: KPI 1.7.1 % of aerOS core services exposed through OpenAPI........ 311
Table 68: KPI 1.7.2 OpenAPI UIs for documenting APIs and generating code 316
Table 69: KPI 1.7.3 Create Protocol Buffers definition for intra-orchestration module communication 318
Table 70: KPI 1.7.4 Reduce time to deploy service functions by non-technical team members using low code tool integrations........ 322
Table 71: KPI 1.8.1 # of federated domains in all aerOS continuums 324
Table 72: KPI 1.8.2 # of continuum functionalities available and operational through the Management Portal 326
Table 73: KPI 1.8.3 Performance of aerOS Federation Context Broker........ 331
Table 74: KPI 1.8.4 Federation asymptote with minimum latency (domains)........ 333
Table 75: KPI 1.8.5 Average offloading ratio of entrypoint balancing in aerOS scenarios........ 336
Table 76: KPI 1.8.6 QoE of Management Portal deployed on pilots........ 340
Table 77: KPI 1.9.1 # pre-packaged functions supported by Embedded Analytics Tool (EAT) 342
Table 78: KPI 1.9.2 # northbound wrappers designed for common operations with EAT 343
Table 79: KPI 1.10.1 Successful conduction of Open Calls (KVI-7.1)........ 344
Table 80: KPI 1.10.2 # of stakeholders deploying aerOS 346
Table 81: KPI 1.10.3 # Energy consumption & e-waste reduction in aerOS adopters 347
Table 82: KPI 2.1.1 Production process accuracy........ 350
Table 83: KPI 2.1.2 Digital service programming time 351
Table 84: KPI 2.1.3 Dimensional quality control productivity 353
Table 85: KPI 2.1.4 Accuracy of the CO2-footprint prediction (%) 355
Table 86: KPI 2.1.5 CO2-footprint measurement (% products)........ 358
Table 87: KPI 2.1.6 CO2 emissions reduction (kg) 360
Table 88: KPI 2.1.7 AGV usage........ 363
Table 89: KPI 2.1.8 AGV availability........ 365
Table 90: KPI 2.1.9 AGV travel saved/valve........ 366
Table 91: KPI 2.1.10 Definition of the calculation model 368
Table 92: KPI 2.1.11 Transparency of CO2/PCF data (minutes)........ 370

Table 93: KPI 2.2.1 Consumed renewable energy based on decision making process of aerOS 372
Table 94: KPI 2.2.2 Effectiveness of task distribution through aerOS to nodes 374
Table 95: KPI 2.2.3 Scalability of task distribution and management through aerOS 375
Table 96: KPI 2.2.4 4 CPU utilization efficiency 376
Table 97: KPI 2.2.5 Carbon awareness share of green energy 377
Table 98: KPI 2.2.6 Number of edge nodes connected in the aerOS continuum 378
Table 99: KPI 2.2.7 Number of batch processing jobs successfully distributed and executed by the system 380
Table 100: KPI 2.2.8 Precision of the Future Price prediction algorithm 381
Table 101: KPI 2.3.1 Performance and connectivity capabilities improvement (single vehicle) 382
Table 102: KPI 2.3.2 Swarm of vehicle performance improvement 384
Table 103: KPI 2.3.3 CO2 emissions reduction thanks to platooning 386
Table 104: KPI 2.4.1 Reduction of CHE idle time due to failures 388
Table 105: KPI 2.4.2 Increase on detection of equipment malfunctions (from manual to automatic) 390
Table 106: KPI 2.4.3 Increase of number of actual damaged containers (manually reported by staff vs automatic system-reports) 391
Table 107: KPI 2.4.4 Performance evaluation metrics of regression AI models (R2) 395
Table 108: KPI 2.4.5 Performance evaluation metrics of regression AI models (MAE/RMSE) for predictive maintenance of CHEs 396
Table 109: KPI 2.4.6 Performance evaluation metrics of classification AI models (accuracy) for damaged containers 398
Table 110: KPI 2.4.7 Performance evaluation metrics of classification AI models (F1) for damaged seals .. 400
Table 111: KPI 2.4.8 Number of models executed on edge nodes 402
Table 112: KPI 2.5.1 Energy use reduction 403
Table 113: KPI 2.5.2 Edge processing performance gains 404
Table 114: KPI 2.5.3 5G capabilities to execute security and privacy functions 406
Table 115: KPI 2.5.4 Service availability 406
Table 116: KPI 2.5.5 Service creation / Scalability 407
Table 117: KPI 2.5.6 Services directly managed by the aerOS orchestrator 408
Table 118: KPI 2.5.7 Improvement of air quality 409
Table 119: KPI 2.5.8 Number of AI models used/adapted for the pilot 410
Table 120: KPI 2.6.1 Validation of aerOS in different use cases (KVI-6.1) 411
Table 121: KPI 2.6.2 Enable fast-track development of new use cases through external partners (e.g., open call third parties) based on aerOS' Open Source Software components and tools from O1 (KVI-6.2) 411
Table 122: KPI 2.6.3 Identification of new application domains to deploy aerOS architecture (KVI-6.3) 412
Table 123: Technical Requirements 415
Table 124: Functional (F) and non-functional (NF) requirements of Pilot 1. 446
Table 125: Functional (F) and non-functional (NF) requirements of Pilot 2. 449
Table 126: Functional (F) and non-functional (NF) requirements of Pilot 3. 451
Table 127: Functional (F) and non-functional (NF) requirements of Pilot 4. 451
Table 128: Functional (F) and non-functional (NF) requirements of Pilot 5 454

List of figures

Figure 1: Success rate of technical KPIs per field 25
Figure 2: KPI achievement status across all Pilots and Overall KPIs 32
Figure 3. Overall technical requirements coverage (Yes, No, Partially) 42
Figure 4. Coverage per priority (Must / Should / Could / Would) 42
Figure 5. Coverage per Technical Area (Yes, No, Partially) 43
Figure 6. Coverage per requirement type (Functional – Non-functional – Constraint) 43
Figure 7. Pilot 1 requirement analysis: per achievement coverage and per priority 46
Figure 8. Pilot 2 requirement analysis: per achievement coverage and per priority 47
Figure 9. Pilot 3 requirement analysis: per achievement coverage and per priority 48
Figure 10. Pilot 4 requirement analysis: per achievement coverage and per priority 48
Figure 11. Pilot 5 requirement analysis: per achievement coverage and per priority 49
Figure 12: The four steps of the adopted KVIs methodology 50
Figure 13: KVI analysis of sub-Pilot 1.1 52
Figure 14: Technical enablers of sub-Pilot 1.1 53
Figure 15: System enablers of sub-Pilot 1.1 53
Figure 16: KVI analysis of sub-Pilot 1.2 54
Figure 17: Technical enablers of sub-Pilot 1.2 55
Figure 18: System enablers of sub-Pilot 1.2 55
Figure 19: KVI analysis of sub-Pilot 1.3 56
Figure 20: Technical enablers of sub-Pilot 1.3 57
Figure 21: System enablers of sub-Pilot 1.3 57
Figure 22: KVI analysis of sub-Pilot 1.4 58
Figure 23: Technical enablers of sub-Pilot 1.4 59
Figure 24: System enablers of sub-Pilot 1.4 59
Figure 25: KVI analysis of Pilot 2 62
Figure 26: Technical enablers of Pilot 2 63
Figure 27: System enablers of Pilot 2 64
Figure 28: KVI analysis of Pilot 3 65
Figure 29: Technical enablers of Pilot 3 67
Figure 30: System enablers of Pilot 3 68
Figure 31: KVI analysis of Pilot 4 70
Figure 32: Technical enablers of Pilot 4 71
Figure 33: System enablers of Pilot 4 72
Figure 34: KVI analysis of Pilot 5 74
Figure 35: Technical enablers of Pilot 5 75
Figure 36: System enablers of Pilot 5 76
Figure 37: Pilot 1.1. aerOS non-basic components 84
Figure 38: Pilot 1.1. Benchmarking Tool in the Management Portal (aerOS non-basic components) 84
Figure 39: Pilot 1.1. Data Catalog available in the management portal 85
Figure 40: Pilot 1.2 Setup and Configuration of a CyberSpark 87
Figure 41: Pilot 1.2 CMM CyberSpark available 87
Figure 42: Pilot 1.2 Cyberspark remote control with virtual joystick 88
Figure 43: Secondary machine remote control with virtual joystick 89
Figure 44: Pilot 1.2 HMI Digital Twin 90
Figure 45: Pilot 1.3 AGV Availability Before & after aerOS comparison 91
Figure 46: AGV Usage Before & After aerOS comparison 92
Figure 47: Order Manager App: Batching 93
Figure 48: ROS Based Navidation system 93
Figure 49: POLIMI Domain 94
Figure 50: Request to Orion-LD to retrieve entities of type Order 95
Figure 51: Pilot 1.4 Python script query all the data and save in an excel format 96
Figure 52: Pilot 1.4 Number of valves per travel from the order quantity 96

Figure 53: RackChiller Rear Door Display 101
Figure 54: Example node on Pilot 2's edge cluster 102
Figure 55: Hardware within the container 103
Figure 56: TOSCA created by manage_services.py script 105
Figure 57: TTTech Hardware build in the tractor 110
Figure 58: Pilot 3 -List and status of service pods deployed in the Kubernetes cluster of the Laptop/ECU-based aerOS Entry/Field Domain hosting any aerOS basic and additional components relevant 111
Figure 59: Pilot 3 - Entry/Field Domain details in the Management Portal 111
Figure 60: Machine fleet operation 112
Figure 61: MECSware campusXG network with Application Server 112
Figure 62: MECSware campusXG network status page 112
Figure 63: Laboratory testbed environment including all infrastructure components installed and configured for KPI evaluation of the aerOS-based Pilot 3 use case in scenarios 1 and 2. 113
Figure 64: Pilot 3 Architecture 114
Figure 65: Pilot 3 Benchmark Dashboard for Batch Size 1 115
Figure 66: Pilot 3 Benchmark Dashboard for Batch Size 3 116
Figure 67: Pilot 3 Benchmark Dashboard for Batch Size 5 117
Figure 68: Pilot 3 - aerOS Portal continuum view and deployed services 118
Figure 69: Pilot 3 Benchmark Dashboard 118
Figure 70: Pilot 3 Field Setup 119
Figure 71: Example of prescription map on John Deere Display 119
Figure 72: Pilot 3 CO2 reduction overview 120
Figure 73: Pilot 3 aerOS Field Integration Overview 120
Figure 74: STS-04 data acquisition flows. 124
Figure 75: STS-05 data acquisition flows. 125
Figure 76: Straddle Carrier SCH-167 sensors data acquisition flows. 125
Figure 77: STS-04 PLC data stored in Elasticsearch cyml.sts.plc index 126
Figure 78: STS-05 PLC data stored in Elasticsearch cyml.sts.plc index 126
Figure 79: Straddle Carrier SCH-167 sensors data stored in Elasticsearch cyml.straddle.rtu.all index 126
Figure 80: Straddle Carrier SCH-173 sensors data stored in Elasticsearch cyml.straddle.rtu.all index. 127
Figure 81: Position of the 'Pos_Limit_Switch_Spare_Gear_Land_Side' along 4 months. There were high peaks that were discarded for the model training. 127
Figure 82: 'Pos_Limit_Switch_Spare_Gear_Land_Side' variable evolution. 128
Figure 83: Joint distribution of Trolley_Piston_Cylinder_Pressure and the target variable Pos_Limit_Switch_Spare_Gear_Land_Side. 128
Figure 84: Hydraulic anomaly detection AI model deployed as an aerOS service through the management portal of the pilot. 130
Figure 85: Confusion matrices for the five ML models: ANN, DT, RF, XGBoost, and GNB 131
Figure 86: Pilot 4 domains available in the management portal. 131
Figure 87: Specific information of the EUROGATE domain in the management portal. 132
Figure 88: Running status of the anomaly detection aerOS service on the STS-04 IE. 132
Figure 89: STS-04 collected videos during October 2024. 132
Figure 90: Web application with the list of containers inferred with the CV models of aerOS. 133
Figure 91: Image and report outputs from the CV models of aerOS 133
Figure 92: Specific information of the CUT domain in the management portal 134
Figure 93: Deployment configuration of the nginx aerOS service into the CUTdomain IE. 134
Figure 94: Running status of the nginx aerOS service on the CUTdomain IE. 135
Figure 95: Pilot5 aerOS Services running 139
Figure 96: Health Score Forecast Calculation per Room 139
Figure 97: Pilot5 Login Page and Web-GUI Log for User Login 140
Figure 98: User Preferences Page: Select Preferred Seating (Room_Desk) for Option 1 (Most Preferred) ... 140
Figure 99: User Preferences Page – List of Preferred Desks Selected 140
Figure 100: Web-GUI Log for User Preferences – Desks Selected 141
Figure 101: Seating Recommendation Page 141

Figure 102: Recommender log for processing request and providing recommendations based on HI and preferences........ 141
Figure 103: Seating Recommendation Results........ 142
Figure 104: WebGUI Log for Fetching Recommendation Results from the Recommender Component........ 142
Figure 105: User Books a Desk........ 142
Figure 106: WebGUI Log for User's Desk Booking........ 143
Figure 107: Forecasting Logs for Desk Occupancy Updates (Occupancy = 1)........ 143
Figure 108: Forecasting Logs for Updating Rooms' Health Score........ 143
Figure 109: User Releases Desk........ 144
Figure 110: Web-gui Logs for Desk Release........ 144
Figure 111: Room Occupancy Updated (Occupancy = 0)........ 144
Figure 112: Forecasting Logs for health Score Updates........ 145
Figure 113: Portal View of Deployed Pilot5 Services........ 145
Figure 114: Recommender Component Deployment Status........ 146
Figure 115: Forecasting Component Deployment Status........ 146
Figure 116: Actuator Component Deployment Status........ 146
Figure 117: Daily (Day-of-Working-Week) Energy Consumption Before/After for Room 105........ 147
Figure 118:Daily (Day-of-Working-Week) Energy Consumption Before/After for Room 106........ 147
Figure 119: Daily (Day-of-Working-Week) Energy Consumption Before/After for Room 208........ 148
Figure 120: Daily (Day-of-Working-Week) Energy Consumption Before/After for Room 209........ 148
Figure 121: Visualization of the energy consumption model behaviour........ 149
Figure 122: Baseline Latency Average between two application nodes (Before aerOS) and Latency Average between Pilot nodes (with aerOS)........ 150
Figure 123: Baseline Memory Usage of IoT K8 (before aerOS)........ 150
Figure 124: Memory Usage (with aerOS)........ 150
Figure 125: Service Recovery Baseline (before aerOS) – IoT App is Down when Master Node is Down.... 150
Figure 126: Service Recovery Baseline (before aerOS) – IoT App is Running when Master Node is Down 151
Figure 127: Pilot5 master node uptime........ 151
Figure 128: Time to deploy in aerOS node in KubeEdge........ 152
Figure 129: Daily (Day-of-Working-Week) CO2 (ppm) Before/After for Room 105........ 153
Figure 130: Daily (Day-of-Working-Week) CO2 (ppm) Before/After for Room 106........ 153
Figure 131: Daily (Day-of-Working-Week) CO2 (ppm) Before/After for Room 208........ 154
Figure 132: Daily (Day-of-Working-Week) CO2 (ppm) Before/After for Room 209........ 154
Figure 133: Visualization of the PM10 prediction model behaviour........ 155
Figure 134: Visualization of the PM2.5 prediction model behaviour........ 156
Figure 135: Visualization of the PM1 prediction model behaviour........ 156
Figure 136: Visualization of the humidity prediction model behaviour........ 157
Figure 137: Visualization of CO2 prediction model behaviour........ 157
Figure 138: Visualization of the temperature prediction model behaviour........ 158
Figure 139: Registration of UE as a subscriber in the open5gs........ 159
Figure 140: UERANSIM registered gNB to open5gs........ 159
Figure 141: Registration of UE (UERANSIM provided) to open5gs........ 159
Figure 142: AMF (open5gs) logs with messages during UE (UERANSIM) registration in the 5G network. 160
Figure 143: aerOS descriptor (TOSCA based) for QoS service deployment........ 160
Figure 144: aerOS HLO proceeding to service deployment........ 161
Figure 145: QoS application (pod in aerOS domain)........ 161
Figure 146: QoS service (service in aerOS domain) exposing network port as requested........ 161
Figure 147: QoS service API exposed over aerOS........ 162
Figure 148: Calling QoS API for registering policy for UE (UERANSIM provided)........ 162
Figure 149: QoS API received request........ 163
Figure 150: Request received, and accepted, at PCF (open5gs)........ 163
Figure 151: VNFs deployment over aerOS continuum........ 164
Figure 152: aerOS Users and assigned roles, as seen from administrator cosmote user) dashboard........ 165
Figure 153: aerOS continuum administrator (“cosmote” user) has full access to all actions........ 165

Figure 154: aerOS product owner (“fsetaki” user) can create and see data products but cannot access service deployments 166
Figure 155: aerOS vertical deployer user (“vpitsilis” user) can see and create service deployments 166
Figure 156: aerOS user (“gpappas” user) can see service deployments 167
Figure 157: aerOS “cosmote” with “Continuum administrator” role embedded in token 167
Figure 158: aerOS “fsetaki” with “Data product owner” role embedded in token 168
Figure 159: aerOS “vpitsilis” with “Vertical deployer” role embedded in token 169
Figure 160: aerOS “gpappas” with “aerOS user” role embedded in token 170
Figure 161: User “gpappas” who is “aerOS user” calling GET /dataProducts and denied with 403 HTTP response code 171
Figure 162: User “gpappas” who is “aerOS user” calling DELETE /dataProducts and denied with 403 response code 171
Figure 163: User “fsetaki” who is “Product owner” calling GET /dataProducts and receiving 200 response code 172
Figure 164: Krakend logs corresponding to the previous presented Postman requests 173

List of acronyms

Acronym	Explanation	Acronym	Explanation
3GPP	3rd Generation Partnership Project	LOT	Linked Open Terms
AAA	Authentication, Authorization and Access	LOV	Linked Open Vocabularies
AGV	Automated Guided Vehicle	LSTM	Long Short-Term Memory
AI	Artificial Intelligence	LTE	Long-Term Evolution
AIC	Artificial Intelligence and Computing	LTS	Long-Term Support
AMCL	Adaptive Monte Carlo Localisation	MAC	Media Access Control
AMD	Advanced Micro Devices	MAE	Mean Average Error
ANN	Artificial Neural Network	MAPE	Mean Absolute Percentage Error
API	Application Programming Interface	MB	Megabytes
AR	Augmented Reality	ML	Machine Learning
ARM	Advanced RISC Machines	MLP	Multi-Layer Perceptron
AWS	Amazon Web Services	MQ	Message Queue
BP	Business Process	MQTT	Message Queuing Telemetry Transport
CA	Certificate Authority	MTSN	Multiport Time-Sensitive Networking
CAPIF	Common API Framework	MVP	Minimum Viable Product
CEPH	Clustered Extensible Highly-scalable Object-oriented Platform	MWh	Megawatt-hour
CHE	Container Handling Equipment	NEF	Network Exposure Function
CI/CD	Continuous Integration/Continuous Deployment	NGSI-LD	Next Generation Service Interfaces-Linked Data
CLI	Command Line Interface	OC	Open Call
CMM	Coordinate Measuring Machines	OPC-UA	Open Platform Communications-Unified Architecture
CMMS	Computerised Management System	OS	Operating System
CNI	Container Network Interface	OSS	Open-Source Software
CNN	Convolutional Neural Network	OTT	
CPU	Central Processing Unit	OWL	Ontology Web Language
CSA	Coordination and Support Actions	PC	Personal Computer
CSV	Comma-Separated Values	PCF	Product Carbon Footprint
CV	Computer Vision	PdM	Predictive Maintenance
DB	Database	PLC	Programmable Logic Controller
DHCP	Dynamic Host Configuration Protocol	PM	Particulate Matter

DLT	Distributed Ledger Technology	PoC	Proof of Concept
DMIPS	Dhrystone Million Instructions Per Second	PPM	Parts Per Million
DNS	Domain Name System	PSU	Power Supply Unit
DoA		PV	Photovoltaic
DPM	Data Product Manager	PVC	Persistent Volume Claim
DPP	Digital Product Passport	QoE	Quality of Experience
DT	Decision Tree	QoS	Quality of Service
EAT	Embedded Analytics Tool	RAM	Random Access Memory
EB	Entrypoint Balancer	RBAC	Role-Based Access Control
ECU	Electronic Control Unit	RBDMS	Relational Database Management System
EMA	Exponential Moving Average	RDF	Resource Description Framework
ERP	Enterprise Resource Planning	REST	Representational State Transfer
EU	European Union	RF	Random Forest
FaaS	Function-as-a-Service	RFID	Radio Frequency Identification
FL	Federated Learning	RISC	Reduced Instruction Set Computing
FLOPS	Floating Point Operations Per Second	RL	Reinforcement Learning
FPS	Frames Per Second	RLOPC	Remote Laboratory On-Platform Communication
FT	Feets	RMSE	Root Mean Square Error
GB	Gigabytes	RNN	Recurrent Neural Network
GFLOPS	Giga Floating Point Operations Per Second	ROS	Robot Operating System
GHz	Gigahertz	RTO	Research and Technology Organizations
GNB	Gaussian Naive Bayes	RTT	Round-Trip Time
GNU	GNU's Not Unix	SC	Service Component
GPL	GNU Public License	SCADA	Supervisory Control And Data Acquisition
GPS	Global Positioning System	SDG	Sustainable Development Goals
GPU	Graphics Processing Unit	SHAP	SHapley Additive exPlanations
GUI	Graphical User Interface	SMC	Sintered Metal Corporation
HLO	High-Level Orchestrator	SME	Small and Medium-sized Enterprises
HMI	Human-Machine Interface	SQL	Structured Query Language
HPC	High-Performance Computing	SSD	Solid State Drive
HPCP	High-Performance Computing Platform	STS	Ship-to-Shore

HTTP	HyperText Transfer Protocol	SUS	System Usability Scale
HVAC	Heating, Ventilation and Air Conditioning	SW	Software
HW	Hardware	TB	Terabytes
ID	IDentifier	TFLOPS	Tera Floating Point Operations Per Second
IdM	Identity Management	TIA	Totally Integrated Automation
IE	Infrastructure Element	TLS	Transport Layer Security
IoT	Internet Of Things	TOSCA	Topology and Orchestration Specification for Cloud Applications
IPC		TRL	Technology Readiness Level
IPTV	Internet Protocol Television	TSN	Time-Sensitive Networking
JSON	JavaScript Object Notation	UI	User Interface
JWT	JSON Web Token	UPF	User Plane Function
KB	Kilobytes	VA	Validation Activity
KPI	Key Performance Indicator	VAE	Vertical Application Enablers
KVI (or KeVI)	Key Value Indicator	VM	Virtual Machine
KWh	Kilowatt-hour	VNF	Virtual Network Function
LAN	Local Area Network	VPN	Virtual Private Network
LDAP	Lightweight Directory Access Protocol	VR	Virtual Reality
LEA	Logistics Execution Application	WG	WireGuard
LLM	Large Language Model	Wi-Fi	Wireless Fidelity
LLO	Low-Level Orchestrator	XAI	Explainable Artificial Intelligence

Executive Summary

Deliverable D5.6 "Technical Evaluation, Validation and Assessment Report (2)" will present the final analysis performed from the implementation and validation activities of aerOS within its five industrial pilots. The document consolidates the results from setup, development, and integration phases, assessing both technical and operational achievements through KPIs.

This is supported by the technical KPI evaluation, which shows an 87% completion rate, reflecting the robustness, interoperability, and scalability of the aerOS architecture with its Meta-OS capabilities. Pilot KPIs were defined in five domains: Smart Manufacturing, Renewable Energy Edge Continuum, Cooperative Mobile Machinery, Smart Ports, and Sustainable Buildings. Several achieved around 92% success, demonstrating the adaptability and efficiency of the platform in different environments. Impact KPIs are strong in communication, dissemination, standardization, and exploitation with most targets exceeded, thus validating the strategic outreach and sustainability of the project.

Moreover, the requirement coverage analysis performed confirms that 98% of the technical requirements and a very high proportion of the user and system requirements were fully or partially covered, thus ensuring full traceability to project objectives. In general, D5.6 validates aerOS as a mature, cross-domain, and value-driven Cloud-Edge-IoT continuum solution ready for large-scale adoption.

1. About this document

This deliverable summarises the final validation, evaluation, and assessment activities conducted within the aerOS project, with a focus on the technical and operational achievements derived from real-world pilot implementations. Building on previous deliverables (D5.4, and D5.5), it brings together the final experimental evidence from pilot deployments, technical KPIs, and impact indicators, providing a comprehensive overview of how aerOS fulfils its design objectives.

Sections 2 to 7 present:

- **Section 2:** Results of setup, integration, and validation across all pilots,
- **Section 3:** Evaluation of technical KPIs confirming system functionality, performance, and interoperability,
- **Section 4:** Validation of pilot KPIs measuring operational efficiency, sustainability, and scalability,
- **Section 5:** Analysis of impact KPIs on dissemination, exploitation, and standardisation,
- **Sections 6–7:** Detailed requirements coverage and final Key Value Indicator analysis.

1.1. Deliverable context

Table 1: Deliverable context

Item	Description
Objectives	The main objective of Deliverable D5.6 is to present the final results of the aerOS validation, technical evaluation, and performance assessment activities. It consolidates the evidence gathered from pilot deployments, verifying the successful integration and operation of the aerOS architecture and its components across diverse industrial and societal domains. Furthermore, it measures the achievement of technical, pilot-specific, and impact KPIs to demonstrate the effectiveness, scalability, and sustainability of the aerOS Meta-OS solution.
Work plan	The work plan followed a structured approach that built upon previous deliverables (D5.2, D5.4, and D5.5). It included: (1) completion of the technical integration and deployment activities across all pilots, (2) execution of the validation plan and KPI measurement framework, (3) analysis of requirement coverage and overall system maturity and (4) assessment of results through Key Value Indicators (KeVIs).
Milestones	There are not any specific milestones associated to the delivery of D5.6. However, it contributes to the achievement of: • MS 2 Use cases and requirements • MS 3 Components defined
Deliverables	The deliverables that are related with D5.6 are: • D5.2 - Integration, evaluation plan and KPIs definition (2) - (M18) • D5.5 - Technical evaluation, validation and assessment report (1) - (M24) • D5.4 – Use cases deployment and implementation (2) - (M37)
Risks	No risks have been identified in D5.6

1.2. The rationale behind the structure

Deliverable D5.6 is structured in such a way as to clearly and logically present first the validation and evaluation outcomes of the aerOS project in an evidence-based manner. It follows the methodological approach defined in D5.4, guaranteeing continuity in the documentation of the technical progress and related validation activities. The integration results are summarized together with the validation processes carried out in all the pilots and the final assessment of KPIs and KeVIs that will jointly prove the maturity of the aerOS ecosystem.

Each part of the deliverable matches a dimension of the evaluation process. Section 2 provides the aggregate outcome of the integration and validation work carried out across the pilots. Then, Sections 3 to 5 detail the technical, pilot, and impact KPIs analyses, respectively. Moving on, Section 6, focuses on the system requirements coverage, drawing a direct relationship between the validation results and the project objectives. Section 7 describes the expanded KeVI methodology in comparison of the analysis which was introduced in D5.5 as well as the results and the key values that reached and indetified in every pilot. Finally, Section 8 provides a summary of the the results of all these efforts and the conclusions, offering a consolidated view of the overall achievements and the final validation of the aerOS ecosystem.

1.3. Deviation and corrective actions

According to Amendment #2, all tasks under Work Package 5 were extended, including Task 5.4, to which Deliverable D5.6 belongs. As a result, D5.6 is submitted in Month 38 instead of Month 36. This extension was applied at the overall project level, providing an additional two months to maximize project impact, fully achieve and even surpass the objectives and KPIs, and ensure high-quality reporting.

2. Final Report on Results of Activities

2.1. Summary of activities reported in previous deliverables

The succeeding paragraphs summarize the pilot activities executed until M35, followed by the reporting of the validation activities in the next section 2.2

Pilot 1.1 — Green manufacturing & CO_2 footprint

The SIPBB "Lighthouse" drone line was reconfigured in early 2025 (several stations rebuilt), which affected the original asset plan and shifted effort toward integration on the updated line.

Architecture actually deployed. A Raspberry Pi (Node-RED) collects station data at the edge; a ProLiant MicroServer hosts two VMs forming a Kubernetes cluster where aerOS services run; Orion-LD receives the station data. The Edge-vs-Cloud table shows the move from manual API pulls to managed services on K8s with edge data collection.

Stations & dataflows used. 3D Printer Farm, Smart Conveyor, Quality Check, Packaging, and the SMC Air Management System were integrated into the pipeline. Figures document Node-RED ingestion, Orion-LD state, and long-term storage (Prometheus/PostgreSQL).

Dashboards & outcomes. Dashboards visualize per-order carbon footprint and energy consumption; aerOS "basic" and "non-basic" components are listed as installed for this pilot.

DPP & compliance activities. The team built a Digital Product Passport data trail: component classification (supplied vs in-house), logistics footprint estimation, packaging, and total footprint calculation for each drone. They also experimented with the Gaia-X Wizard for credential issuance/signing.

Pilot 1.2 — Automotive smart factory zero-defect metrology

Deployment path. First a PoC with Entry Domain services at NASERTIC (identity/access control + secure connectivity), then full migration to Innovalia; final deployment at AIC with multiple CMMs operating in a live metrology setup.

The **final architecture** comprises three domains: (1) Entry (Keycloak/OpenLDAP, gateway), later moved to Innovalia; (2) Innovalia Metrology domain with CMM-adjacent edge IEs; and (3) M3 Software domain for metrology operations. Communication uses the OPC-UA RobotLink Server; CMMs are linked to the edge infrastructure (no direct HLO on machines).

Standardized machine access. A dockerized RLOPC service exposes machine parameters and methods over OPC-UA for aerOS/M3/HMI—documented screens list real-time attributes (positions, speeds, probe angles, energy values).

aerOS components & Ops. Screens show the installed aerOS services per domain (Entry and Metrology) and the Management Portal views (users, domains, continuum map).

Validated capabilities. Remote configuration/operation of CMMs, Digital Twin assembly for monitoring/early deviation detection, and dynamic execution of measurement services. The Edge-vs-Cloud table documents the shift of execution/control from cloud to edge with Self* modules and OPC-UA

Pilot 1.3 — Zero Ramp-up Safe PLC Reconfiguration

Three-tier setup online. Nuremberg runs the physical line (4 AGVs, mobile robot arm, safety door) with edge processing; Munich hosts a TSN lab; a restricted cloud tier runs orchestration only, all bridged with secure VPNs. Industrial Edge services now execute vision and data collection locally on SIMATIC IPCs, in line with Siemens' data policies.

Shop-floor network and safety stack in place. Dedicated TechHall subnet with SCALANCE firewall/topology; SICK NanoScan3 scanners + Safety PLCs integrated; AGV fleet manager configured with maps/paths.

Operational low-code orchestration. A behavior-tree (BT) workflow drives orders → asset relocation → sorting → opportunistic charging; aerOS handles container lifecycle (deploy/start/stop) for skills like navigation, lift, safety, ROS–TIA bridge.

TSN lab and zero-trust connectivity. SoCe MTSN switch with IPC + Raspberry Pi nodes; OpenZiti overlay links the TSN lab and primary aerOS domain; management/federation deployed (basic + non-basic components; Docker/NATS LLOs).

Federated secondary domain. NASERTIC configured as secondary for resilience/data-residency; Federator sync via OpenZiti entrypoints.

Pilot 1.4 — AGV Swarm, Zero Break-down Logistics

Multi-domain pilot stood up. aerOS entry/management visible across the pilot continuum; network and domain infrastructure prepared (MADE dedicated network; single-node K3s + Raspberry Pi 5 at MADE/POLIMI).

Order-to-execution flow integrated. Order-manager apps (MADE & POLIMI) containerized with dual-arch images and registered in aerOS; NGSI-LD entities in Orion-LD enable order state and event sharing; a synthetic order generator supports end-to-end tests.

AGV navigation stack validated. ROS Noetic with AMCL localization, global planner, RPLIDAR, RViz monitoring; web AGV-Commander + Flask API bridge to ROS; end-to-end test from order creation to AGV mission confirmed.

Pilot 2 — Green Edge Processing

Dual-site IE deployment. Two edge nodes on-prem with Kubernetes; central management on CloudFerro. Electrum's SCADA/IoT backend is connected; PV and RDHx cooling telemetry streams into Orion-LD.

Energy-price forecasting pipeline running. tgescrapper (ingest), price_prediction (model), and data_connector (publish) are containerized and orchestrated; models retrain/roll periodically to improve accuracy.

Right-sized placement. Computationally heavy cloud-mask processing kept in the cloud to balance CPU/GPU, while price forecasting, PV, and RDHx analytics execute at the edge.

LLO improvements and security. LLO patched for nodepool IE selection; semantic annotator/translator removed in favor of Data-Fabric connectors; (Kata) runtimeClassName limitations noted for future hardening.

Pilot 3 — Cooperative Mobile Machinery

Vehicle and controller stack in place. SESAM electric tractor (JD) and R975i sprayer integrated; AutoTrac RTK guidance operational; TTControl Motion Board with Jetson AGX Xavier runs perception/control. Lab testbed is prepared for KPI runs and field validation windows (seasonal).

Pilot 4 — Smart EU Port

Predictive maintenance live. Data acquisition from Siemens S7-1500 (STS) and Omron CS1G (straddle carriers) via Siemens IoT2050 gateways (Node-RED); 4G backhaul; EUROGATE domain server aggregates and orchestrates; EntryPoint moved to AWS and linked by site-to-site VPN.

Visual inspection pipeline. Large, labelled datasets for container ID/damage/seal; dashboards for damaged-container evidence; Jetson Orin IE nodes onboarded; aerOS core services deployed at CUT and on Jetsons; inference/storage validated. Regarding **Continuum integration & security**, Port entrypoint and domains visible in the management portal; EAT and WireGuard overlay configured to traverse CG-NAT and secure east-west traffic.

Pilot 5 — Energy-Efficient, Health-Safe Smart Buildings

End-to-end smart-building loop closed. Two aerOS domains (Entry + Main) with KubeEdge at far-edge; IoT backend and Home Assistant containerized; actuator controls HVAC/air-purifier/dehumidifier from optimization targets.

AI + Optimization + Recommender integrated. Health Index and Environmental Forecasting (XGBoost) publish to Orion-LD; an Energy-Efficiency regressor complements the stack; an optimizer computes target room conditions; a rule-based desk Recommender (Docker+Helm) serves the GUI over MQTT/NGSI-LD.

Data Fabric as the backbone. 11 IoT data products defined; Forecasting, Optimizer, Recommender, and GUI are all wired through the Data Fabric/Orion-LD; real-time user flows (presence → recommendation) demonstrated.

Scenario 2 — 5G security extensions. OpenCAPIF deployed to expose NEF securely; UPF VNF onboarded in the continuum and validated with UERANSIM; RBAC enforced via LDAP/Keycloak/KrakenD.

2.2. Report of Final Validation Activities

The following tables summarize the last activities performed by each pilot and their results. All the **Validation activities** –including KPIs—and the pending **Integration activities**, reported on the previous D5.4. Together with these activities, descriptions of the pilots and their scenarios can be found. Appendix I includes descriptions of the pilots and their scenarios, and complete activity reports, divided by pilots, sub-pilots and activities.

The following table summarises the latest activities carried out by Pilot 1. This table is also subdivided into the different sub-pilots of aerOS Pilot 1. The table describes the activity code, its name and a brief summary of the results obtained.

Table 2: Activities carried out by Pilot 1

Pilot 1 - Data-Driven cognitive production lines	
Pilot 1.1 - Green manufacturing (zero net-energy) and CO2 footprint monitoring	
P1-BP1-IA13: aerOS non-basic components	All non-essential aerOS components required by the pilot have been successfully installed
P1-BP1-IA14: Integration of data analysis service for reports and statistics creation	The LLM model was tested in SIPBB systems

P1-BP1-VA19: Data quality verification	The completeness of the time series was verified, and the values were found to be within their range. The CO_2 footprint predictions were evaluated and compared with the actual values
P1-BP1-VA20: Improvement activities	Data flow has been optimized. CO_2/PCF data for specific products can also be obtained
P1-BP1-VA21: KPIs validation	Please refer to the KPIs section
P1-BP1-VA2: Qualitative validation	Please refer to the KeVI analysis
P1-BP1-VA23: Evaluation and reporting	This section refers to the final evaluation of the pilot (if any) and the reporting / documenting of our activities in D5.4 and D5.6
Pilot 1.2 - Automotive Smart Factory Zero Defect Manufacturing	
P1-BP2-VA8: Remote configuration/set-up of the CMM instrumentation robotic and kinematic configuration	The configuration has been successfully completed
P1-BP2-VA9: Remote tactile operation of CMM	The movement of multiple axes and tools connected to the gauge can be controlled remotely
P1-BP2-VA10: aerOS assist and optimize the process of Digital Twin creation	The physical gage is digitally reflected through a digital twin running on an aerOS-enabled computing continuum
P1-BP2-VA11: Dynamic execution of metrology services and Data assembling	Metrology services can now be executed in a semi-automated manner, requiring only minimal intervention from the metrologist
Pilot 1.3 - Zero Ramp-up safe PLC reconfiguration for Lot-Size-1 Production	
P1-BP3-VA1: KPI 2.1.8 validation: AGV availability > 95%	The AGVs are now able to charge autonomously
P1-BP3-VA2: KPI 2.1.7 validation: AGV usage > 80 %	The productivity of the robotic arm stations has been increased
Pilot 1.4 - AGV Swarm Zero break-down logistics for Lot-Size-1 Production	
P1-BP4-VA1: Distributed order management across MADE and POLIMI domains	aerOS has enabled more efficient order scheduling, achieving measurable reductions in unnecessary movements of automated guided vehicles (AGVs) and downtime
P1-BP4-VA2: AGV path planner and navigation system	ROS-based navigation system was validated within the POLIMI domain
P1-BP4-VA3: AI/ML-based outsourcing model	The model correctly predicted outsourcing decisions with high reliability, ensuring that POLIMI could take over orders dynamically
P1-BP4-VA4: Edge-first deployment of aerOS services	92% of deployed applications now run on edge nodes
P1-BP4-VA5: Integration with Orion-LD and inter-domain communication	The Orion-LD context broker was validated as the backbone for semantic interoperability across MADE and POLIMI
P1-BP4-VA6: KPIs 2.1.5 & 2.1.9	Please refer to the KPIs section

The table below provides a summary of the most recent activities conducted by Pilot 2, including the activity code, its name, and a brief overview of the results achieved.

Table 3: Activities carried out by Pilot 2

Pilot 2 - Data-Driven cognitive production lines	
P2-BP1-VA17: First Containerized Edge Node test	Power and temperature parameters were checked and successfully verified

P2-BP1-VA18: K8s setup and test	K8s installation was verified as a stable installation for both clusters.There were no connectivity issues beteween the nodes in each cluster
P2-BP1-VA19: Second Containerized Edge Node test	Power and temperature parameters were checked and successfully verified in the testing environment
P2-BP1-VA20: Both Containerized Edge Node run test with aerOS	The activity has been successfully validated
P2-BP1-VA21: HW installation and run test in container	The containers were installed on CF site. Second batch of hardware was mounted and connected inside
P2-BP1-VA22: Scenario 1 deployment and test	Hardware (metal containers with PV power source) and software (aerOS components) environments are prepared
P1-BP1-VA23: Scenario 1 lessons learned	Experience has been gained and lessons learned in relation to Scenario 1
P2-BP1-VA27: KPI validation (1st version)	The KPI definitions had been completed and the specifications for measuring them had also been finished
P2-BP2-VA24: Configuration Validation test	Validation tests were conducted: network connection, overall health checks
P2-BP2-VA25: Scenario 2 deployment and test	The deployment and tests have been carried out
P2-BP2-VA26: Scenario 2 lessons learned	Experience has been gained and lessons learned in relation to Scenario 2

The following table outlines the latest activities performed by Pilot 3, detailing the activity code, its name, and a concise summary of the outcomes.

Table 4: Activities carried out by Pilot 3

Pilot 3 - High Performance Computing Platform for Connected and Cooperative Mobile Machinery to improve CO2 footprint	
P3-BP1-VA1: KPI validation (Lab)	Please refer to the KPIs and Appendix I sections

The table below gives an overview of the recent activities carried out by Pilot 4, highlighting the activity code, the name, and a short summary of the results obtained.

Table 5: Activities carried out by Pilot 4

Pilot 4 - Smart edge services for the Port Continuum	
P4-BP1-VA1: Data acquisition	Different testbenches have been performed for the verification of data acquisition from the different data sources
P4-BP1-VA2: Data Storage	Two parallel NoSQL databases have been used for data storage
P4-BP1-VA3: STS and Straddle Carriers AI model inference verification	The different AI-based models have been verified on STS and Straddle Carriers real time maintenance
P4-BP1-VA4: aerOS entrypoint domain – EUROGATE domain communication	Proper communication between two of the pilot domains (from the entry point to the EUROGATE domain) has been verified
P4-BP2-VA1: Video storage	The video streams captured by the IPTV cameras are properly recorded and stored for further used as datasets on CV models training

P4-BP1-VA3: aerOS entrypoint domain – CUT domain communication	Proper communication between two of the pilot domains (from the entry point to the CUT domain) has been verified

The table that follows summarizes the recent activities of Pilot 5, listing the activity code, its name, and a brief description of the results obtained.

Table 6: Activities carried out by Pilot 5

Pilot 5 - Energy Efficient, Health Safe & Sustainable Smart Buildings	
P5-BP1-VA25: End-to-End Demonstrator (Seating Recommendation)	The validation scenario has been demonstrated through several sequences of activities
P5-BP1-VA26: Pilot Services Created, Managed and Operated by aerOS Orchestrator	This validation activity evaluates the KPI 2.5.6. Please refer to the KPIs and Appendix I sections
P5-BP1-VA27: Energy use Reduction	This validation activity evaluates the KPI 2.5.1. Please refer to the KPIs and Appendix I sections
P5-BP1-VA28: Edge Processing Performance Gains	This validation activity evaluates the KPI 2.5.2. Please refer to the KPIs and Appendix I sections
P5-BP1-VA29: Service Availability within the aerOS IE	This validation activity evaluates the KPI 2.5.4. Please refer to the KPIs and Appendix I sections
P5-BP1-VA30: Service Creation / Scalability	This validation activity evaluates the KPI 2.5.5. Please refer to the KPIs and Appendix I sections
P5-BP1-VA31: Improvement of Air Quality	This validation activity evaluates the KPI 2.5.7. Please refer to the KPIs and Appendix I sections
P5-BP2-VA1: 5G E2E deployment validation with VNFs over aerOS (UERANSIM)	A new NEF capability has been integrated and implemented through aerOS in the edge domain
P5-BP2-VA2: Access Control based on established RBAC Rules	It has been confirmed that the RBACs integrated into aerOS effectively enforce the defined policies across the entire continuum

3. Technical KPIs for aerOS

The purpose of the technical KPI dimension has been to address the existence, functionality and availability of the technical components and features defined and implemented in the aerOS platform. To analyse the technical parameters, the assessment has been built upon the results and outcomes from integration, and testing activities. To do so, the dimension was in turn split into 11 fields. The below bar chart graph shows the percentage achievement of each field withing the technical dimension.

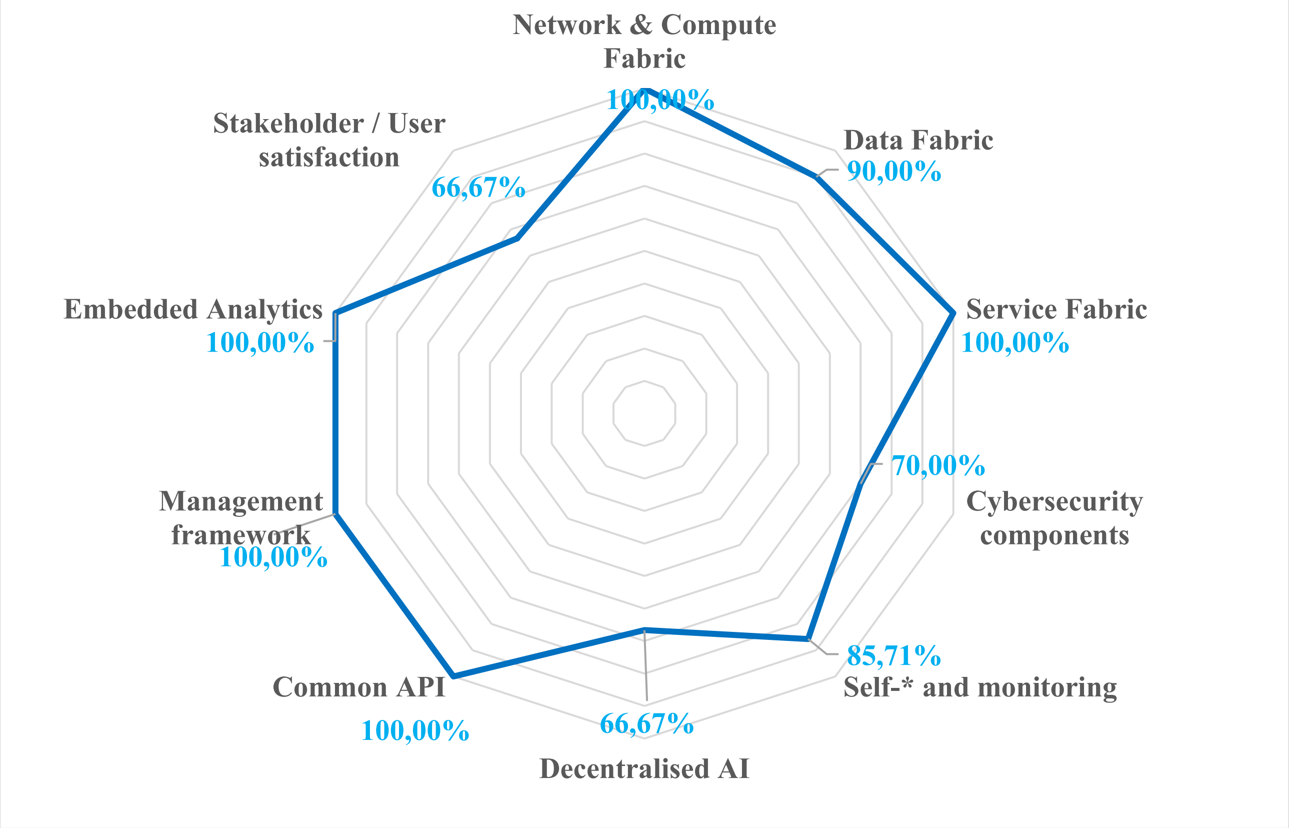


Figure 1: Success rate of technical KPIs per field

A recap of each field is as follows:

1. **aerOS network and compute fabric**: Formed by 6 KPIs, which all of them were beyond the targeted values. In particular, a response time for the orchestration of IoT applications about 4.6 seconds (KPI 1.1.1), with a 100% consistency of deployment compared to app blueprints (KPI 1.1.6) was achieved thanks to the implementation of up to 37 Open-source components (KPI 1.1.2) in order to enable aerOS to deploy and manage applications spanning the continuum. In addition, the usage of 5G native APIs in 2 scenarios (KPI 1.1.3) as well as TSN (KPI 1.1.4) in one more has guaranteed that aerOS is a valid platform for supporting the demand high levels of network determinism and reliability, as well as network awareness. Their use among the different pilots in the project was proved by deployment into 69 old equipment units that were turned on actionable aerOS nodes (KPI 1.1.5),
2. **aerOS data fabric:** 10 KPIs were set up, over which only one was not fulfilled, i.e., a 90% success rate. In detail, seven scenarios supported data pipelines (KPI 1.2.1) and semantic and syntactic interoperability (KPI 1.2.2), thanks to, among others the support of 3 semantic annotators (KPI 1.2.6), with data sources semantically annotated and exposed via Data Fabric (KPI 1.2.7). aerOS was not only focused on providing tools for the pilots of the project, but also for its holistic CEI platform, where 3 ontologies (KPI 1.2.3), 5 data sovereignty initiatives (KPI 1.2.4), 4 type of data sources (KPI 1.2.8), and 3 data models on open markets (KPI 1.2.5) were provided. It should be noted that the target value of this last KPI was originally set to 5, but it has been shown during the project lifetime that the 3 data models already implemented are more than enough for the distributed cloud-edge-iot paradigm of

aerOS. Moreover, the use of up to 15 concurrent data pipelines (KPI 1.2.10) was demonstrated, guaranteeing latencies in the range of only 0.44s (KPI 1.2.9).

3. **aerOS service fabric** field was overviewed with 5 different KPIs, all of them fulfilled. It has incorporated more than 100 aerOS services (KPI 1.3.2), including 6 VNF/NetApps to improve performance and self-* network reconfiguration (KPI 1.3.1), allowing at least 4 service components can be run in different domains although they form part of a single functional service (thanks to aerOS network components in KPI 1.3.4), and running along 7 different access network type, including LAN, WiFi, 4G, 5G, RFID, or Zigbee (KPI 1.3.5). Last, but not least, it should be noted that 7 out of these services can be deployed with the DevPrivSecOps CI/CD pipelines defined in the project (KPI 1.3.3).
4. **aerOS cybersecurity and trust components** field was also formed by other 10 KPIs, with 3 not accomplished. It included the delivery of 4 dedicated aerOS components all as open-source software (KPI 1.4.1), 3 DevPrivSecOps cookbook and good practices manual (KPI 1.4.3). Their groundbreaking performance has been proven through a 97,7% of users/device/services properly authenticated (KPI 1.4.4), with up to 150 authentication requests being handled in parallel (KPI 1.4.5), as well as 100% users/device/services properly authorized (1.4.6), with up to 15 petitions handled by the API Gateway per second (KPI 1.4.7). Unfortunately, only 4 out of the 8 pilot scenarios proved them (KPI 1.4.2), but it has a clear view of the benefits of the cybersecurity mechanisms. Regarding the trust component of the project, the use of IOTA in aerOS allowed up to 3-4 transactions per minute per pilot domain (KPI 1.4.8, although it was expected to achieve up to 5), with a minimal load increased with respect to not its use (KPI 1.4.9), and with an average latency difference below 2% (KPI 1.4.10).
5. **aerOS self-* and monitoring**: 7 KPIs were identified during the initial phases of the project to ensure the monitoring component of aerOS is warranted. All of them except one have been fulfilled. They included the support of up to 10 different topologies (KPI 1.5.2), 28 attributes potentially monitored per node (KPI 1.5.3), allowing to support the reduction of 30 % of the total running time of a node (KPI 1.5.1), and permitting any redeployment when needed (KPI 1.5.5). From the self-security side, aerOS is capable of autonomously recover at least 5 affected parts of the system (KPI 1.5.6) and detect any type of port scanning or DoS attack intrusion (KPI 1.5.7). Like previous field, some of these metrics have only been assess in 4 demonstrable scenarios (KPI 1.5.4), which goes below the initial target of 5. Nevertheless, it is considered internally among project partners as a huge success.
6. **aerOS decentralised AI** field is contemplated along 6 different KPIs, with a relative success rate of 4 out of 6. While the platform is able to realised a decentralized AI/ML with scalability comparable to centralized approach with 3 different applications (KPI 1.6.1), validated with a comprehensive support with 2 XAI and 2 frugal applications (KPI 1.6.3), plus 3 decentralized frugal AI techniques (KPI 1.6.5) and 2 explainable AI techniques (KPI 1.6.6), only 1 cookbook/good practices manual for explainable frugal AI near the edge has been delivered (KPI 1.6.4), which can be the reason of acknowledgement of project partners, so that only 15% of energy consumption has been reduced due to moving AI from cloud to the edge (KPI 1.6.2).
7. **aerOS common API** field is related to the easiness of integration inside and outside aerOS platform. To enable that approach, 4 KPIs were set, which have been achieved. In particular, 88% of aerOS core services are exposed through OpenAPI standard (KPI 1.7.1), including 5 protocol buffers (KPI 1.7.3) that enhances the efficiency and reliability of data interchange between modules. This could not have been achieved if there was not the 2 OpenAPI UIs for documenting APIs and generating code (KPI 1.7.2) that have help to decrease the time required for non-technical team members to deploy service functions within aerOS from 40s to 20s (KPI 1.7.4).
8. **aerOS management framework** was the central entrance to the users to the core aerOS platform. To guarantee it was capable of handling all the underlying services with sufficient QoE from end users and practitioners, 6 KPIs were identified and finally fulfilled. It included 13 continuum functionalities available and operational through the Management Portal (KPI 1.8.2), with 3,400 updates/s from the aerOS Federation Context Broker (KPI 1.8.3), and an average offloading ratio of entry point balancer of 50% (KPI 1.8.5), leading to 70.8 SUS score in the QoE surveys distributed along internal and external parties of aerOS. Their easiness of use has been proved by just taking a look to the connection of 47 aerOS domains in 17 different continuums (KPI 1.8.1).

9. **aerOS embedded analytics** field KPIs were mainly focused on proving the availability of the Embedded Analytics Tool (EAT). There were 2 successfully achieved KPIs, 3 pre-packaged functions supported (KPI 1.9.1) and 3 northbound wrappers designed for common operations (KPI 1.9.2).
10. Finally, although it might be considered not in their appropriate section, the **stakeholder's satisfaction** field was also considered within the technical dimension. In that sense, 2 out of 3 KPIs were identified and fulfilled. Whereas at least 14 stakeholders have deployed aerOS generating the necessary evidence to support future adoption of Meta-OS assets (KPI 1.10.2) and also tackling relevant social challenges, including energy consumption and e-waste, with a reduction of 54% in power consumption in Pilot 3 and 15% in Pilot 5 (KPI 1.10.3), the goal of embracing 80 open call applicants were undermined. Nevertheless, given that 72 applications were received, and the 15 granted open calls awardees successfully executed their goals, it must be considered that the quality of the proposals has lived up to the expectations (even surpassed it).

To sum up, it has been considered that the technical dimension KPIs assessment has been a huge success. Only 8 out of the 59 devised KPIs almost two years ago have not been achieved. From the project perspective it is considered as a proof of the work done so far, with an **87% of the technical goals fulfilled**, showing a sustainable and promising future to aerOS in the metaOS and CEI paradigm. This is further strengthened by taking a look at the 16 KPIs that were also identified back then in the proposal phase as KVIs. In that sense, 12 have been fulfilled, with a final 75% success rate for Key Values in the project.

Finally, the next table provides a complete overview of all the technical KPIs of the project, where red coloured are the ones that unfortunately have not been fulfilled. For more details, please, refer to Annex I of this report.

Table 7: Master technical KPIs table. Green coloured achieved. Red coloured not fulfilled

KPI#	KVI #	Title	Field	Target	Endline
KPI-1.1.1	KVI 1.1	Response time for the orchestration of IoT applications	1.1 - aerOS Network and compute fabric	8.5 seconds	4.8 seconds
KPI-1.1.2	KVI 1.2	Open-source components for aerOS to deploy and manage applications spanning the continuum	1.1 - aerOS Network and compute fabric	3 open-source components	37 open-source components
KPI-1.1.3	KVI 1.3	Usage of 5G native APIs (3GPP NEF and SEAL)	1.1 - aerOS Network and compute fabric	2 scenarios	2 scenarios
KPI-1.1.4	KVI 1.4	Usage of TSN	1.1 - aerOS Network and compute fabric	1 scenario	1 scenario
KPI-1.1.5		Number of old equipment units turned on actionable aerOS nodes	1.1 - aerOS Network and compute fabric	20 old equipment units	69 old equipment units
KPI-1.1.6		Consistency of deployment compared to service blueprints	1.1 - aerOS Network and compute fabric	95% consistent deployments	100% consistent deployments
KPI-1.2.1	KVI 5.1	Full support for data pipelines in all use cases (incl. open calls), identified through requirements elicitation	1.2 - aerOS Data Fabric	6 scenarios	7 scenarios
KPI-1.2.2	KVI 5.2	Semantic and syntactic interoperability between all data producers and consumers in all use cases	1.2 - aerOS Data Fabric	6 scenarios	7 scenarios

KPI#	KVI #	Title	Field	Target	Endline
KPI-1.2.3	KVI 5.3	Reference implementation for a data infrastructure supporting full user-control in the definition of data sources, consumers and flows.	1.2 - aerOS Data Fabric	3 use cases	3 use cases
KPI-1.2.4		# of data sovereignty initiatives	1.2 - aerOS Data Fabric	5 initiatives	5 initiatives
KPI-1.2.5		aerOS data models in open markets	1.2 - aerOS Data Fabric	5 data models	3 data models
KPI-1.2.6		Semantic annotation support for commonly used data format	1.2 - aerOS Data Fabric	3 semantic components	3 semantic components
KPI-1.2.7		% data sources from aerOS scenarios to be semantically annotated and exposed via Data Fabric	1.2 - aerOS Data Fabric	50% scenarios	50% scenarios
KPI-1.2.8		Support for multiple types of data sources	1.2 - aerOS Data Fabric	3 types of data sources	4 types of data sources
KPI-1.2.9		Data pipeline latency for data integration	1.2 - aerOS Data Fabric	1 second	0.44 seconds
KPI-1.2.10		Simultaneous data pipeline execution	1.2 - aerOS Data Fabric	5 data pipelines	15 data pipelines
KPI-1.3.1	KVI 2.3	Number of VNF/NetApps to improve performance and self-* network reconfiguration	1.3 - aerOS Service fabric	6 services with NetApps	6 services with NetApps
KPI-1.3.2		Total services delivered by aerOS	1.3 - aerOS Service fabric	50 aerOS services	100 aerOS services
KPI-1.3.3		# of successful CI/CD pipelines implemented in the project	1.3 - aerOS Service fabric	4 CI/CD pipelines	7 CI/CD pipelines
KPI-1.3.4		Number of different service components running in different domains that form functional services thanks to aerOS network components	1.3 - aerOS Service fabric	4 service components	4 service components
KPI-1.3.5		Different types of networks managed by aerOS in pilot deployment	1.3 - aerOS Service fabric	2 network accesses	7 network accesses
KPI-1.4.1	KVI 3.1	Delivery of dedicated aerOS components as Open Source SW for cybersecurity, privacy and trust	1.4 - aerOS cyber security components	100% OSS cybersecurity services	100% OSS cybersecurity services
KPI-1.4.2	KVI 3.2	# scenarios with security, privacy and trust by design deployed	1.4 - aerOS cyber security components	4 pilots	4 pilots
KPI-1.4.3	KVI 3.3	Delivery of a DevPrivSecOps cookbook and good practices manual	1.4 - aerOS cyber security components	3 cookbooks	3 cookbooks

KPI#	KVI #	**Title**	**Field**	**Target**	**Endline**
KPI-1.4.4		% of users/device/services properly authenticated	1.4 - aerOS cyber security components	95% authenticated users	97.7% authenticated users
KPI-1.4.5		# of parallel successfully authenticated user/devices/services	1.4 - aerOS cyber security components	150 parallel users	150 parallel users
KPI-1.4.6		% of users/device/services properly authorized	1.4 - aerOS cyber security components	95% authorized users	100% authorized users
KPI-1.4.7		# of petitions handled by the API Gateway per second	1.4 - aerOS cyber security components	15 petitions / second	14 petitions / second
KPI-1.4.8		% trusted scenarios that make use of IOTA's DLT	1.4 - aerOS cyber security components	5 data transactions / minute	4 data transactions / minute
KPI-1.4.9		Network overload limit due to the usage of IOTA and Tangle	1.4 - aerOS cyber security components	30% network load	30% network load
KPI-1.4.10		Trust Score Recalculation and Resource Balance	1.4 - aerOS cyber security components	30% increase	2% increase
KPI-1.5.1		Average overload time of IEs	1.5 - aerOS self-* and monitoring	20% reduction	30% reduction
KPI-1.5.2		Number of different topologies and hardware/software combinations of IEs supported	1.5 - aerOS self-* and monitoring	10 topologies	10 topologies
KPI-1.5.3		# of metrics monitored from IEs	1.5 - aerOS self-* and monitoring	15 attributes	28 attributes
KPI-1.5.4		# of avoided service downgrade experience cases	1.5 - aerOS self-* and monitoring	5 scenarios	4 scenarios
KPI-1.5.5		% of reorchestration requests issued by decentralized IEs	1.5 - aerOS self-* and monitoring	25% reorchestration	100% reorchestration
KPI-1.5.6		# of IoT healing scenarios covered	1.5 - aerOS self-* and monitoring	5 scenarios	5 scenarios
KPI-1.5.7		% of intrusion detected by the self-security	1.5 - aerOS self-* and monitoring	90% intrusions detected	100% intrusions detected
KPI-1.6.1	KVI 4.1	Realizing decentralized AI/ML with scalability comparable to centralized approach.	1.6 - aerOS decentralized AI	3 decentralized applications	3 decentralized applications
KPI-1.6.2	KVI 4.2	Energy consumption reduction due to moving AI from cloud to the edge	1.6 - aerOS decentralized AI	50% energy reduction	35% energy reduction
KPI-1.6.3	KVI 4.3	Validation of comprehensive support, by aerOS, for distributed frugal AI components with explainability	1.6 - aerOS decentralized AI	2 frugal AI + 2 XAI apps	2 frugal AI + 2 XAI apps
KPI-1.6.4	KVI 4.4	Delivery of a cookbook/good practices	1.6 - aerOS decentralized AI	3 cookbooks	1 cookbook

KPI#	KVI #	Title	Field	Target	Endline
		manual for explainable frugal AI near the edge.			
KPI-1.6.5		Decentralized frugal AI use cases for application	1.6 - aerOS decentralized AI	3 frugal AI techniques	3 frugal AI techniques
KPI-1.6.6		AI explainability techniques available	1.6 - aerOS decentralized AI	2 XAI techniques	2 XAI techniques
KPI-1.7.1		% of aerOS core services exposed through Open API	1.7 - aerOS common API	50% aerOS exposed	88% aerOS exposed
KPI-1.7.2		OpenAPI UIs for documenting APIs and generating code	1.7 - aerOS common API	2 OpenAPI Uis	2 OpenAPI Uis
KPI-1.7.3		Create Protocol Buffers definition for intra-orchestration module communication	1.7 - aerOS common API	3 Protocol buffers	5 Protocol buffers
KPI-1.7.4		Reduce time to deploy service functions by non-technical team members using low code tool integrations	1.7 - aerOS common API	40 seconds	20 seconds
KPI-1.8.1		# of federated domains in all aerOS continuums	1.8 - aerOS management framework	15 domains / 8 continuums	47 domains / 17 continuums
KPI-1.8.2		# of continuum functionalities available and operational through the management portal	1.8 - aerOS management framework	10 functionalities	13 functionalities
KPI-1.8.3		Performance of aerOS Federation Context Broker	1.8 - aerOS management framework	2500 updates / second	3400 updates / second
KPI-1.8.4		Federation asymptote with minimum latency (domains)	1.8 - aerOS management framework	4 domains	24 domains
KPI-1.8.5		Average offloading ratio of entrypoint balancing in aerOS scenarios	1.8 - aerOS management framework	30% offloading	50% offloading
KPI-1.8.6		QoE of management portal deployed on pilots	1.8 - aerOS management framework	68 SUS score	70.8 SUS score
KPI-1.9.1		# pre-packaged functions supported by Embedded Analytics Tool (EAT)	1.9 - aerOS Embedded analytics	3 functions	3 functions
KPI-1.9.2		# northbound wrappers designed for common operations with EAT	1.9 - aerOS Embedded analytics	3 wrappers	3 wrappers
KPI-1.10.1	KVI 7.1	Successful conduction of Open Calls	1.10 - Stakeholder/user satisfaction/OpenCall	80 applicants	72 applicants
KPI-1.10.2		# of stakeholders deploying aerOS	1.10 - Stakeholder/user satisfaction/OpenCall	5 stakeholders	14 stakeholders

KPI#	KVI #	**Title**	**Field**	**Target**	**Endline**
KPI-1.10.3		# Energy consumption & e-waste reduction in aerOS adopters	1.10 - Stakeholder/user satisfaction/OpenCall	2% energy / e-waste reduction	Pilot 3: 54% power reduction Pilot 5: 15% power reduction

4. aerOS pilot KPIs

Section 4 focuses on the Pilot KPIs. A total of 38 KPIs were defined across all five Pilots, and almost all of them successfully reached their target values. The validation methods and the corresponding outcome ellaboration for each Pilot are described in Section 2. Also, three more overall KPIs were defined and achieved. Furthermore, detailed tables presenting each KPI, its description, requirements, measured values, and target values, are included in Appendix C.

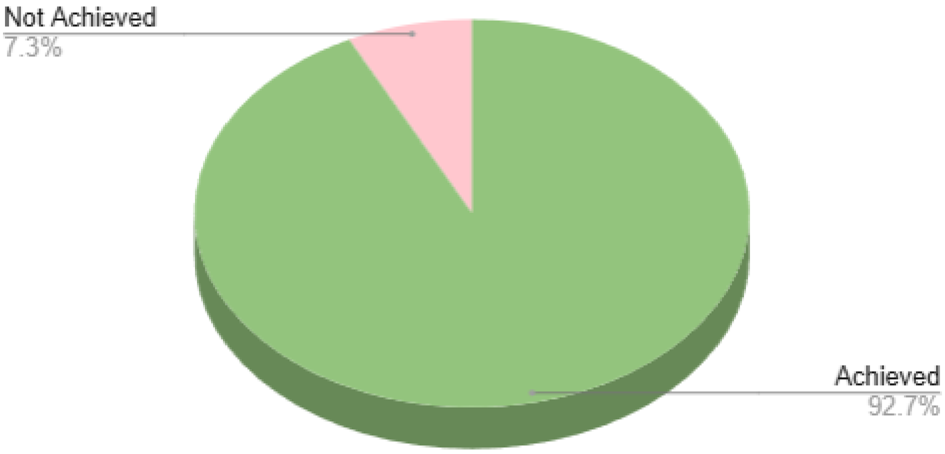


Figure 2: KPI achievement status across all Pilots and Overall KPIs

The aerOS project defined a total of 38 KPIs across five pilots. Each pilot focused on various technological and operational dimensions of the aerOS platform. Pilot 1 ("Smart Manufacturing") had 11 KPIs, of which 8 achieved or exceeded their targets, showing significant gains in process automation, quality control, and CO_2 transparency, while 3 remained slightly below target. Pilot 2 ("Energy and Edge Continuum") was monitoring 8 KPIs, all of which have achieved or surpassed their targets, confirming strong results regarding renewable energy integration, task distribution efficiency, and edge scalability. Pilot 3 ("High-performance computing for connected and cooperative mobile machinery") comprised 3 KPIs, of which two were achieved and one was partly achieved, showing major improvements in vehicle computing and network capabilities as well as measurable CO_2 reduction through platooning. Pilot 4 ("Smart Ports") included 8 KPIs, which were fully achieved or overachieved, reflecting notable progress in AI-based predictive maintenance, operational reliability, and edge deployment. Pilot 5 ("Energy-efficient, health-safe, and sustainable smart buildings") had 8 KPIs (100 % achievement in all indicators), confirming the improvements in energy efficiency, air quality, and AI model deployment. Last but not least, three cross-pilot KPIs (2.6.1–2.6.3) validated the scalability of the platform, its readiness for open source, and the cross-domain applicability; all these reached or surpassed their target. In general, the aerOS KPIs confirm a high level of success across pilots, with about 90 % of all KPIs fully achieved or exceeded, showing the maturity, performance, and impact of the platform within diverse industrial domains.

Table 8: Summary of Pilot and Overall KPIs

KPI#	**Title**	**Baseline**	**Target**	**Final measured value**
KPI-2.1.1	Production process accuracy	Dependent on product GD&T complexity	10% increase	9.2% increase (92%)
KPI-2.1.2	Digital service programming time	2 weeks	2 days	2.3 days

KPI#	Title	Baseline	Target	Final measured value
KPI-2.1.3	Dimensional quality control productivity	3 parts/hour (depending on GD&T complexity)	5 parts/hour	5.06 parts/hour (101%)
KPI-2.1.4	Accuracy of the CO2-footprint prediction (%)	N/A	>80%	83.4% - drone type A 90.4% - drone type B
KPI-2.1.5	CO2-footprint measurement (% products)	N/A	10% - 100%	100%
KPI-2.1.6	CO2 emissions reduction (kg)	0%	<20%	39.42%
KPI-2.1.7	AGV usage	54%	>80%	80%
KPI-2.1.8	AGV availability	69 %	>95%	96 %
KPI-2.1.9	AGV travel saved/valve	0% (1 Travel per Valve)	<20%	39.42%
KPI-2.1.10	Definition of the calculation model	120 minutes	> 30%-time reduction	< 1 second (>99,99% reduction)
KPI-2.1.11	Transparency of CO2/PCF data (minutes)	N/A	< 2 minutes	~0.025 seconds
KPI-2.2.1	Consumed renewable energy based on decision making process of aerOS	0 MWh/month	20 MWh/month	19391.54 MWh (97%)
KPI-2.2.2	Effectiveness of task distribution through aerOS to nodes	N/A	99.5% of tasks executed on schedule	100%
KPI-2.2.3	Scalability of task distribution and management through aerOS	N/A	10k tasks executed/month	up to 250k jobs/month
KPI-2.2.4	CPU utilization efficiency	0%	80%	Average 84%
KPI-2.2.5	Carbon awareness share of green energy	0%	60%	100%
KPI-2.2.6	Number of edge nodes connected in the aerOS continuum	0	2	2
KPI-2.2.7	Number of batch processing jobs successfully distributed and executed by the system	0	300k	475.718
KPI-2.2.8	Precision of the Future Price prediction algorithm	0%	85%	85-90%
KPI-2.3.1	Performance and connectivity capabilities improvement (single vehicle)	For performance: GPU: 2x128 GFLOPS FP 16 CPU: 26000 DMIPS. For connectivity: No network available.	For performance: GPU: 12.6 FP16 TFLOPS; CPU: SPEC int 2k6: 22, SPEC int rate: 140 Gflops. For connectivity: 4G/5G network available.	For performance: GPU: 12.6 FP16 TFLOPS; CPU: SPEC int 2k6: 22, SPEC int rate: 140 Gflops. For connectivity: 4G / 5G network available
KPI-2.3.2	Swarm of vehicle performance improvement	The baseline frame rate of 4 frames per second (FPS) per	6 FPS pro Camera and 18 km/h	During the lab and field testing it was proved to increase the FPS to 6.25

KPI#	Title	Baseline	Target	Final measured value
		camera represents the current processing capacity for the exemplary task in the use case.		by ensuring the field operating speed of 20km/h.
KPI-2.3.3	CO2 emissions reduction thanks to platooning	89,31 kg CO2/ha (33,7 l Diesel/ha)	A reduction of 80% - 17,9 kg CO2/ha	Thanks to the implementation of the aerOS components we could measure the following results for 40% CO2 reduction for diesel and electric tractors in a swarm environment.
KPI-2.4.1	Reduction of CHE idle time due to failures	Total 2023 downtime of 4 straddle Carriers: 900h Total 2023 downtime of 2 STS: 297.70h	20-30%	Q1-Q2 2025 downtime of 4 Straddle Carriers: 403h Q3 2025 downtime of 2 STS: 46 hours Straddle carriers 21.5% less downtime hours STS 31.3% less downtime hours Average 26.4% less downtime hours
KPI-2.4.2	Increase on detection of equipment malfunctions (from manual to automatic)	In 2023: 30 unplanned failures detected (manual), 0 predictive (automatic)	30-40% with respect to 2023	Q1-Q2 2025: 20 unplanned failures detected (manual), 8 predictive (automatic) = 28 detected → +86% detections
KPI-2.4.3	Increase of number of actual damaged containers (manually reported by staff vs automatic system-reports)	350 damaged containers reported by terminal staff + 30 damaged containers not reported and claimed	30-40%	Q1-Q3 2025:396 damaged containers reported by terminal staff + 37 damaged containers not reported and claimed (not using CV models). Oct 17-24, 2025: 11 damaged containers reported by terminal staff. 60 damaged containers reported by CV. Increase: 445%
KPI-2.4.4	Performance evaluation metrics of regression AI models (R2)	N/A	0.8	83.3% (110%)
KPI-2.4.5	Performance evaluation metrics of regression AI models (MAE/RMSE) for predictive maintenance of CHEs	N/A	20%	F1-score: 98.0%

KPI#	Title	Baseline	Target	Final measured value
KPI-2.4.6	Performance evaluation metrics of classification AI models (accuracy) for damaged containers	N/A	60%	mAP50: 75%
KPI-2.4.7	Performance evaluation metrics of classification AI models (F1) for damaged seals	N/A	60%	mAP50: 86%
KPI-2.4.8	Number of models executed on edge nodes	N/A	5	10 (200%)
KPI-2.5.1	Energy use reduction	Baseline measurements vary per room, but some indicative consumptions to be reported without the aerOS optimisation range from 40Kwh - 150Kwh.	20% reduction of the daily baseline consumption.	100%
KPI-2.5.2	Edge processing performance gains	The pilot is implemented on premises and dedicated networks already and typical values monitored include: Latency: 2-3 ms Memory: 1.5 Gbps.	The measurement of the Edge processing performance gains is a composite KPI that can be approximated by collecting the following sub-KPIs 1. Exhibit average E2E Communication Latency < 100 ms for the aerOS nodes deployed locally (in the edge), measured through ping tools. 2. Demonstrate the gains of KubeEdge vs. K8 deployments utilising light devices at the far edge gaining 20 % less memory resources consumption	100% Latency of communication between the pilot5 aerOS nodes (ms): Average: 0.919 ms Memory utilization when deploying IoT Application in a KubeEdge node: 730 Mbytes Time to recover IoT application when master node is down

KPI#	Title	Baseline	Target	Final measured value
			comparing the cluster reported average measurement values. 3. Demonstrate the gains of Kube Edge for service resilience, measuring the service recovery time under various disruptive conditions showcasing 90% increase in recovery time (Kubeedge vs. K8)	
KPI-2.5.3	5G capabilities to execute security and privacy functions	0	2	100% 2 5G VNFs deployed over aerOS
KPI-2.5.4	Service availability	Manual operation	99.99% in the service window of operations	100% 99,9999% in the service window a period of one month for at least one pilot node. Uptime: 25 days in the service window of 1 month
KPI-2.5.5	Service creation / scalability	Mannual	< 10 min end-to-end	100% Time-to-deploy: 34 secs
KPI-2.5.6	Services directly managed by the aerOS orchestrator	0	3	Exhibit the operation of 3 pilot5 services in the aerOS-capable infrastructure (K8s/KubeEdge) (100%)
KPI-2.5.7	Improvement of air quality	Relative value per room. Spike values in the range of 1200-1500 ppm are measured.	A typical acceptable target is set to be 400-600 ppm per room for the demo, average > 20% improvement. Especially for the rooms of the pilot, and the specific demo situation, the target is set to me to reduce the max CO2 lower than 1000 ppm in all cases.	100% For all rooms of the pilot, max CO2 is less than 1000 ppm at all times. Significant improvement in air quality with the deployment of the pilot.

KPI#	Title	Baseline	Target	Final measured value
KPI-2.5.8	Number of AI models used/adapted for the pilot	No AI used	6 models	7 (110%)
KPI-2.6.1	Validation of aerOS in different use cases	0	>5	5
KPI-2.6.2	Enable fast-track development of new use cases through external partners (e.g., open call third parties) based on aerOS' Open-Source Software components and tools from O1	0	14	15
KPI-2.6.3	Identification of new application domains to deploy aerOS architecture	N/A	3	10

Pilot 1 has demonstrated how the aerOS MetaOS turns four very different manufacturing scenarios into a single, data-driven continuum -linking machines, people, and sites through common orchestration, real-time data, and built-in trust. In sub-pilot 1.1, aerOS enabled full CO_2 transparency and automation for a highly customizable drone line: footprint prediction exceeded targets for the 2 main product types, coverage reached 100% of products, model setup time fell from 120 min to under a second, and access to CO_2/PCF data became effectively real-time. Sub-pilot 1.2 shifted metrology from manual, on-site workflows to remote, autonomous operation at the edge; self-recovery kept probe/scan pipelines aligned (driving a 9.2% accuracy gain), setup time dropped from ~10 days to just over two, and throughput surpassed 5 parts/hour. Moving on, to sub-pilot 1.3 showed how reusable "skills" deployed via aerOS raise intralogistics performance: AGV availability reached 96% and usage rose to 84%, as vehicles took on new tasks (including repositioning robot-arm workstations) without major hardware changes. Finnaly, sub-pilot 1.4 extended optimization across sites: smarter order grouping and route planning cut AGV trips by ~39% and reduced CO_2 emissions by ~39.4% for the validated flow.

Pilot 2 defined a total of 8 KPIs to assess the impact of the aerOS Meta-OS on shifting computation of workloads into edge nodes and optimizing the use of renewable (photovoltaic) energy source, in changing green-energy availability conditions. By the final evaluation period (M36-M38), seven out of eight KPIs had been achieved. KPI 2.2.1 which measured total amount of renewable energy consumed on monthly basis reached 19391.54MWh (30 days) and it is 97% of the target value. KPI 2.2.2 showing the share of scheduled tasks completed on time achieved 100%. KPI 2.2.3 focused on scalability of task distribution and management through aerOS reached up to 250k jobs per month. KPI 2.2.4 measuring average CPU consumption by worker nodes reached 84% and we saw the strong correlation between KPI 2.2.1 and 2.2.4 – higher CPU usage caused higher power consumption. KPI 2.2.5 which measured green energy share for jobs was exceeded with 100% of energy used was coming from green energy sources. KPI 2.2.6 measuring number of edge nodes connected in the aerOS continuum was achieved with 2 pilot's edge nodes on remote physical location deployed. KPI 2.2.7 focused on number of batch jobs scheduled, orchestrated and executed by aerOS continuum was achieved with the count 475 718 as of 2025-10-14. KPI 2.2.8 measuring precision of Future Energy Price Prediction algorithm was reached with overall accuracy of algorithm of ~88%. The most of out KPIs focus on data processing energy consumption and its efficiency. What is important to see that pilot consumes less energy when there is low processing demand (k8s autoscaling mechanism is shutting down compute nodes). However, the high values of KPIs 2.2.1 & 2.2.4 forced us to maximise CPU and energy consumption via the very high processing demand for one of our scenarios.

Pilot 3 focused on "High performance computing platform for connected and cooperative mobile machinery" and tracked 3 KPIs (KPI 2.3.1 to KPI 2.3.3). All three KPIs achieved or exceeded their target values by the final measurement period M38. The primary objectives of these KPIs were Performance and Connectivity Enhancement, AI-driven Efficiency, and Sustainability through CO_2 Reduction. More specifically, KPI 2.3.1 demonstrated that the integration of the aerOS platform enabled the achievement of the target computational performance (GPU: 12.6 FP16 TFLOPS and CPU: SPEC int rate 140 Gflops) while establishing reliable 4G/5G connectivity in rural environments. KPI 2.3.2 validated swarm-level performance improvements through AI-supported applications, with the frame rate increasing from 4 FPS to 6.25 FPS per camera and the field operation speed reaching 20 km/h, surpassing the 20% improvement target. Finally, KPI 2.3.3 confirmed a 40% reduction in CO_2 emissions thanks to the adoption of platooning and coordinated operations among electric tractors. The evaluation period covered laboratory and field testing phases between M24 and M38, confirming that aerOS significantly improved computing performance, connectivity reliability, and environmental sustainability within agricultural and construction machinery operations.

Pilot 4 focused on " Smart edge services for the port continuum " where 8 KPIs (KPI 2.4.1 to KPI 2.4.8) were identified at the first half of the project. From all of them, whereas 8 KPIs have achieved their target values. The primary objectives of these KPIs were Operational efficiency and personnel safety. More specifically, the reduction of CHE idle time due to failures by 26% (KPI 2.4.1) as well as the increase on detection of equipment malfunctions (from manual to automatic) by 86% (KPI 2.4.2) have shown how the predictive maintenance on the edge provides a relevant benefit to EUROGATE operational efficiency. This was endorsed with the achievement of KPI 2.4.4 and KPI 2.4.5 related to the performance evaluation of the specific AI models in terms of precision (83.3%) and F1 (98%). On the other hand, despite the fulfilment of KPI 2.4.6 and KPI 2.4.7 proved from a performance evaluation that the developed AI models regarding the accuracy of detected damaged containers (with an mAP50 of 75%), and detected wrongly sealed containers (with an mAP50 of 86%), respectively, the final outcome in terms of business benefits related to the increase of number of actual damaged containers manually reported by staff vs automatic system-reports (KPI 2.4.3) . Finally, last Pilot 4 KPI 4.2.8 has proved the frugality and lightweight services envisioned in aerOS can be achieved, as the IEs that are being used in Port Continuum pilot have low processing capabilities, but are able to support up to 10 AI models being executed on them. The validation activities about how these KPIs were achieved are provided in various subsections of D5.4 and in the annex of this report.

Pilot 5 focused on "Energy efficient, health safe and sustainable smart buildings" and it tracked 8 KPIs (KPI 2.5.1 to KPI 2.5.8). All 8 KPIs achieved their target values, 4 of them completed by the final measurement period M38 and the rest had been already completed by M24. The primary objectives of these KPIs were Sustainability and Health, Edge Computing and Performance, and Service Reliability and Scalability. More specifically, the reduction of energy use by 20% through frugal AI (KPI 2.5.1) improves sustainability and achieves a significant improvement of air quality - max CO2 less than 1000 ppm- (KPI 2.5.7) for health safety. This health and energy optimization was supported by the development of 7 AI models (KPI 2.5.8) for forecasting and prediction. Edge Computing and Performance can be addressed through Edge processing performance gains (KPI 2.5.2). Additionally, it validated the use of 5G capabilities by deploying 3 VNFs (Virtual Network Functions) (KPI 2.5.3) over aerOS for security and privacy functions. Last but not least, is Service Reliability and Scalability. Ensuring high service availability (KPI 2.5.4), rapid service creation/scalability (KPI 2.5.5), and confirming the management of at least 3 services/workloads (KPI 2.5.6) directly by the aerOS orchestrator. The validation activities about how these KPIs were achieved are provided in various subsections of the D5.6, such as P5-BP1-VA28 Edge Processing Performance Gains, P5-BP1-VA29 Service Availability, and P5-BP2-VA1 5G E2E deployment validation as well as in .

5. aerOS impact KPIs

In this section, the impact KPIs are presented along with their final values in the context of communication, dissemination, standardization, and exploitation and business models. Further details regarding the validation of these KPIs are provided in D6.3.

Table 9: List of aerOS Impact KPIs

Field	KPI id	Name	Target	M24	M38
	KPI.3.1.1	# of Website unique visitors / page views	4000/10000	5,115/20,505	7.147/20.706

3.1 Communication	KPI.3.1.2	# of aerOS posts in social networks/ #of newsletters issued	1000/12	736/7	1491/11 (& 1 under editing)
	KPI.3.1.3	# of aerOS social-media community members across all-sites	1000	1.018	2096
	KPI.3.1.4	# of videos delivered about aerOS technical and global advances / webinars-workshops organised	20 / 6	11/15	48/35
	KPI.3.1.5	# of interviews/articles/press releases with external relevant dissemination targets	30	19	30
	KPI.3.1.6	# of liaison with other projects of the cluster including CSA events	35 actions	40	>60
3.2 Dissemination	KPI.3.2.1	# of scientific papers published in conferences / Q1-Q2 journals	20 / 8	5/11	23/24
	KPI.3.2.2	# of activities towards Education institutions (courses, lectures, PhDs)	15	4	18
	KPI.3.2.3	# of presentations and other activities in events/conferences/fairs by aerOS partners	35	39	77
	KPI.3.2.4	# of workshops organised / average participants in each workshop	3 / 60	10/20	18/30
	KPI.3.2.5	# of PhD and MSc theses started about aerOS	6	10	11
3.3 Standardisation	KPI.3.3.1	Contributions to standardisation bodies	12	18	12
	KPI.3.3.2	Exploitation to entry-points into standardisation bodies	25	15	25
	KPI.3.3.3	aerOS contributions to European pre-normatives	3	2	3

	KPI.3.3.4	aerOS contributions to data-related clusters and initiatives	10	1	10
3.4 Exploitation and business models	KPI.3.3.5	# of contributions to relevant data spaces (GAIA-X, IDSA)	10	2	10
	KPI 3.4.1	Contribution to OSS projects	12	11	15
	KPI.3.4.2	Business plans for exploitable assets, stakeholders and key alliances identified and contacted	100%	0	100%
	KPI 3.4.3	New business lines on aerOS by partners	2	0	2
	KPI 3.4.4	# of startups adopting aerOS results as technological baseline for business	1	0	11
	KP 3.4.5	# of tech-transfer contracts signed based on aerOS (from Universities/RTOs)	1	0	(0) In progress
	KPI 3.4.6	Private investments in aerOS and related open technologies	10 M	0	(0)In progress
	KPI 3.4.7	Market share in edge-cloud-computing of Europe vs world	32,00%	0	~24%

6. Requirements coverage assessment

During the first part of the project, in the context of WP2, an exercise was done to define the requirements of the project. These requirements were collected, first in deliverable D2.2 and, later, the final list was released within deliverable D2.3, submitted back in February 2024.

As part of the duties of WP5 in the last part of the project (task T5.3), the team of aerOS has performed an assessment of the coverage of those requirements once the solution has been finalized and tested in all 5 pilots of the project.

This section includes a short summary reflecting on the results of this analysis, while in Appendix D the complete list of requirements can be found. There, a full relation of technical and pilot-related requirements is provided, as per how they were expressed in D2.3 (and D2.2, before that) but adding two more columns: *"Accomplishment degree"* and *"Evidences"*. This material has been provided in order to allow the reader to check that the coverage is well reported and referenced in technical or pilot-related deliverables. Therefore, an effort of reflection has been performed, and is analysed below.

6.1. Analysis of technical requirements

A number of **102 technical requirements** were recorded following the methodology described in deliverable D2.2 (M9 – May 2023). After a refined revision of the 66 initial requirements in content and scope, 36 new requirements were identified and described in the period M9-M18 (up to February-2024). Since those were categorized and described following a clear methodology, find below a very shot summary and numbers related to those:

Table 11: Summary of aerOS technical requirements per type.

Requirement type	Quantity
Data	24
Infrastructure	16
AI	17
Security	11
Meta-OS	11
Application	3
Development	6
Services	3
Network	6

Table 10: Summary of aerOS technical requirements per classification.

Requirement classification	Quantity
Functional	26
Non-functional	72
Constraints for design	4

Table 12: Summary of aerOS technical requirements per priority.

Requirement priority	Quantity
Must	71
Should	32
Could	2

In the previous tables, some statistics are shown regarding the recorded requirements; presented **based on the area they refer to, their type and priority**. Please, note that the previous is only a sub-set of the categorization existing (e.g., also per role, per domain…). As it can be seen, the majority of requirements gathered were Non-

Functional ones, prioritized as Must-have and mostly referring to Data, Infrastructure, AI, Security and Meta-**OS areas of the aerOS project.**

Bearing that in mind, below there is presented **an analysis of how those have been covered at the end of the project (M38 – October 2025)**. In particular, the same differentiation (area, type and priority) has been conducted:

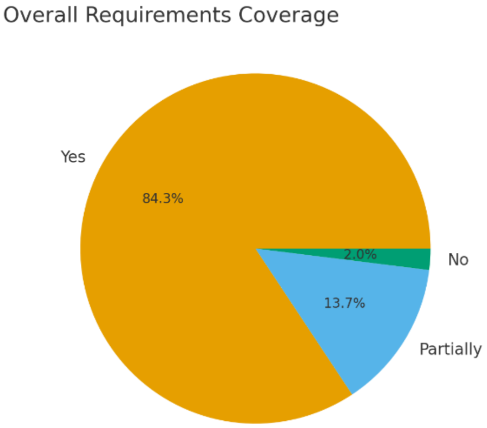


Figure 3. Overall technical requirements coverage (Yes, No, Partially)

As it can be seen, the aerOS technical alignment and accuracy with requirements has been superb. Out of a total of 102 requirements, 86 (a **84,3%)** were directly covered with the developments as-is. Fourteen of them (13,7% %) were partially covered and only two (2 out of 102, a ~2%) were not covered.

This means that **98% of aerOS technical requirements** were either directly or indirectly (or partially) covered, making the final reflection concluding as a success.

The ones that failed to pass are TR-25 and TR-102. *TR-25: Resource availability* was a MUST requirement revolving around eliminating the possibility of resources starvation in the continuum. Since aerOS has not focused on resources provisioning (e.g., OpenStack, OpenNebula, Terraform, Ansible…) but in service orchestration, this requirement was not aligned with aerOS Meta-OS, therefore **should not be considered for a real analysis of coverage**. On another note, *TR-102- Communication of distributed services in real time*, which was a SHOULD requirement, was disregarded in WP3, since there has not been the need (and was not the focus) to measure and guarantee certain jitter and latency thresholds met.

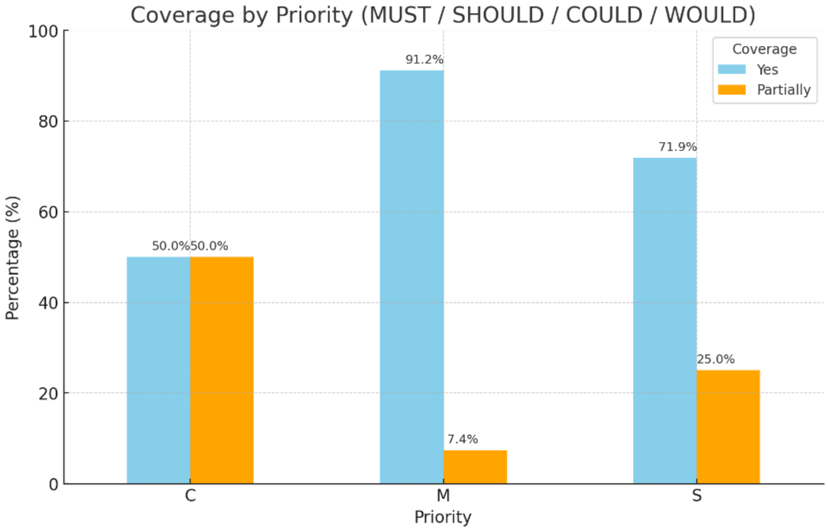


Figure 4. Coverage per priority (Must / Should / Could / Would)

With regards to those covered partially it is relevant first to analyse that, according to the figure above, those are distributed oddly per priority type of requirement. As it can be seen, the technical requirements tagged as

"MUST" were tackled to maximize coverage and accuracy, thus only 7,4% (5 out of 68 MUST requirements) were partially covered, in contrast to the other 63, that were directly covered. However, the percentage of alignment in SHOULD requirements is more relaxed. There, a 25% was partially covered (8 out of 32 SHOULD requirements) whereas 75% of them (still an amazing value) – 24 requirements- were satisfied so far. With regards to COULD requirements, these were only 2, with one fully covered and one partially covered.

Looking globally to those requirements only partially covered (TR-9, TR-26, TR-39, TR-40, TR-50, TR-53, TR-59, TR-61, TR-66, TR-68, TR-81, TR-85, TR-86, TR-101), they were scattered in a reasonably uniformly way across the Technical Areas of aerOS, as it can be observed in the next bar chart:

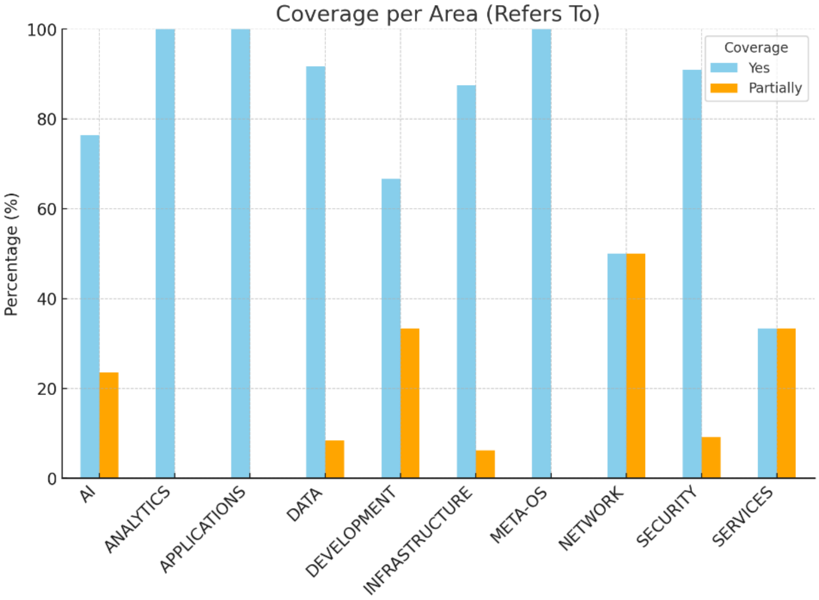


Figure 5. Coverage per Technical Area (Yes, No, Partially)

Those partially-covered related to **AI** revolved around AI jobs concept that had no representation in a real integration structure, therefore its full potential remained at low automation range. Also, task description for AI did not require the incorporation of additional parameters in service (TOSCA) or data description. With regards to **data**, the privacy labelling was not prioritised since alternative mechanisms were already in place in the Meta OS. Touching upon **network**, three requirements were only partially addressed since aerOS focused mainly on workload orchestration rather than complete network slicing or programmability and since latency was not a design priority, thus performance was not key for optimization. Security-wise, coding environment was prepared but not automated in Meta OS auxiliary **services** development pipeline (not necessary but possible in the future). Finally, service availability metrics were addressed as a secondary priority, same as resource availability in K8s-only (self-scaling) scenarios.

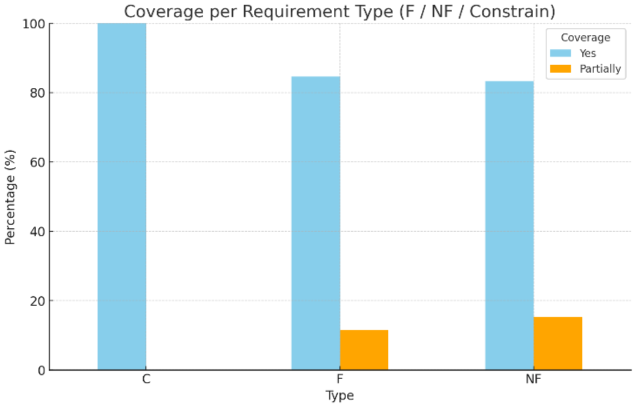


Figure 6. Coverage per requirement type (Functional – Non-functional – Constraint)

Finally, regarding the requirement type division, those that were considered Constraints (2) were 100% covered, whereas both Functional and Non-Functional requirements showed more than 80% of coverage, with slightly more "partial" coverage in those tagged as Non-Functional.

6.2. Analysis of user and system requirements (pilot-related)

Same as for technical requirements, user and system requirements (those related to pilots) were initially identified in M9 (May 2023) and revised and enhanced on M18 (February 2024). **A total of 74 requirements** were defined, appointed to the 5 pilots of the project. These requirements were defined to clarify the system's subject matter, the non-functional requirements refer to the behavioural properties that the specified functions must have, such as performance, usability, etc. Both functional and non-functional aerOS requirements listed below are result of intensive communication among stakeholders, as their analysis was performed for all pilots separately.

Some useful numbers and categorization emanated from such an effort.

Table 14: Summary of aerOS user and system requirements per classification.

Requirement classification	Quantity
Functional	50
Non-functional	24

Table 13: Summary of aerOS user and system requirements per type.

Requirement type	Quantity
System	48
User	26

Table 15:Summary of aerOS user and system requirements per priority.

Requirement priority	Quantity
Would	4
Must	50
Should	13
Could	6

By this deliverable (D5.6, in M38), a parallel analysis has been performed (also for all pilots separately). The complete relation of user and system requirements (per pilot) and a brief explanation of how those have been covered is available in Appendix D.

Such analysis has derived into the following results and conclusions:

6.2.1. Pilot 1 - Data-Driven cognitive production lines

Pilot 1 revolves around manufacturing and is composed of four different scenarios. The analysis of the activities done in the last two months of the project can be found in Section 2, while the completion and achievement of KPIs is summarised in Section 4 and extensively discussed in Appendix H.

The two figures below represent how the requirements identified more than 20 months ago (and even before) have been covered in the pilot (please, bear in mind that the previous encompasses the whole pilot). First, the coverage of the total of 17 requirements is indicated in a pie chart divided in YES, NO or Partially. Second, a reflection on those that have been (either totally or partially) achieved per Priority (Must, Should, Could, Won't) is provided with a bar chart.

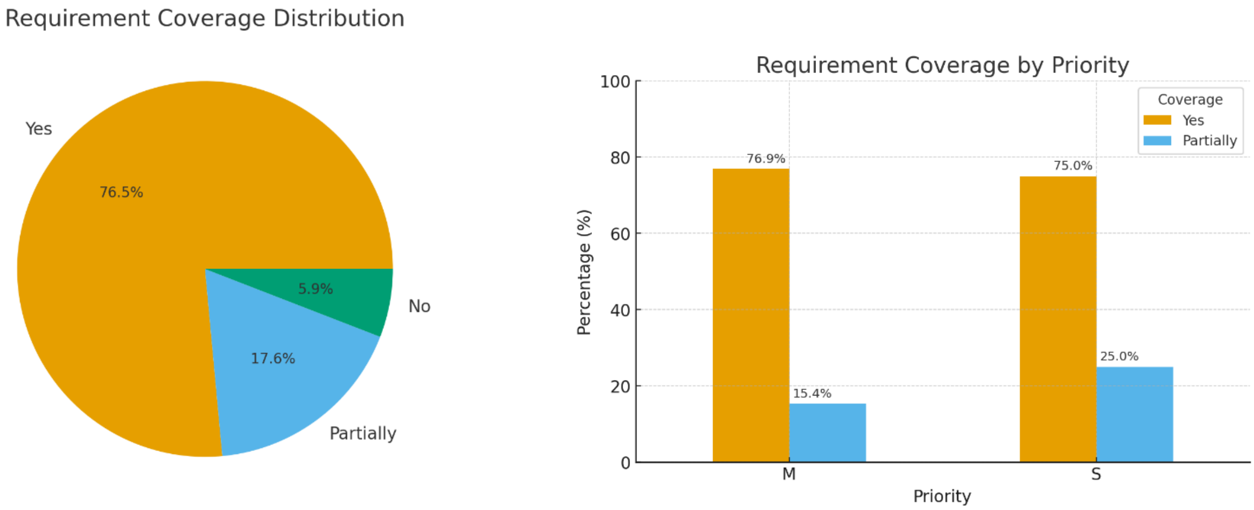


Figure 7. Pilot 1 requirement analysis: per achievement coverage and per priority

As it can be seen, the percentages are very favorable: 76,5% of the requirements (13 out of 17) were directly positively covered, related mostly to edge-to-cloud data sharing, secure communication, data interoperability, support of various type of devices, integration with existing systems including TSN networking and on-demand re-scheduling of workloads. 17,6% of the requirements (3 out of 17) were reported as partially achieved. These requirements (R-P1-11, 12 and 13) are transversal to the 4 scenarios, and focus on the automatic selection of AI models and the establishment of specific Human Machine Interfaces for them. Since not all scenarios in P1 have needed to deliver such elements to achieve their goals, those are indicated as partial. In particular, P1.4 (MADE/POLIMI) has successfully demonstrated their completion.

On the negative note, one MUST requirement was NOT achieved (R-P1-15). However, analysing the content of it, pilot 1.3 team (SIEMENS) specifically recognized that this action track has not had a direct relation with the goals of the pilot. As a matter of fact, object recognition has simply not been tested in the experiment but could easily be understood as a byproduct for potential validation. The scenario has focused on real-time critical services and critical task re-scheduling and re-allocation, using Behaviour Trees and integration with legacy systems and innovative (own) edge solutions such as SIEMENS Industrial Edge. Therefore, the reflection is very positive.

To sum up in light of prioritisation, 92,3% of the MUST requirements (16 out of 17) were either directly or partially achieved, where all (100%) of SHOULD requirements have been successfully checked.

6.2.2. Pilot 2 - Containerised edge computing near renewable energy sources

Pilot 2 focuses on the delivery, test and integration of containerised edge data centers as part of renewable energy-powered continuum. Same as for the other pilots, the analysis of the activities done in the last two months of the project can be found in Section 2, while the completion and achievement of KPIs is summarised in Section 4 and extensively discussed in Appendix H.

The two figures below represent, same as in the other pilots, how the 11 requirements of pilot 2 have been covered, either by final result (YES, NO or Partially) and those achieved per Priority.

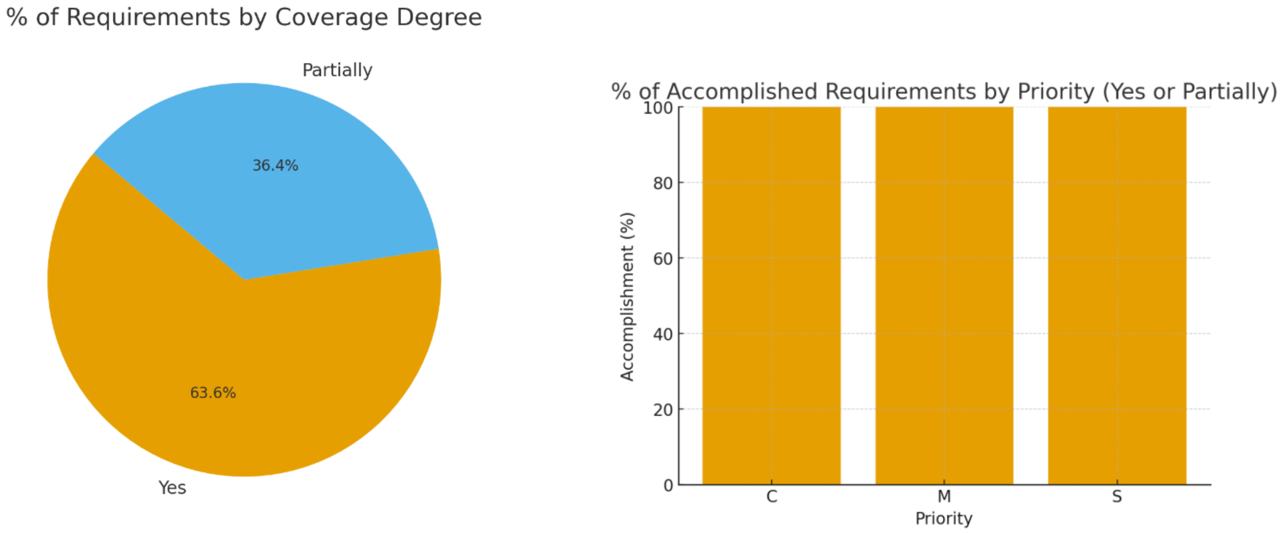


Figure 8. Pilot 2 requirement analysis: per achievement coverage and per priority

As it can be observed, all requirements of pilot 2 (regardless of their priority rate) have been successfully accomplished. 63,6% of them (7 out of 11) were directly covered by the developments and experimentation executed in aerOS.

Only 4 (out of 11), meaning 36,4%, have been considered partially met. R-P2-1 and R-P2-2 were identified and described so that aerOS should react to changing circumstances and adapt heavy workloads and applications, redirecting to queues and the type of energy available. In these regards, the users of aerOS can delete tasks, and different resources of a *nodepool* are automatically selectable and shifted. However, large applications are not prioritised and the amount of energy available sources have prevented the team to mark them as fully achieved. Notwithstanding, the validation is better than expected, as it can be seen in Section 4. On another note, definition of tenant separation (the other two partially covered requirements) is possible but not automatically incorporated in a flow.

6.2.3. Pilot 3 - High Performance Computing Platform for Connected and Cooperative Mobile Machinery to improve CO2 footprint

Pilot 3 focuses on agricultural machinery operations, in particular in real-time communication in rural areas between electric and fuel-based tractors that equip distributed computing elements; those act as a swarm that, thanks to aerOS technologies, optimize AI operations and certain agricultural activities. Same as for the other pilots, the analysis of the activities done in the last two months of the project can be found in Section 2, while the completion and achievement of KPIs is summarised in Section 4 and extensively discussed in Appendix H.

The two figures below represent, same as in the other pilots, how the 5 requirements of pilot 3 have been covered, either by final result (YES, NO or Partially) and those achieved per Priority.

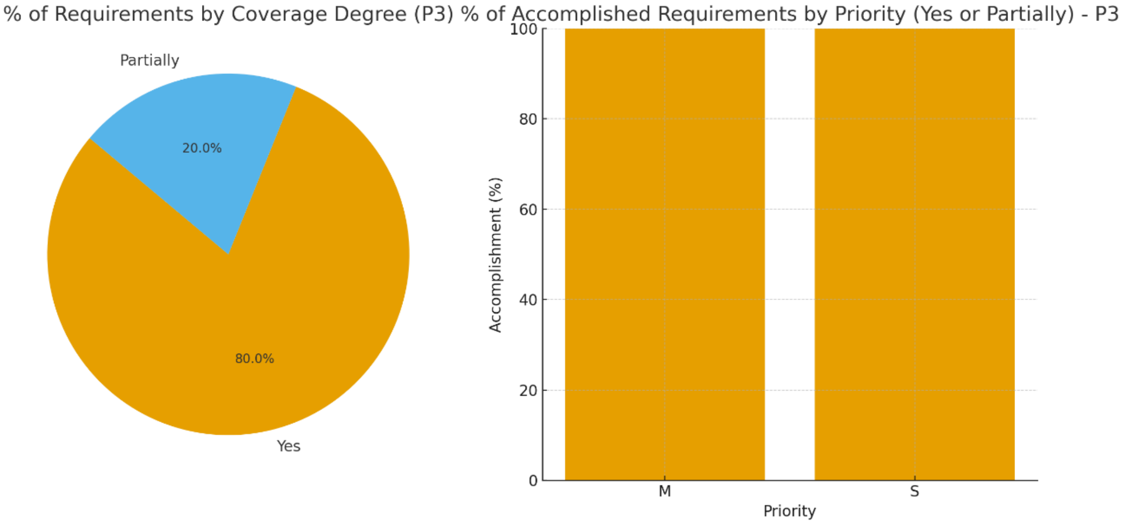


Figure 9. Pilot 3 requirement analysis: per achievement coverage and per priority

As observed in the figures, all MUST and SHOULD requirements of pilot 3 have been either completely or partially covered. 80% of them (4 out of 5) can be considered directly achieved.

R-P3-2 (SHOULD priority) defined the requisite of integrating TTControl's HPCP in John Deere tractors, defining (should, if possible) a specific tolerable latency in the communication among tractors and with edge nodes. Since the latency is not monitored natively (not a goal of the pilot) but given that the integration has been fully achieved (check Section 2.2.3), this was considered partial.

6.2.4. Pilot 4 - Smart edge services for the Port Continuum

Pilot 4 focuses on applying innovative Cloud-Edge-IoT technologies over a continuum in a Smart Port in Limassol, Cyprus. In particular, cranes information is gathered from PLCs and a distributed computing approach is adopted to apply novel AI models to detect defects in containers, and other operational purposes in the port terminal. Same as for the other pilots, the analysis of the activities done in the last two months of the project can be found in Section 2, while the completion and achievement of KPIs is summarised in Section 4 and extensively discussed in Appendix H.

The two figures below represent, same as in the other pilots, how the 20 requirements of pilot 4 have been covered, either by final result (YES, NO or Partially) and those achieved per Priority.

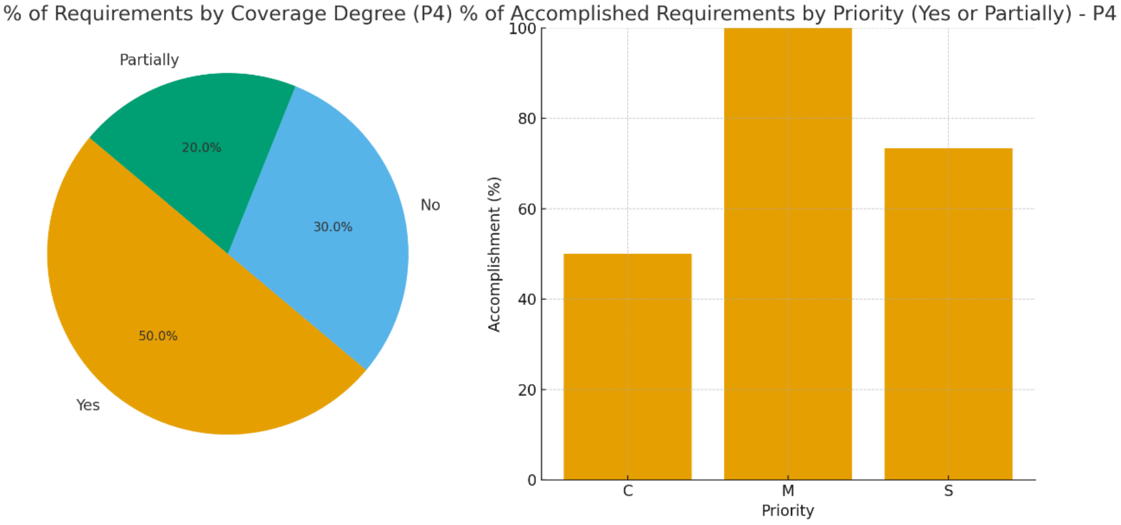


Figure 10. Pilot 4 requirement analysis: per achievement coverage and per priority

Observably above, all MUST requirements of Pilot 4 have been achieve, either completely or partially. This fact, despite some of the numbers above, speaks wonder about pilot 4's capacity to overachieve expectations: According to the KPIs in Section 4, Pilot 4 has accomplished its goals and surpassed the target values.

Regarding requirements, it is clear that a reflection should be done about the 30% (6 out of 20) that were not met. Actually, the reality is more complex than a non-achievement explanation. Four of those requirements R-P4-6 to 10 were referencing works related to applying AI models over Engines of the motor of certain cranes. This departed from the assumption that the STS cranes in the terminal (fuel-based) were an essential element of the pilot. However, the evolution of the pilot (as it can be checked in D5.3, D5.4 and Section 4 of this deliverable D5.6) required the team to focus in other AI models that were necessary to complete the goals: Hydraulic system of straddle carriers, Container plate identification, Detection of damaged containers at the edge with Computer Vision models and the application of Federated Learning in the pilot. Therefore, it can be concluded that, since those requirements were either COULD or SHOULD, and that ALL MUST requirements were met, it is not considered a shortcoming. Lastly, requirement R-P4-2 suggested the integration of the Terminal Operating System (TOS) into aerOS, which was not necessary since the information contained there was not relevant for the pilot purposes, having been substituted by real-time monitoring (IoT in the port).

On another note, 20% of the requirements (4 out of 20) were only partially accomplished, particularly R-P4-11 and 12, which expected the usage of straddle carriers telemetry to estimate and forecast genset vibrations (COULD) and inclination issues (SHOULD). As the parentheses illustrate, it was not essential to tackle such developments, but in any case those are well compensated by the fact of utilising (thus, validating) telemetry from straddle carriers to predict an overtemperature of the engine of the inverters in such type of container handler equipment.

6.2.5. Pilot 5- Energy Efficient, Health Safe & Sustainable Smart Buildings

Pilot 5 has deployed IoT scenarios in an innovative Smart Building testbed in the premises of OTE in Greece. There, several AI models for sit recommendation and energy efficiency have been collated to optimize health and safety in working environments. A myriad (and a quantity) of sensors have provided data via aerOS' Data Fabric, and services have been orchestrated through aerOS Meta OS to achieve the goals of this pilot. Same as for the other pilots, the analysis of the activities done in the last two months of the project can be found in Section 2, while the completion and achievement of KPIs is summarised in Section 4 and extensively discussed in Appendix H.

The two figures below represent, same as in the other pilots, how the 21 requirements of pilot 5 have been covered, either by final result (YES, NO or Partially) and those achieved per Priority.

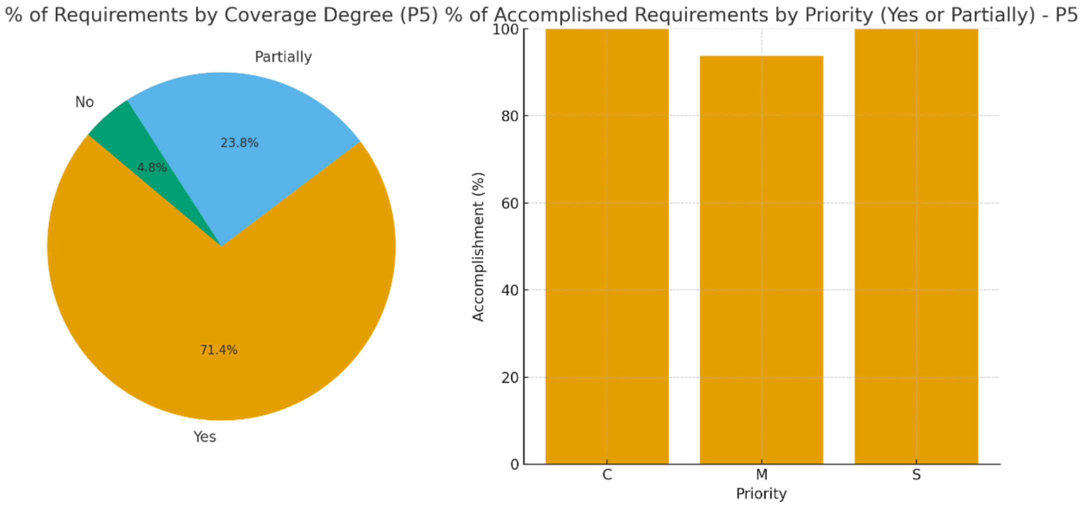


Figure 11. Pilot 5 requirement analysis: per achievement coverage and per priority

A total of 71,4% (15 out of 21) of Pilot 5's requirements were achieved directly by the development of the team involved in the use case. Also, analysing by priority, all COULD and SHOULD requirements were either totally or partially achieved.

With respect to MUST requirements, only 1 failed to be achieved. Such requirement that NOT covered, the analysis is clear: there is absolutely acceptable to consider Pilot 5 a total success. It is the case of *R-P5-1 (Cloud storage and Federated Learning)*, FL has finally not been used in this pilot, since the priority has been given to the crucial operational tasks that should lead to the successful execution of the scenarios. In this sense, AI models have been developed for the currently available building. FL is demonstrated in pilot 4, therefore the compatibility of these methods and technologies is guaranteed in aerOS.

Therefore, 20 out of 21 requirements were either completely or highly partially achieved. Those that are not totally covered (5 out of 21, 2 of priority MUST and 3 of priority SHOULD) revolve around non-essential activities of a very successful trial in aerOS. In particular R-P5-5 investigated the automation of IoT configuration matters, which is partially covered by self-healing and self-orchestration capabilities in aerOS' Infrastructure Elements. On another side, R-P5-12 and 13 are not considered fully achieved since the acceptance criteria established checks that have finally not been included in the demonstrator, such as detection of false positives or network overhead connected to QoE surveillance. Since those were not fundamental activities, and the pilot has concluded satisfactorily, there is not an issue with this action.

All in all, Pilot 5 has redounded in an excellent complete pilot, with exceeding KPIs, necessary coverage of requirement and outstanding level of learning and scientific and impact production.

7. Final KeVI analysis

In Deliverable 5.5, our method of KVI analysis was structured in four general steps. We first determined the Key Values (KVs) **of social significance, provided from the UN Sustainable Development Goals**. These values, for instance, sustainability (economic, environmental and social), digital inclusion, personal health etc., were then linked to Key Value Indicators (KeVIs). KeVIs capture their measurable societal impact. Moving on, Key value enablers were reviewed to determine the drivers that influence the adoption and scalability of the use cases. Finally, related Key Performance Indicators (KPIs) were related to KeVIs. This connection ensures that technical performance metrics could be aligned with broader societal goals. This process provided a systematic way to connect technological development with value-focused outcomes.

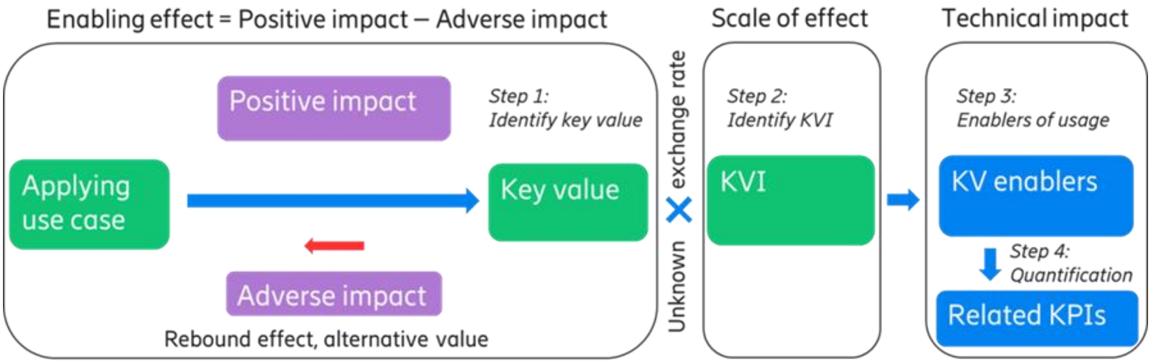


Figure 12: The four steps of the adopted KVIs methodology

In the current deliverable, the methodology has been formulated and expanded following the recently established KVI framework presented in the paper titled "Key value indicators: A framework for values-driven next-generation ICT solutions.". While the initial approach in D5.5, had already combined societal Key Values (KVs) with Key Value Indicators (KeVIs), enablers, and KPIs, the expanded methodology adds conceptual clarity by distinguishing values as criterion and goal and values outcome and incorporating the technical and system enablers.

KeV as criterion and goal establishes the overall objectives of each use case, identifying what societal priorities are met and what Sustainable Development Goals (SDGs) are targeted. KeV as outcome determines the measurable outcomes expected after applying the use case, detailing involved stakeholders and affected areas, e.g. processes, service etc. The Use Case KVIs then define the indicators that measure the impact that can be economic, social, environmental etc. Indicators can be approached by quantitative measures like reduction of emissions or qualitative way such as user satisfaction. Following is the connection of these effects to technical performance measures, the KPIs. As KPIs, we define the operational metrics utilized in measuring the KVIs, while the Target Values define the desired goals or standards for success attainment.

To conclude this value-based analysis, the framework also incorporates enablers. Technical Enablers specify the technologies to enable the use case, i.e., analytics software, AI models, or 5G networks etc. and their corresponding Enabler KVIs specify the benefits and potential risks involved in deploying these technologies. Their potential benefits or risks include improvements in efficiency and accuracy as well s the challenges such as interoperability or security risks. At a more general level, System Enablers characterize the infrastructural and organizational aspects, such as data fabrics or edge services, which combine technical components into a working ecosystem. Their Enabler KVIs prioritize the systemic advantages these enablers contribute, for example reducing power usage or improving resilience, and lists possible drawbacks, such as higher complexity or higher investment. All these fields indicate an entire sequence from abstract societal values to technical implementation, making the analysis of use cases both value-driven and operationally grounded.

7.1.1. Pilot 1 Data driven cognitive production lines

Use case: Data-Driven Cognitive Production Lines

Pilot 1 demonstrates how aerOS can transform industrial operations into more sustainable, flexible, and intelligent systems. Across four complementary use cases, the pilot addresses challenges ranging from energy efficiency and CO_2 footprint monitoring to advanced metrology, agile production reconfiguration, and cross-factory logistics. By embedding intelligence at the edge, orchestrating resources seamlessly, and enabling real-time monitoring and control, the pilot shows how manufacturers can achieve both environmental and economic sustainability while moving toward zero-defect production.

The sub-pilots highlight this vision in practice: optimizing energy consumption on a drone production line in Switzerland; connecting metrology machines into a shared “Metrology Continuum” in Spain; enabling dynamic reconfiguration of production with AGVs and robotic arms in Germany; and coordinating AGV swarms across two connected factories in Italy. Together, these scenarios showcase how aerOS provides the foundation for resilient, data-driven, and sustainable manufacturing, aligning with Europe’s goals for innovation, competitiveness, and responsible resource use.

Sub-Pilot 1.1

Sub-Pilot 1.1 “Green Manufacturing (Zero Net-Energy) & CO_2 Footprint Monitoring” takes place at the Swiss Smart Factory (Switzerland Innovation Park Biel/Bienne). It focuses on improving the energy efficiency and sustainability of a drone production line. The use case monitors the carbon footprint of manufacturing by tracking machine energy consumption in real time. Using a network of IoT sensors and actuators, the production line optimizes utility usage. The objective is to minimize wasteful energy use while maintaining production performance. By deploying the aerOS platform in this smart factory environment, Sub-pilot 1.1 aims to demonstrate how edge orchestration and analytics can autonomously adjust industrial operations to be more frugal and sustainable.

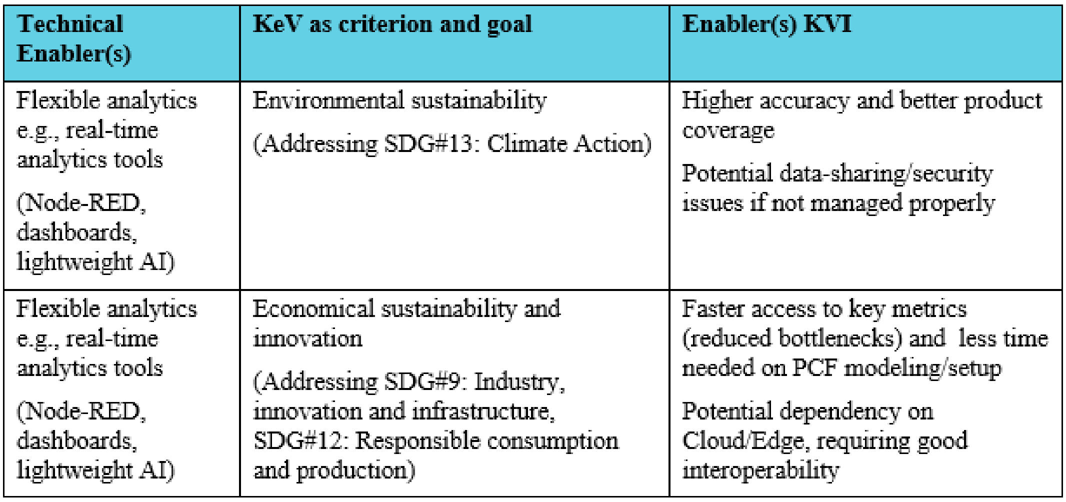
On the **environmental side**, the pilot targets the decrease of carbon footprint through precise CO_2 calculations and broad product coverage. This is reflected in KeV1, with KVIs like accurate CO_2 prediction and measurement per product. The corresponding KPIs, such as achieving >80% accuracy in CO_2 prediction and covering 10–100% of products, provide measurable milestones. For Sub-Pilot 1.1, this means embedding aerOS components to enable real-time monitoring and predictive analytics, supporting manufacturers and customers in lowering production-related emissions.

KeV as criterion and goal	KeV as outcome	Use case KVIs	KPIs	Target Value	Achieved
Environmental sustainability (Addressing SDG#13: Climate Action)	**KeV1:** Decrease of the carbon footprint via precise calculation and broad coverage of products **Stakeholder:** Industrial Manufacturer, Customers **Effect on:** Production Processes, Environment	**KVI1:** Accurate CO2-footprint prediction **KVI2:** CO2-footprint measurement for individualized produced products	**KPI 2.1.4:** Accuracy of CO2-footprint prediction	>80%	YES
			KPI 2.1.5: C02-footprint measurement	10 – 100%	YES
Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production)	**KeV1:** Efficient definition of calculation models. **KeV2:** Transparent CO2/PCF data access. **Stakeholder:** Operators, Technical Teams **Effect on:** Modeling, Data access Times, Productivity	**KVI1:** Improved efficiency in calculation model definition. **KVI2:** Fast access to CO2/PCF data.	**KPI 2.1.10:** Definition of the calculation model	>30%-time reduction (min)	YES
			KPI 2.1.11: Time required to access CO_2/PCF data (transparency)	< 2 minutes	YES

Figure 13: KVI analysis of sub-Pilot 1.1

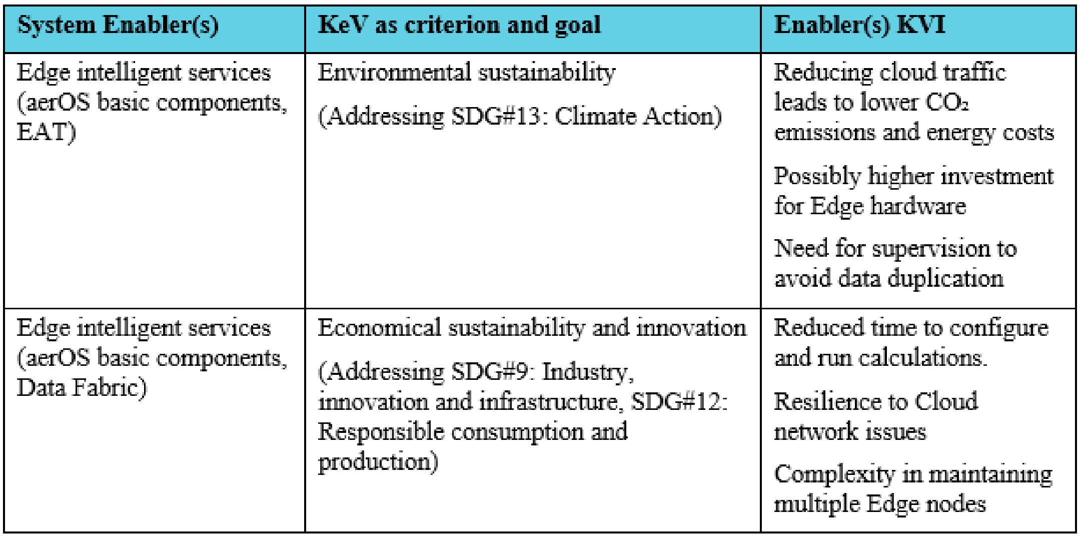
Economically, Sub-Pilot 1.1 addresses efficient calculation models and transparent data access (KeV1 and KeV2). The use of improved models and faster data retrieval ensures operators and technical teams can work with up-to-date CO_2/Product Carbon Footprint (PCF) data, minimizing delays and enhancing productivity. KPIs such as reducing calculation time by over 30% and accessing CO_2/PCF data in under two minutes ensure that innovation translates into real-world time and cost savings. These improvements are especially relevant in manufacturing settings where delays or inaccuracies can ripple across entire supply chains.

To achieve these goals, technical and system enablers play a crucial role. Flexible analytics tools (like Node-RED, lightweight AI, and real-time dashboards) contribute to higher prediction accuracy, better product coverage, and faster access to key metrics, though they also introduce challenges around data sharing and cloud dependencies. Additionally, edge intelligent services provided through aerOS, such as EAT and Data Fabric, reduce cloud traffic, lower emissions, and enhance resilience to network issues, while requiring careful supervision to prevent data duplication and manage the complexity of multiple edge nodes. Together, these enablers ensure that Sub-Pilot 1.1 not only delivers on its sustainability promises but does so in a technically robust and economically scalable way.



Technical Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Flexible analytics e.g., real-time analytics tools (Node-RED, dashboards, lightweight AI)	Environmental sustainability (Addressing SDG#13: Climate Action)	Higher accuracy and better product coverage Potential data-sharing/security issues if not managed properly
Flexible analytics e.g., real-time analytics tools (Node-RED, dashboards, lightweight AI)	Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production)	Faster access to key metrics (reduced bottlenecks) and less time needed on PCF modeling/setup Potential dependency on Cloud/Edge, requiring good interoperability

Figure 14: Technical enablers of sub-Pilot 1.1



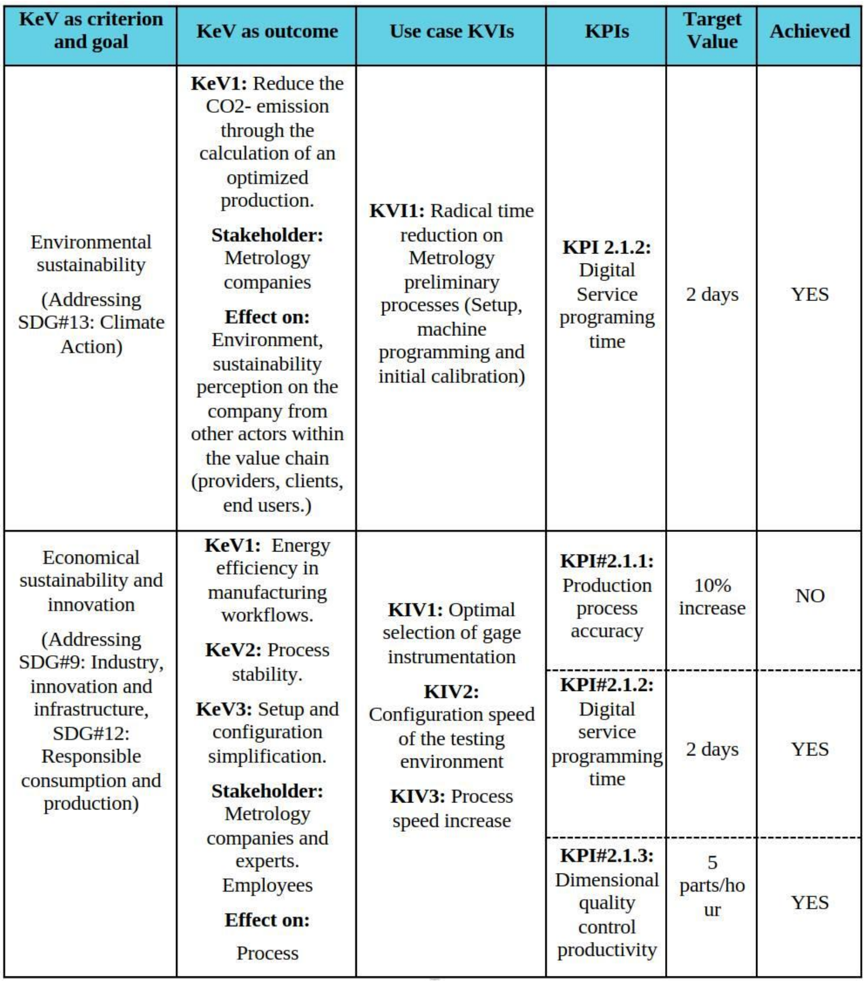
System Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Edge intelligent services (aerOS basic components, EAT)	Environmental sustainability (Addressing SDG#13: Climate Action)	Reducing cloud traffic leads to lower CO_2 emissions and energy costs Possibly higher investment for Edge hardware Need for supervision to avoid data duplication
Edge intelligent services (aerOS basic components, Data Fabric)	Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production)	Reduced time to configure and run calculations. Resilience to Cloud network issues Complexity in maintaining multiple Edge nodes

Figure 15: System enablers of sub-Pilot 1.1

Sub-Pilot 1.2

As companies demand ever higher levels of productivity, flexibility, and excellence, the metrology sector must evolve. Production metrology in particular has to adopt new technologies to become faster, more accurate, and more resilient. Pilot 1.2 addresses this challenge at Innovalia's Didactic Factory in the Automotive Intelligence Center, where the aerOS Meta-OS has been deployed to unlock the full potential of IoT–Edge–Cloud capabilities. With aerOS, services can be deployed and managed remotely, authentication is handled in a secure and robust way, and a stronger edge layer ensures reliability on the shop floor.

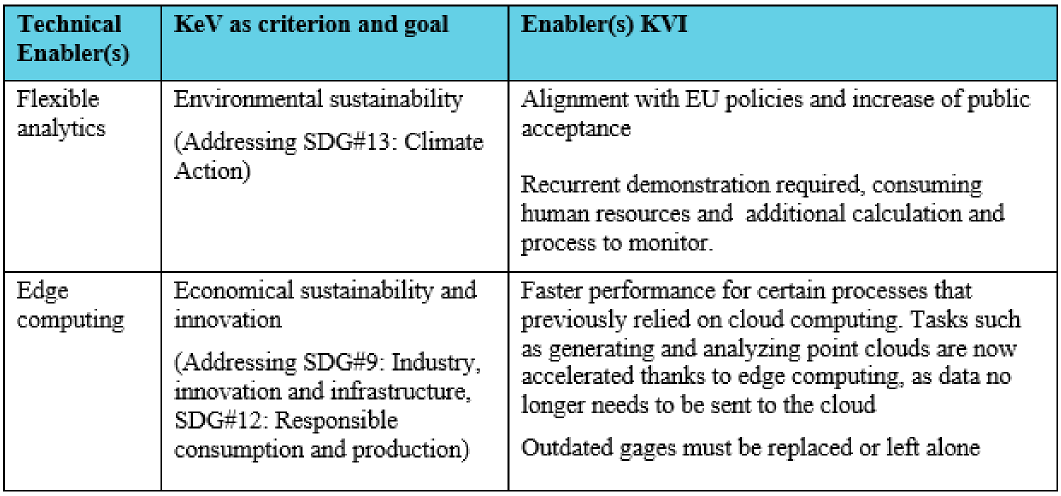
The impact of this pilot is measured through both environmental and economic sustainability objectives. On the environmental side, the focus is on reducing CO_2 emissions by making metrology processes more efficient. This is captured in the target of **reducing digital service programming time to just two days**, cutting energy use and demonstrating responsible production practices. On the economic side, the pilot improves process stability, setup simplification, and workflow efficiency. Clear performance targets guide progress: a 10% increase in process accuracy, maintaining digital service programming time at two days, and achieving a throughput of five parts per hour in dimensional quality control.



KeV as criterion and goal	KeV as outcome	Use case KVIs	KPIs	Target Value	Achieved
Environmental sustainability (Addressing SDG#13: Climate Action)	**KeV1:** Reduce the CO2- emission through the calculation of an optimized production. **Stakeholder:** Metrology companies **Effect on:** Environment, sustainability perception on the company from other actors within the value chain (providers, clients, end users.)	**KVI1:** Radical time reduction on Metrology preliminary processes (Setup, machine programming and initial calibration)	**KPI 2.1.2:** Digital Service programing time	2 days	YES
Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production)	**KeV1:** Energy efficiency in manufacturing workflows. **KeV2:** Process stability. **KeV3:** Setup and configuration simplification. **Stakeholder:** Metrology companies and experts. Employees **Effect on:** Process	**KIV1:** Optimal selection of gage instrumentation **KIV2:** Configuration speed of the testing environment **KIV3:** Process speed increase	**KPI#2.1.1:** Production process accuracy	10% increase	NO
			KPI#2.1.2: Digital service programming time	2 days	YES
			KPI#2.1.3: Dimensional quality control productivity	5 parts/hour	YES

Figure 16: KVI analysis of sub-Pilot 1.2

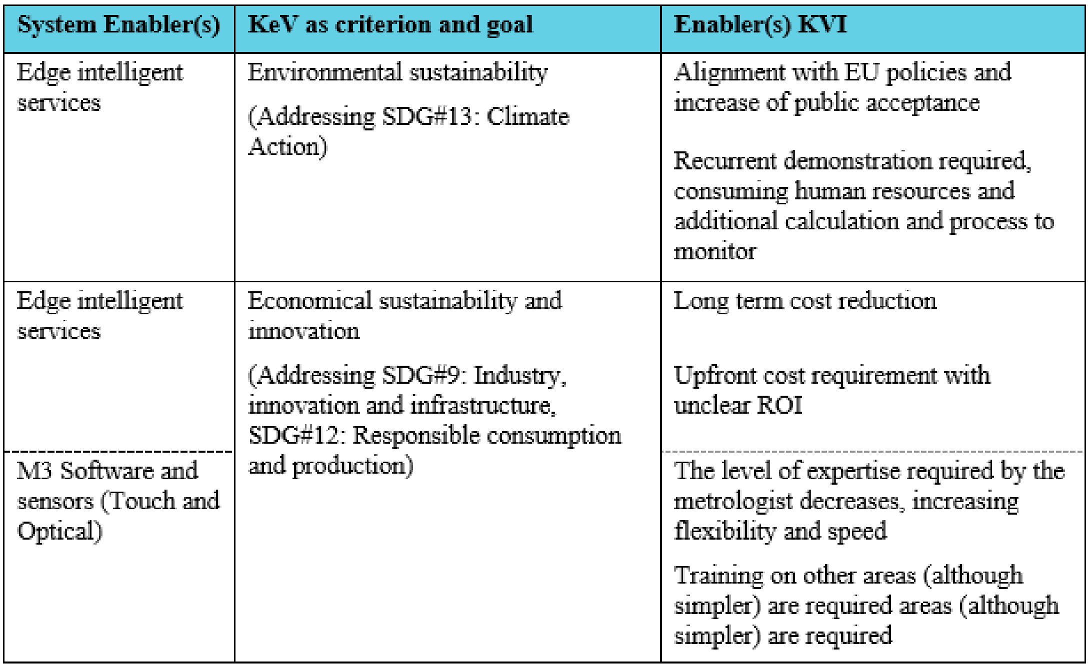
To reach these outcomes, the pilot leverages powerful technical enablers. Flexible analytics ensure alignment with EU sustainability policies and increase public acceptance, though they require recurrent demonstrations and monitoring. Edge computing accelerates processes such as point cloud generation and analysis, eliminating the delays of cloud-only approaches, even if it requires replacing outdated gages. On top of this, system enablers like edge intelligent services drive long-term cost reductions while M3 software with touch and optical sensors lowers the expertise barrier for operators, increasing speed and flexibility without compromising quality.



Technical Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Flexible analytics	Environmental sustainability (Addressing SDG#13: Climate Action)	Alignment with EU policies and increase of public acceptance Recurrent demonstration required, consuming human resources and additional calculation and process to monitor.
Edge computing	Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production)	Faster performance for certain processes that previously relied on cloud computing. Tasks such as generating and analyzing point clouds are now accelerated thanks to edge computing, as data no longer needs to be sent to the cloud Outdated gages must be replaced or left alone

Figure 17: Technical enablers of sub-Pilot 1.2

Traditionally, Coordinate Measuring Machines (CMMs) worked in isolation, confined to special factory rooms, separated from production, and each with its own controller and environment. This approach is incompatible with zero-defect manufacturing strategies, which require machines that are connected, intelligent, and secure. Through aerOS, these machines are now part of a shared environment known as the Metrology Continuum. Operators can view all available machines, choose where to run a program, and even activate remote controls such as a virtual joystick. Instead of being tied to a single device, they can combine different hardware and measurement tools, selecting the most suitable machine for each task and accessing IoT-connected devices whenever needed.



System Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Edge intelligent services	Environmental sustainability (Addressing SDG#13: Climate Action)	Alignment with EU policies and increase of public acceptance Recurrent demonstration required, consuming human resources and additional calculation and process to monitor
Edge intelligent services	Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production)	Long term cost reduction Upfront cost requirement with unclear ROI
M3 Software and sensors (Touch and Optical)		The level of expertise required by the metrologist decreases, increasing flexibility and speed Training on other areas (although simpler) are required areas (although simpler) are required

Figure 18: System enablers of sub-Pilot 1.2

Digital twins play a central role in this transformation. Each machine is mirrored virtually, allowing operators to follow its performance in real time, analyze its behavior, and continuously optimize measurement quality.

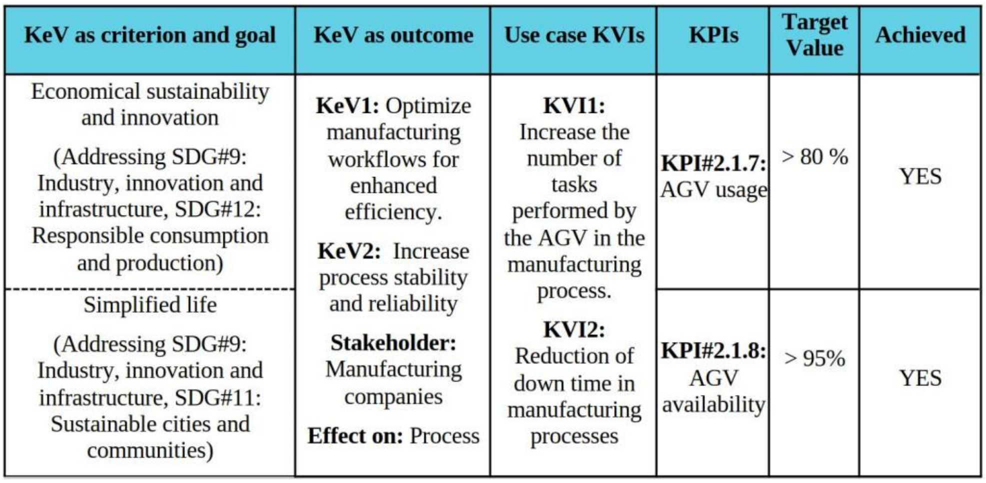
This results in greater precision, higher efficiency, and a step change in flexibility. Metrology is no longer a bottleneck, it is becoming a connected, intelligent ecosystem that supports smarter and more resilient manufacturing.

Through these advances, Pilot 1.2 demonstrates how aerOS transforms metrology into a connected, intelligent, and flexible ecosystem. It improves precision and efficiency today, while laying the foundations for the zero-defect, resilient manufacturing of tomorrow.

Sub-Pilot 1.3

The “Zero ramp-up safe PLC reconfiguration for lot-size-1 production” use case focuses on establishing a versatile production system that demonstrates the potential of modular, efficient, and adaptable manufacturing operations. The scenario showcases the integration of Automated Guided Vehicles (AGVs) and robotic arms through aerOS decentralized intelligence and Siemens SIMATIC Industrial EDGE technology. This implementation enables dynamic production line modifications and demonstrates the feasibility of flexible manufacturing solutions that overcome the limitations of traditional static production systems.

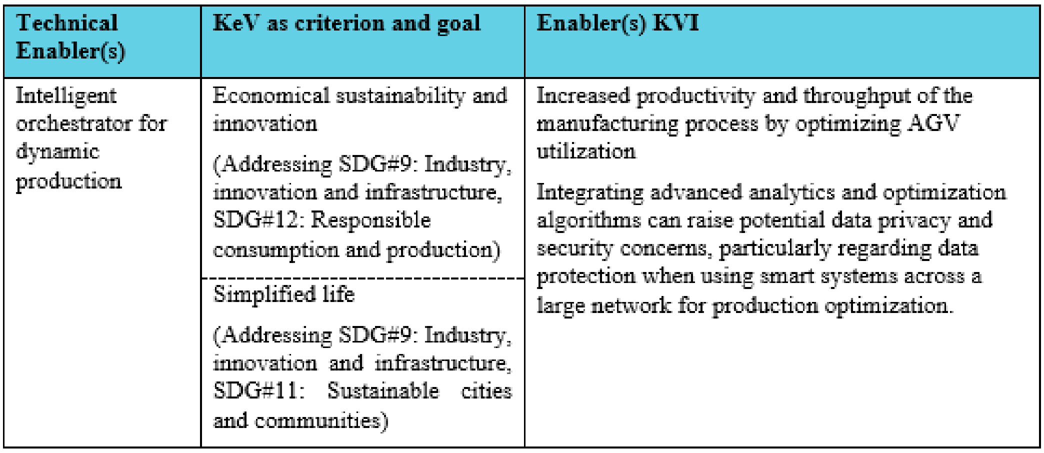
The scenario promotes **economic sustainability and innovation** through its focus on optimizing manufacturing workflows and increasing process stability. The KeVs under this domain target enhanced efficiency and reliability in production processes. The KVIs assess operational aspects such as AGV task performance and manufacturing downtime reduction. Performance is measured through **KPI#2.1.7**, which targets AGV usage above 80%, demonstrating the system's ability to maintain high utilization rates of automated resources. Additionally, **KPI#2.1.8** measures AGV availability with a target of above 95%, highlighting the system's reliability and operational stability.



KeV as criterion and goal	KeV as outcome	Use case KVIs	KPIs	Target Value	Achieved
Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production)	**KeV1:** Optimize manufacturing workflows for enhanced efficiency. **KeV2:** Increase process stability and reliability **Stakeholder:** Manufacturing companies **Effect on:** Process	**KVI1:** Increase the number of tasks performed by the AGV in the manufacturing process. **KVI2:** Reduction of down time in manufacturing processes	**KPI#2.1.7:** AGV usage	> 80 %	YES
Simplified life (Addressing SDG#9: Industry, innovation and infrastructure, SDG#11: Sustainable cities and communities)			**KPI#2.1.8:** AGV availability	> 95%	YES

Figure 19: KVI analysis of sub-Pilot 1.3

The technical enabler "Intelligent orchestrator for dynamic production" addresses these sustainability goals by increasing productivity and throughput through optimized AGV utilization. This is achieved while maintaining robust data security measures to protect sensitive manufacturing information. The system enabler, Siemens Industrial Edge computing platform, complements these capabilities by providing reduced latency and faster response times for real-time production optimization, while also offering lower infrastructure and maintenance costs compared to traditional cloud-based solutions.

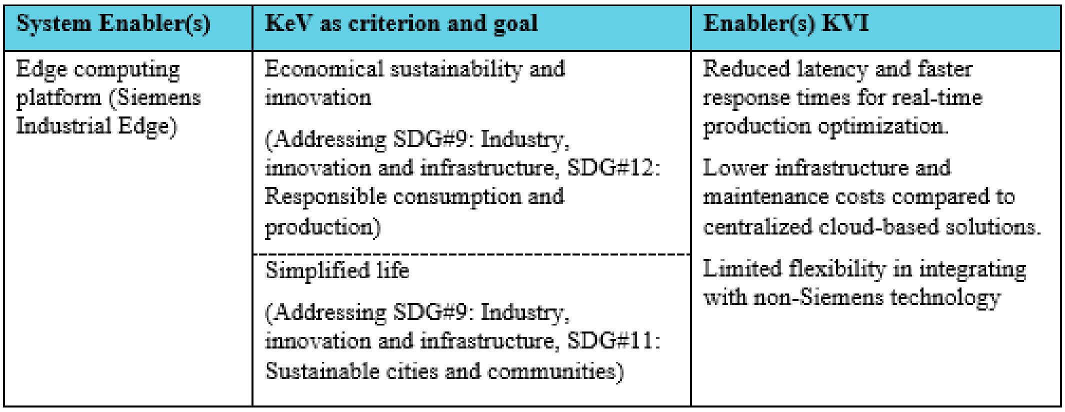


Technical Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Intelligent orchestrator for dynamic production	Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production) Simplified life (Addressing SDG#9: Industry, innovation and infrastructure, SDG#11: Sustainable cities and communities)	Increased productivity and throughput of the manufacturing process by optimizing AGV utilization Integrating advanced analytics and optimization algorithms can raise potential data privacy and security concerns, particularly regarding data protection when using smart systems across a large network for production optimization.

Figure 20: Technical enablers of sub-Pilot 1.3

The pilot's implementation leverages sophisticated hardware setup in a Siemens Tech laboratory, including three AGVs and two mobile robot arm modules. This infrastructure is supported by advanced networking capabilities through Wi-Fi connectivity and industrial-grade protocols. The integration activities focus on implementing key aerOS components such as the High-Level Orchestrator (HLO) and self-capabilities, to enable intelligent service allocation and resource management.

The **simplified life** aspect is addressed through the system's ability to automate complex manufacturing processes and provide flexible, adaptable production solutions. This aligns with both SDG#9 (Industry, innovation and infrastructure), SDG#11 (Sustainable cities and communities) and SDG#12 (Responsible consumption and production) by demonstrating how advanced manufacturing technologies can contribute to more sustainable and efficient industrial operations. The intelligent resource management and optimization capabilities ensure optimal use of manufacturing resources, while the automated orchestration of AGVs and robotic arms minimizes idle time and reduces waste in production processes, directly supporting sustainable manufacturing practices and responsible resource utilization.



System Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Edge computing platform (Siemens Industrial Edge)	Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production) Simplified life (Addressing SDG#9: Industry, innovation and infrastructure, SDG#11: Sustainable cities and communities)	Reduced latency and faster response times for real-time production optimization. Lower infrastructure and maintenance costs compared to centralized cloud-based solutions. Limited flexibility in integrating with non-Siemens technology

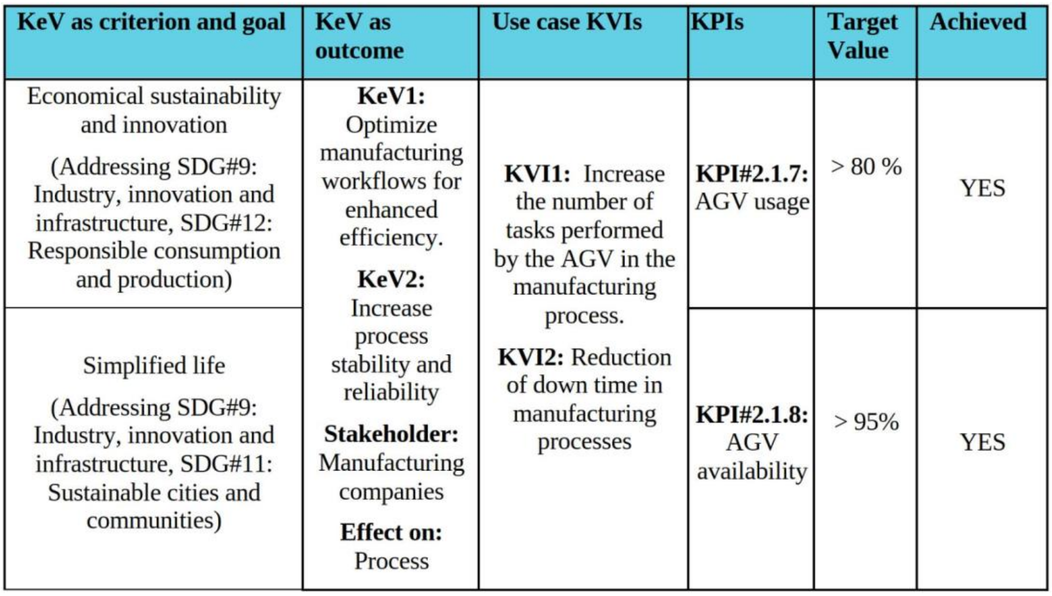
Figure 21: System enablers of sub-Pilot 1.3

The validation activities will be conducted in the laboratory setting, focusing on the system's ability to autonomously make informed decisions about task allocation and resource utilization. The defined KPIs provide clear metrics for evaluating the success of the implementation, ensuring that the scenario delivers measurable improvements in manufacturing efficiency and reliability.

Sub-Pilot 1.4

Sub-Pilot 1.4 demonstrates AGV Swarm, Zero Break-down Logistics, across two connected factories (MADE Competence Center and POLIMI) coordinated through aerOS to optimize intralogistics and outsource work when the primary line nears saturation. Orders are ingested and aggregated by containerized Order Manager apps and exposed as NGSI-LD entities in Orion-LD; a synthetic order generator feeds realistic workloads for end-to-end tests. When thresholds (e.g., queue length, line/AGV status) indicate overload, an ML-based optimizer (Random Forest) dispatches work to the external POLIMI line, where a ROS-based AGV stack, executes predictable linear paths for perceived safety, with a Flask/ROS bridge enabling direct missions from the front-end “AGV Commander.” The multi-domain continuum (NASERTIC entry, MADE edge, POLIMI edge) is orchestrated via the aerOS Management Portal with role-based access (aerOS AAA), spanning K8s/K3s/Docker to reflect platform-agnostic deployment. Together, this validates order-to-execution flow, decentralized logistics intelligence, and smart cross-site allocation in a realistic valve-production scenario.

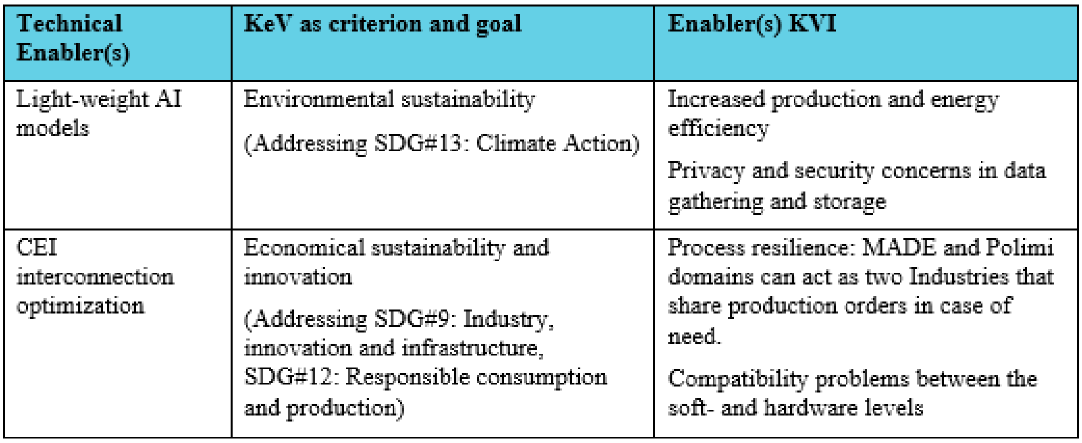
Key values are addressed via KVIs/KPIs, technical and system enablers. More specifically, **economic impact**, aggregating orders and batching material moves with the Order Manager, plus ML-guided outsourcing to POLIMI, targets shorter order lead time, higher throughput, and reduced logistics cost per unit. KPIs are computed from order-state entities and orchestrator logs (e.g., missions per hour, line utilization, outsourced vs. local orders). Technical enablers include the dual Order Manager (MADE/POLIMI), the outsourcing optimizer, and REST APIs added to MADE’s LEA platform for programmatic order launch and status retrieval.



KeV as criterion and goal	KeV as outcome	Use case KVIs	KPIs	Target Value	Achieved
Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production)	**KeV1:** Optimize manufacturing workflows for enhanced efficiency. **KeV2:** Increase process stability and reliability **Stakeholder:** Manufacturing companies **Effect on:** Process	**KVI1:** Increase the number of tasks performed by the AGV in the manufacturing process. **KVI2:** Reduction of down time in manufacturing processes	**KPI#2.1.7:** AGV usage	> 80 %	YES
Simplified life (Addressing SDG#9: Industry, innovation and infrastructure, SDG#11: Sustainable cities and communities)			**KPI#2.1.8:** AGV availability	> 95%	YES

Figure 22: KVI analysis of sub-Pilot 1.4

From the **environmental sustainability** perspective, sub pilot 1.4 pushes computation and control to the edge (12 of 13 applications, ~92%, now run at edge IEs), cutting needless traffic to the cloud and enabling local, lower-latency decisions that reduce AGV idle time and empty runs. KVIs/KPIs include % workloads at edge (baseline “all cloud” → ~92% edge after aerOS), AGV travel per delivered order, and energy per mission derived from Orion-LD telemetry. These are enabled by Orion-LD semantics (NGSI-LD) and federated orchestration over heterogeneous IE nodes.

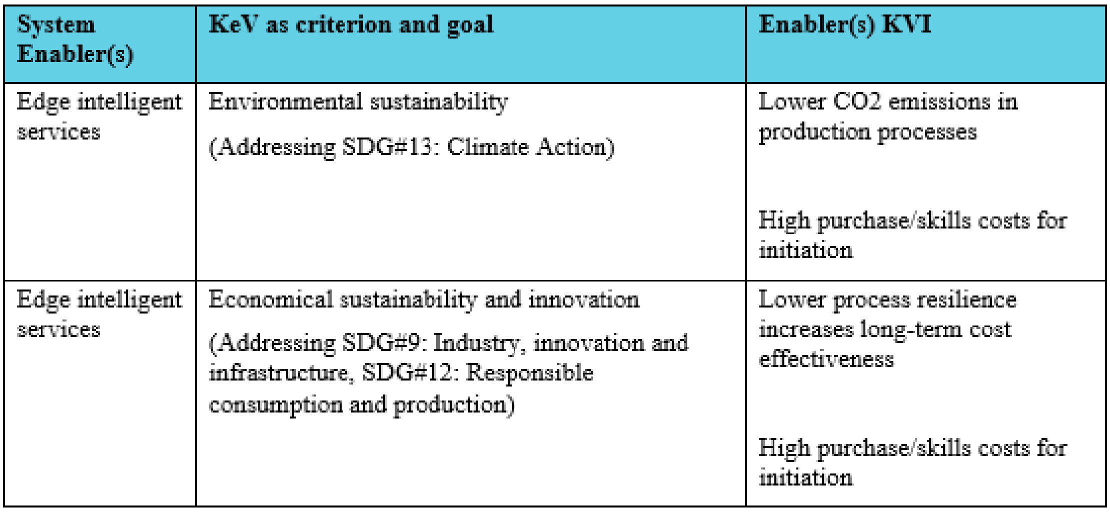


Technical Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Light-weight AI models	Environmental sustainability (Addressing SDG#13: Climate Action)	Increased production and energy efficiency Privacy and security concerns in data gathering and storage
CEI interconnection optimization	Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production)	Process resilience: MADE and Polimi domains can act as two Industries that share production orders in case of need. Compatibility problems between the soft- and hardware levels

Figure 23: Technical enablers of sub-Pilot 1.4

Operational safety and reliability are related to ROS navigation stack which was deliberately adapted away from reactive local planners toward predictable, orthographic movements; obstacle encounters trigger controlled stops (not spin-recovery), improving operator trust.

Interoperability and scalability are accomplished through continuum spans NASERTIC (entrypoint) and two factory domains with K8s/K3s/Docker and mixed ARM64/AMD64 images. Standard interfaces (NGSI-LD/Orion-LD, REST), containerization, and role-based access (aerOS AAA) are used throughout, measured via KVIs like successful cross-domain deployments and API conformance checks. System and technical enablers include the aerOS Management Portal, identity services, Orion-LD integration, and packaged apps (Order Manager, Order Generator, AGV Commander).



System Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Edge intelligent services	Environmental sustainability (Addressing SDG#13: Climate Action)	Lower CO2 emissions in production processes High purchase/skills costs for initiation
Edge intelligent services	Economical sustainability and innovation (Addressing SDG#9: Industry, innovation and infrastructure, SDG#12: Responsible consumption and production)	Lower process resilience increases long-term cost effectiveness High purchase/skills costs for initiation

Figure 24: System enablers of sub-Pilot 1.4

Overall, sub-Pilot 1.4 evidence the aerOS value proposition for lot-size-1 logistics: decentralized intelligence to keep lines flowing, external-factory spillover when saturated, and measurable improvements in edge execution, utilization, and safety, validated through end-to-end scenario tests from order creation to AGV mission completion

7.1.2. Pilot 2 Containerised edge computing near renewable energy sources

Use case: Containerized Edge Computing near Renewable Energy Sources

As it was described in Deliverable 5.5, Pilot 2 addresses the critical challenge of achieving sustainable computing through the strategic deployment of containerized edge computing infrastructure directly at renewable energy sources. The pilot demonstrates the aerOS platform's capability to intelligently orchestrate computing tasks between federated edge nodes and cloud resources while maximizing the utilization of renewable energy and minimizing overall energy consumption. Through two complementary scenarios: Green Edge Processing and Secure Federation of edge/cloud, the pilot validates energy-conscious computing practices that align computational workloads with renewable energy availability, delivering measurable improvements across environmental, economic, societal, and data privacy sustainability domains.

The pilot promotes **environmental sustainability** by focusing on the reduction of carbon footprint and energy waste through strategic placement of edge computing resources at renewable energy premises and intelligent, energy-aware workload distribution. These are critical given the increasing environmental impact of traditional cloud computing infrastructure. The KeVs under this domain include renewable energy utilization maximization, energy consumption optimization, and carbon footprint reduction through intelligent task scheduling. The KVIs assess operational aspects such as monthly renewable energy consumption, green energy share for carbon-aware workloads, and energy efficiency improvements through edge processing. Performance is measured through **KPI 2.2.1**, which targets 20 MWh/month of consumed renewable energy, demonstrating substantial energy usage powered by clean sources. Additionally, **KPI 2.2.5** measures carbon awareness by targeting 60% green energy share for jobs with green energy preference labels, ensuring that environmentally conscious workloads are prioritized for renewable energy sources when available.

Economic sustainability focuses on achieving cost-effective operations through optimized resource utilization, efficient task distribution, and scalable infrastructure deployment that maximizes computational efficiency while minimizing operational costs. The KeVs in this domain emphasize operational efficiency through intelligent resource management, scalability of computing infrastructure, and cost reduction through renewable energy integration. The KVIs relate to CPU utilization efficiency, task execution success rates, and infrastructure scalability indicators. Performance is measured through **KPI 2.2.4,** targeting 80% CPU utilization efficiency to ensure optimal resource usage and energy conservation through proper autoscaling solutions. Furthermore, **KPI 2.2.3** measures the scalability of task distribution with a target of 10,000 tasks executed per month, demonstrating the economic viability and efficiency of the aerOS orchestration system.

KeV as criterion and goal	**KeV as outcome**	**Use case KVIs**	**KPIs**	**Target Value**	**Achieved**
Environmental sustainability (Addressing SDG#13: Climate Action)	**KeV1:** Reduced overall energy usage **KeV2:** Enhancement of energy and resource optimization efforts **KeV3:** Decarbonation **Stakeholder:** Companies **Effect on:** Cost, Environment	**KVI1:** Improved processor utilisation **KVI2:** Percentage of operations utilizing renewable energy sources **KVI3:** Decrease in CO_2 footprint	**KPI#2.2.1:** Consumed renewable energy based on decision making process of aerOS	20 MWh/month	YES
			KPI#2.2.5: Carbon awareness share of green energy	60%	YES
			KPI#2.2.4: CPU utilisation efficiency	80%	YES
Economic sustainability and innovation (Addressing SDG#8: Decent Work and Economic Growth)	**KeV1:** Reduced system's capital intensity **Stakeholder:** Companies **Effect on:** Process	**KVI1:** Improvement in Task Execution Accuracy **KVI2:** Increase the number of tasks performed **KVI3:** Improved economic planning and resource allocation	**KPI#2.2.2:** Effectiveness of task distribution through aerOS to nodes	99.5% of tasks executed on schedule	YES
			KPI#2.2.3: Scalability of task distribution and management through aerOS	10k tasks executed/month	YES

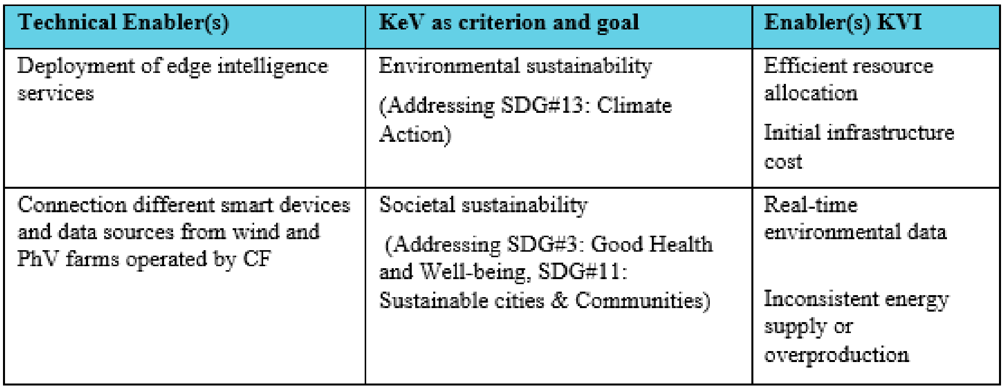
		through accurate energy price forecasting	**KPI#2.2.8:** Precision of the Future Price prediction algorithm	85%	YES
Societal sustainability (Addressing **SDG#3:** Good Health and Well-being, SDG#11: Sustainable cities & Communities)	**KeV1:** Better public health **KeV2:** Better air quality in neighbouring areas **Stakeholder:** Society, Humans **Effect on:** Environment, Humans	**KVI1:** Decrease in CO_2 footprint	**KPI#2.2.5:** Carbon awareness share of green energy	60%	YES

Figure 25: KVI analysis of Pilot 2

Societal sustainability addresses the broader impact of sustainable computing practices on communities and the transition to clean energy infrastructure, focusing on supporting renewable energy adoption and demonstrating responsible technology deployment in critical energy infrastructure environments. The KeVs encompass community benefit through renewable energy integration, technological advancement in sustainable computing, and workforce development in green computing practices. The KVIs track renewable energy infrastructure utilization rates, community energy resilience improvements, and technology adoption indicators. This domain shares **KPI 2.2.1** with the environmental domain, reinforcing the 20 MWh/month renewable energy consumption goal. Additionally, **KPI 2.2.6** evaluates the deployment of federated edge infrastructure, with a target of connecting 2 edge nodes at different renewable energy locations, demonstrating the pilot's contribution to building distributed, community-integrated computing infrastructure.

Environmental and Economic sustainability is addressed through renewable energy monitoring systems which contribute to dynamic energy management and computational load balancing. These systems continuously track parameters such as renewable energy generation from wind and photovoltaic farms, computational demand, and network performance, enabling real-time optimization of task distribution to maximize green energy utilization. As a result, dynamic energy management and the reduction of operational costs occur. That can lead directly to reduced carbon footprint and more efficient utilization of renewable energy sources. However, accuracy must be a priority since the possibility of hardware malfunctions, environmental interferences, or calibration issues may lead to inaccurate sensor readings, which could affect decision-making processes.

The next technical enabler is aligned with **environmental sustainability** as well as with **economic sustainability**. More specifically, energy-aware task orchestration and intelligent workload distribution is very important to ensure optimal resource utilization and carbon footprint reduction. This involves collecting real-time data on renewable energy availability, computational requirements, and system performance to determine the most suitable placement for computing tasks. This can lead to improved energy efficiency and cost optimization since it can provide intelligent allocation of computational resources based on green energy availability. Yet, this technology is not without risk—incorrect predictions or missed optimizations from energy-aware scheduling systems could lead to suboptimal resource allocation or a failure to maximize renewable energy utilization when intervention is required.



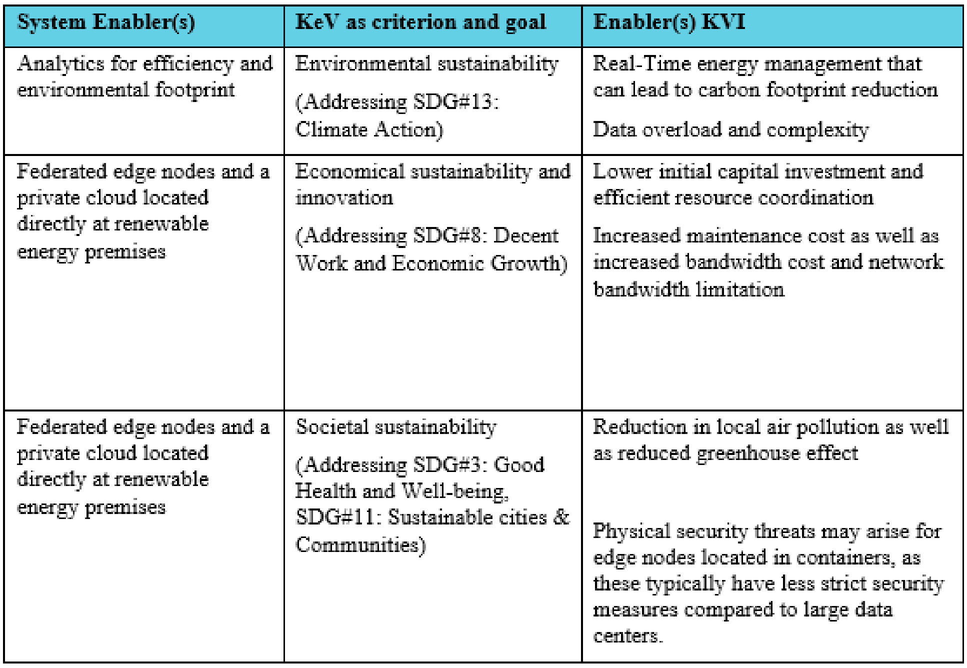
Technical Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Deployment of edge intelligence services	Environmental sustainability (Addressing SDG#13: Climate Action)	Efficient resource allocation Initial infrastructure cost
Connection different smart devices and data sources from wind and PhV farms operated by CF	Societal sustainability (Addressing SDG#3: Good Health and Well-being, SDG#11: Sustainable cities & Communities)	Real-time environmental data Inconsistent energy supply or overproduction

Figure 26: Technical enablers of Pilot 2

In the context of **economic sustainability**, the development of containerized deployment and autoscaling technologies has been conducted. These technologies allow intelligent management of computing resources and support scalable decision-making for resource optimization. To develop efficient and cost-effective infrastructure, it can ensure that resource allocation is not only optimal but also adaptable. This is essential for meeting operational efficiency requirements, but it is also scalable and economically viable. These attributes can assure stakeholders that infrastructure investments are based on proven scalability principles. The implementation of a comprehensive containerized system may come with high initial investment costs, such as infrastructure setup, system integration, and staff training.

From the network and connectivity perspective, federated edge infrastructure management can contribute to a stable, reliable, and flexible connectivity between distributed computing nodes. These networks and components allow faster data transmission and support real-time analytics across the federated infrastructure. They can also enable dynamic execution of load balancing protocols as well as improve coordination among the distributed nodes, supporting operational efficiency. Besides all these benefits, careful management of network connectivity, data synchronization, and fault tolerance is essential since federated systems introduce new complexity, distributed nodes can act as potential failure points affecting overall system reliability.

Scalable task distribution and batch processing capabilities are a vital connection between computational demand and system performance. Stimulated by intelligent algorithms, they enable real-time autonomous scheduling of computing tasks across the federated infrastructure e.g., AI processing, data analytics, or batch job execution based on energy availability and system capacity. In that way, it maintains computational efficiency and resource optimization without any manual intervention from operators. This system enabler serves both **environmental and economic goals** by reducing energy waste and encouraging optimal resource utilization. However, it is possible to have scheduling conflicts or resource allocation failures in upstream task management systems.



System Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Analytics for efficiency and environmental footprint	Environmental sustainability (Addressing SDG#13: Climate Action)	Real-Time energy management that can lead to carbon footprint reduction Data overload and complexity
Federated edge nodes and a private cloud located directly at renewable energy premises	Economical sustainability and innovation (Addressing SDG#8: Decent Work and Economic Growth)	Lower initial capital investment and efficient resource coordination Increased maintenance cost as well as increased bandwidth cost and network bandwidth limitation
Federated edge nodes and a private cloud located directly at renewable energy premises	Societal sustainability (Addressing SDG#3: Good Health and Well-being, SDG#11: Sustainable cities & Communities)	Reduction in local air pollution as well as reduced greenhouse effect Physical security threats may arise for edge nodes located in containers, as these typically have less strict security measures compared to large data centers.

Figure 27: System enablers of Pilot 2

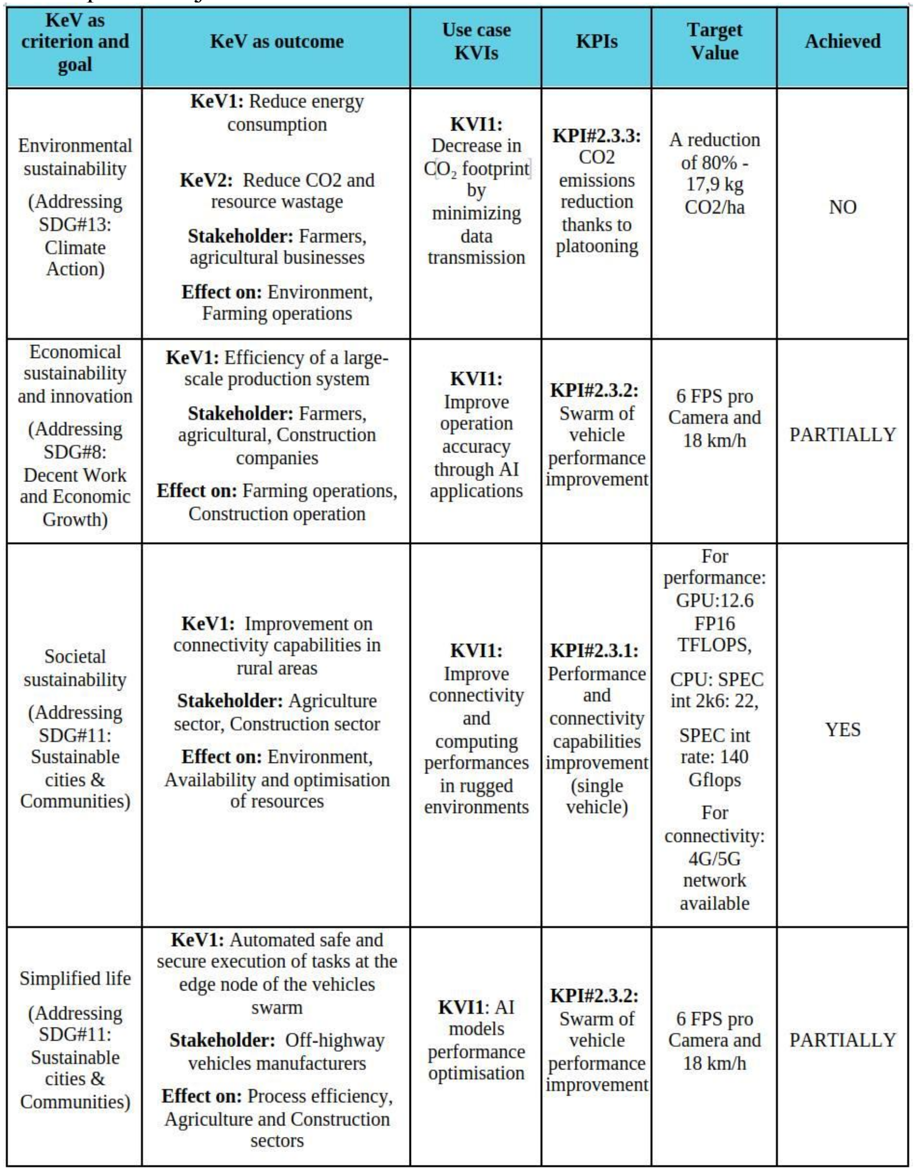
Lastly, comprehensive monitoring and analytics infrastructure is a very important aspect especially in case of achieving long-term operational sustainability in federated edge computing deployments. This system enabler supports the centralized monitoring and distributed analysis of system performance, energy consumption, and **environmental as well as societal impact**. The system has the ability to provide detailed insights effectively over time without exponential monitoring overhead. On the other hand, initial capital investments for monitoring platform setup, analytics tools integration, and comprehensive logging systems can be significant.

7.1.3. Pilot 3 High performance computing platform for connected and cooperative mobile machinery

Use case: High Performance Computing Platform for Connected and Cooperative Mobile Machinery to improve CO2 footprint

As detailed in D5.5, Pilot 3 focuses on the introducing advanced digital technology in agriculture for reducing resource consumption and increasing efficiency, thus providing overall environmental benefits. In particular, the pilot addresses the challenges related to lack of stable connectivity in rural areas by leveraging edge computing and combined with limited network infrastructures such as 4G and 5G, to enable real-time control of farming operations without relying on constant cloud access. This is especially important for coordinating and optimizing tractor activities, contributing to more efficient and sustainable farming. Existing systems, such as networked and collaborative agricultural devices, often struggle with challenges like limited data access, processing capacity, data protection, and consistent cloud availability. In-vehicle edge units (e.g., TTC edge platform), working together with other smart sensors and devices, communication modules, and the broader compute continuum, will be enhanced by the deployment of the MetaOS framework that supports secure and efficient automation.

Pilot 3 addresses **environmental sustainability** by focusing on reducing energy consumption, CO_2 emissions, and overall resource use in smart farming operations. By executing AI analytics directly at the edge,closer to the machinery, latency and reaction times are minimized, enabling faster and more efficient decision-making. A key enabler is the use of platooning, where multiple vehicles operate in coordinated formations to optimize fuel efficiency and reduce emissions. As part of the pilot, preliminary AI applications have been tested to detect weed in the fields using camera-based analytics, allowing for targeted spraying. This approach significantly reduces pesticide use and shortens tractor operation time, contributing to lower CO₂ emissions. These outcomes are directly linked to **KPI#2.3.3**, which measures emission reductions achieved through collaborative vehicle strategies. Overall, the pilot demonstrates how digital technologies can reduce CO₂ emissions by up to 40% for both diesel and electric tractors operating in swarm configurations, supporting the EU Green Deal and sustainable rural development objectives.



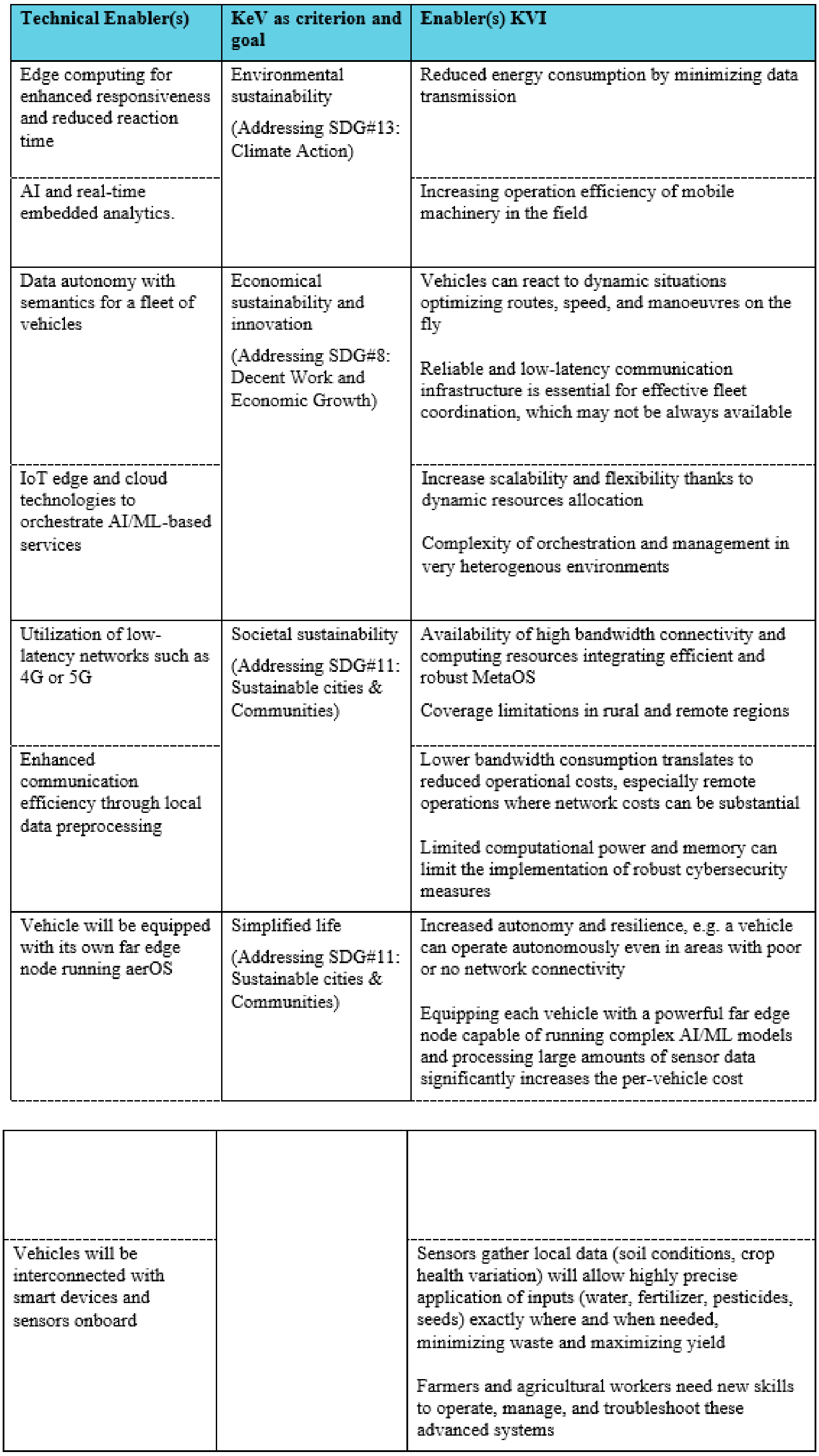
KeV as criterion and goal	KeV as outcome	Use case KVIs	KPIs	Target Value	Achieved
Environmental sustainability (Addressing SDG#13: Climate Action)	**KeV1:** Reduce energy consumption **KeV2:** Reduce CO2 and resource wastage **Stakeholder:** Farmers, agricultural businesses **Effect on:** Environment, Farming operations	**KVI1:** Decrease in CO_2 footprint by minimizing data transmission	**KPI#2.3.3:** CO2 emissions reduction thanks to platooning	A reduction of 80% - 17,9 kg CO2/ha	NO
Economical sustainability and innovation (Addressing SDG#8: Decent Work and Economic Growth)	**KeV1:** Efficiency of a large-scale production system **Stakeholder:** Farmers, agricultural, Construction companies **Effect on:** Farming operations, Construction operation	**KVI1:** Improve operation accuracy through AI applications	**KPI#2.3.2:** Swarm of vehicle performance improvement	6 FPS pro Camera and 18 km/h	PARTIALLY
Societal sustainability (Addressing SDG#11: Sustainable cities & Communities)	**KeV1:** Improvement on connectivity capabilities in rural areas **Stakeholder:** Agriculture sector, Construction sector **Effect on:** Environment, Availability and optimisation of resources	**KVI1:** Improve connectivity and computing performances in rugged environments	**KPI#2.3.1:** Performance and connectivity capabilities improvement (single vehicle)	For performance: GPU:12.6 FP16 TFLOPS, CPU: SPEC int 2k6: 22, SPEC int rate: 140 Gflops For connectivity: 4G/5G network available	YES
Simplified life (Addressing SDG#11: Sustainable cities & Communities)	**KeV1:** Automated safe and secure execution of tasks at the edge node of the vehicles swarm **Stakeholder:** Off-highway vehicles manufacturers **Effect on:** Process efficiency, Agriculture and Construction sectors	**KVI1:** AI models performance optimisation	**KPI#2.3.2:** Swarm of vehicle performance improvement	6 FPS pro Camera and 18 km/h	PARTIALLY

Figure 28: KVI analysis of Pilot 3

Pilot 3 also supports **economic sustainability and innovation** by enhancing the efficiency and scalability of large-scale agricultural production systems. The capability to orchestrate the execution of AI analytics across the IoT-edge-Cloud continuum, enables real-time decision-making and automation in the field. The proposed application involves the use of deep neural networks to process images from the cameras mounted on the tractor. These AI models are trained to identify weed infested areas in real-time in order to optimize spraying paths and reduce the amount of pesticide that is used. The efficiency of the AI algorithms is measured in frame per second and reflects the performance of AI services executed at the edge on robust hardware platforms operating in remote and demanding environments. By leveraging the aerOS framework, the **KPI#2.3.2** related to performance improvement at the edge has been achieved and the initial FPS rate has been increased by more than 20% at field operating speed of 20km/h.

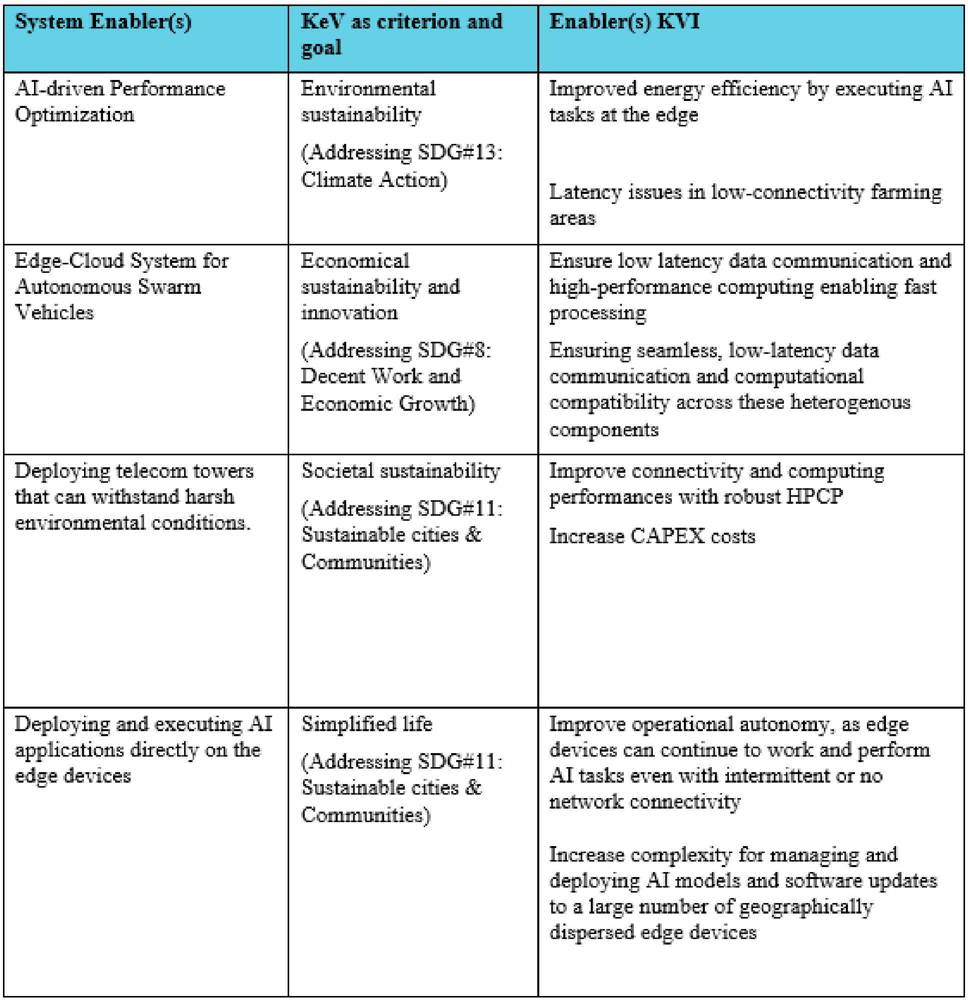
The pilot supports **societal sustainability** by enhancing connectivity capabilities in rural agricultural areas, where network infrastructure is often limited. Through the deployment of low-latency networks such as 4G and 5G, combined with data preprocessing techniques, the pilot improves communication efficiency and enables real-time responsiveness in smart farming operations. Each vehicle is equipped with high-performance computing platform with improved performances such as GPU: 12.6 FP16 TFLOPS and CPU: SPECint 2006 score of 22, SPECint rate of 140 GFLOPS. This upgrade allows for advanced processing directly on the machine, even in demanding field conditions. These capabilities demonstrate the achievement of **KPI#2.3.1**, which measures performance and connectivity improvements at the single-vehicle level.

By ensuring reliable, high-speed communication and processing in remote environments, we support bridging the digital divide, empowering also rural communities, and aligning with EU goals for a more inclusive digital transformation. Lastly, the pilot aims to simplify life in sustainable cities and communities by enabling automated, safe, and secure execution of tasks directly at the edge of vehicle swarms. Each vehicle is equipped with its own far-edge node running aerOS framework, allowing for decentralized intelligence and autonomous operation in complex agricultural environments. The vehicles are interconnected with onboard smart devices and sensors, forming a responsive and adaptive system capable of executing advanced automation operations without relying on constant cloud connectivity. This setup contributes to reducing human intervention and enhances safety, reliability and efficiency of rural production systems. The impact is measured through **KPI#2.3.2**, which tracks performance improvements that have been achieved by executing the AI analytics at the edge.



Technical Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Edge computing for enhanced responsiveness and reduced reaction time	Environmental sustainability (Addressing SDG#13: Climate Action)	Reduced energy consumption by minimizing data transmission
AI and real-time embedded analytics.		Increasing operation efficiency of mobile machinery in the field
Data autonomy with semantics for a fleet of vehicles	Economical sustainability and innovation (Addressing SDG#8: Decent Work and Economic Growth)	Vehicles can react to dynamic situations optimizing routes, speed, and manoeuvres on the fly Reliable and low-latency communication infrastructure is essential for effective fleet coordination, which may not be always available
IoT edge and cloud technologies to orchestrate AI/ML-based services		Increase scalability and flexibility thanks to dynamic resources allocation Complexity of orchestration and management in very heterogenous environments
Utilization of low-latency networks such as 4G or 5G	Societal sustainability (Addressing SDG#11: Sustainable cities & Communities)	Availability of high bandwidth connectivity and computing resources integrating efficient and robust MetaOS Coverage limitations in rural and remote regions
Enhanced communication efficiency through local data preprocessing		Lower bandwidth consumption translates to reduced operational costs, especially remote operations where network costs can be substantial Limited computational power and memory can limit the implementation of robust cybersecurity measures
Vehicle will be equipped with its own far edge node running aerOS	Simplified life (Addressing SDG#11: Sustainable cities & Communities)	Increased autonomy and resilience, e.g. a vehicle can operate autonomously even in areas with poor or no network connectivity Equipping each vehicle with a powerful far edge node capable of running complex AI/ML models and processing large amounts of sensor data significantly increases the per-vehicle cost
Vehicles will be interconnected with smart devices and sensors onboard		Sensors gather local data (soil conditions, crop health variation) will allow highly precise application of inputs (water, fertilizer, pesticides, seeds) exactly where and when needed, minimizing waste and maximizing yield Farmers and agricultural workers need new skills to operate, manage, and troubleshoot these advanced systems

Figure 29: Technical enablers of Pilot 3



System Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
AI-driven Performance Optimization	Environmental sustainability (Addressing SDG#13: Climate Action)	Improved energy efficiency by executing AI tasks at the edge Latency issues in low-connectivity farming areas
Edge-Cloud System for Autonomous Swarm Vehicles	Economical sustainability and innovation (Addressing SDG#8: Decent Work and Economic Growth)	Ensure low latency data communication and high-performance computing enabling fast processing Ensuring seamless, low-latency data communication and computational compatibility across these heterogenous components
Deploying telecom towers that can withstand harsh environmental conditions.	Societal sustainability (Addressing SDG#11: Sustainable cities & Communities)	Improve connectivity and computing performances with robust HPCP Increase CAPEX costs
Deploying and executing AI applications directly on the edge devices	Simplified life (Addressing SDG#11: Sustainable cities & Communities)	Improve operational autonomy, as edge devices can continue to work and perform AI tasks even with intermittent or no network connectivity Increase complexity for managing and deploying AI models and software updates to a large number of geographically dispersed edge devices

Figure 30: System enablers of Pilot 3

7.1.4. Pilot 4 Smart edge services for the port continuum

Use case: Smart edge services for the Port Continuum

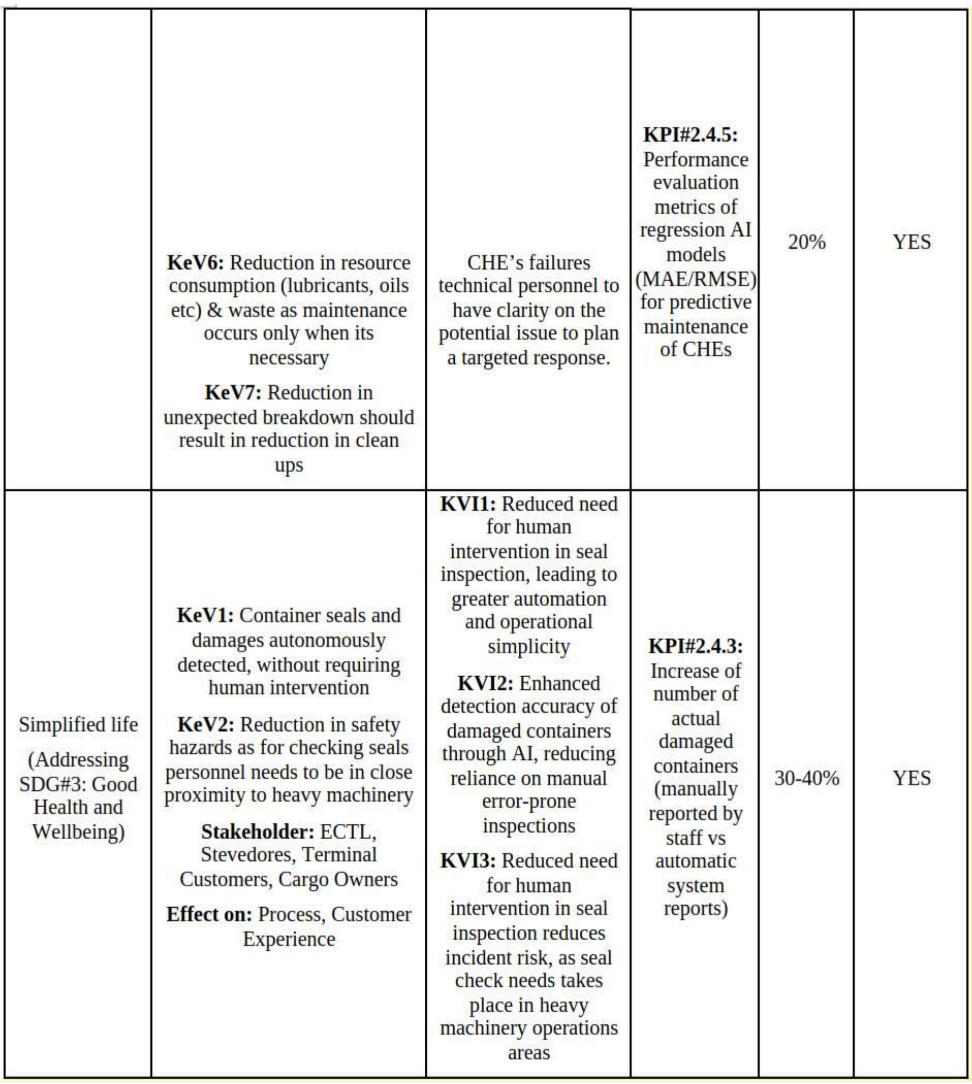
Pilot 4 aims to enhance cargo operations at EGCTL through smart edge services for the Port Continuum. Currently, Quay and Yard cranes at EGCTL rely on multiple PLC controllers, which are the most accurate source of data on crane status. However, Big Data, AI/ML, and IoT technologies are primarily based on remote servers or cloud platforms, creating a gap between the precise data from PLCs and the KPIs used for analysis and predictions. This gap results in real-time observability challenges and latency issues, hindering terminal efficiency and potentially causing operational disruptions. As physical expansion of terminals is not feasible, improving operational performance necessitates adopting the Industry 4.0 digitalization paradigm. This approach enhances decision-making by improving information availability and presentation. While first-generation IoT architectures cannot support advanced computer vision and predictive maintenance services directly at the edge, aerOS enables the orchestration of smart edge services. This allows maritime companies to react more quickly without relying on high-performance cloud processing. Not only from an operational perspective, but also by embracing digital transformation, port terminals like EUROGATE Container Terminal Limassol aim to reduce human error, and foster a safer working environment, thereby aligning with global standards and Sustainable Development Goals (SDGs).

One of the key value drivers of the pilot and for EGCTL is the enhancement of **personal health and protection** from harm, directly addressing SDG#3: Good Health and Wellbeing. By implementing AI-based predictive analytics, maintenance teams can make faster and more accurate decisions, thereby reducing human errors and associated safety risks (KeV1). This proactive approach ensures that machinery undergoes maintenance precisely when needed, minimizing unexpected failures and incidents (KeV2). The anticipated outcomes included KPIs such as 20-30% reduction in equipment idle time due to failures or 30-40% increase in the detection of equipment malfunctions through automated systems compared to 2023 levels.

Economic sustainability and innovation are also central to the terminal's digitalization strategy, aligning with SDG#9: Industry, Innovation, and Infrastructure. The adoption of AI-enabled early fault detection and optimized scheduling is expected to reduce operational (KeV4) and maintenance costs (KeV6). Furthermore, the integration of AI models for predictive maintenance of Container Handling Equipment (CHEs) aims to extend their lifespan (KeV3), a higher asset utilization in operation (KeV5). To do so, it was set that at least 80% precision and F1-score performance metrics shall be guaranteed. Additionally, the implementation of classification AI models for damaged seals is targeted to reach 60% accuracy and F1 scores, enhancing the reliability and efficiency of operations.

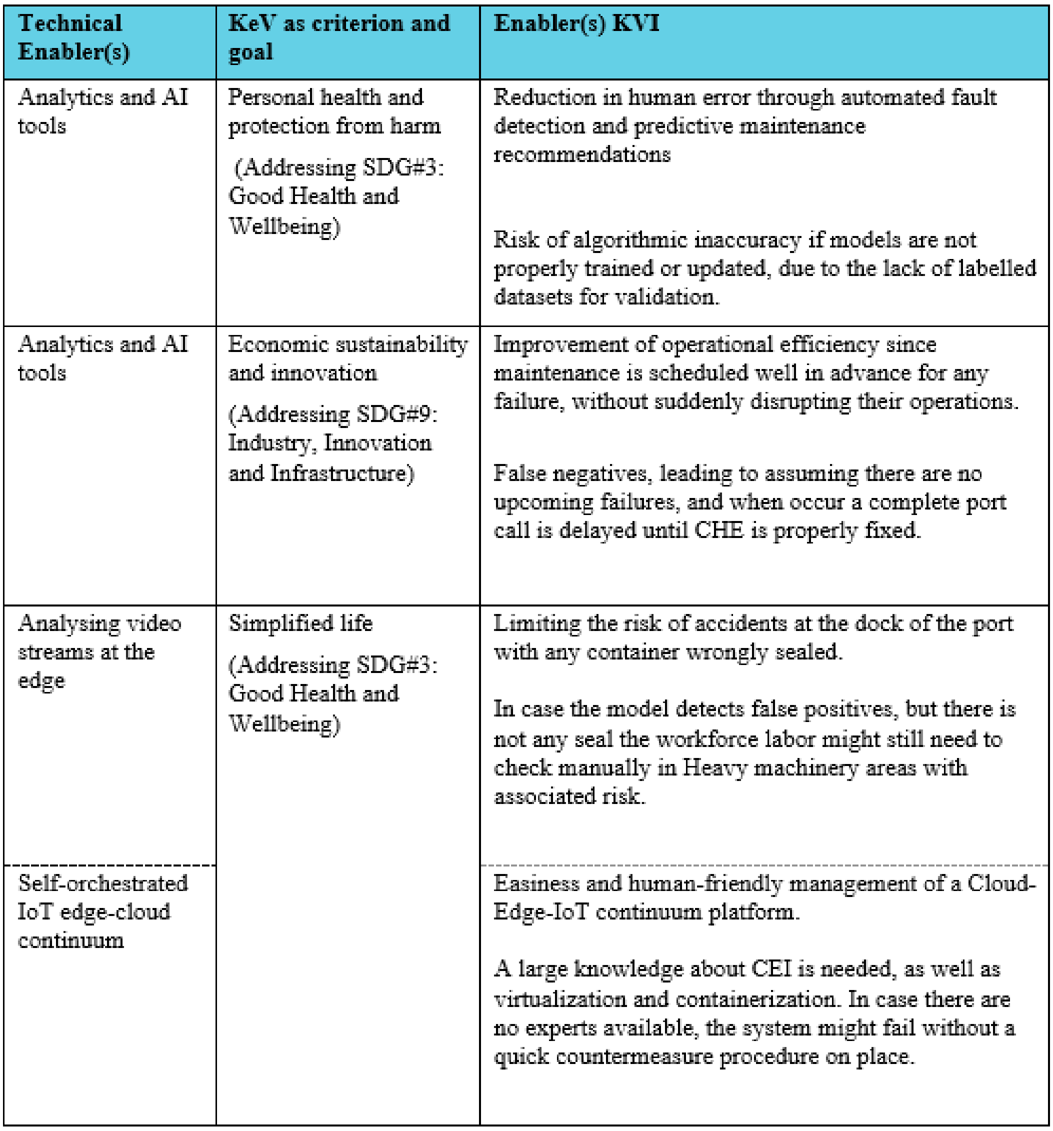
Finally, the terminal plans to implement autonomous container seal inspections, simplifying operational processes, contributing also to **Simplified life** and to SDG#3: Good Health and Wellbeing. This scenario is helping on reducing the need for human intervention (KeV7), and thereby decreasing safety hazards associated with personnel working in close proximity to heavy machinery (KeV8). This automation was achieved by fulfilling **KPI-2.4.3**, leading to a 30-40% increase in the detection of actual damaged containers, comparing manual reports by staff to automatic system reports.

KeV as criterion and goal	KeV as outcome	Use case KVIs	KPIs	Target Value	Achieved
Personal health and protection from harm (Addressing SDG#3: Good Health and Wellbeing)	**KeV1:** Maintenance team takes faster and better decisions, lowering human mistakes and safety risks **KeV2:** Machine comes for maintenance only when its required reducing occurrences and incident risk **Stakeholder:** EGCTL, Technical Department workforce, Container terminal Customers, Users **Effect on:** Operating & Maintenance, Repair Process	**KVI1:** Reduction in the risk of health and safety incidents related to unplanned equipment failures and inefficient manual interventions **KVI2:** Improved decision-making speed and accuracy for maintenance teams through AI-based predictive analytics	**KPI#2.4.1:** Reduction of CHE idle time due to failures	20-30%	YES
			KPI#2.4.2: Increase on detection of equipment malfunctions (from manual to automatic)	30-40% with respect to 2023	YES
			KPI#2.4.5: Performance evaluation metrics of regression AI models (MAE/RMSE) for predictive maintenance of CHEs	20%	YES
Economic sustainability and innovation (Addressing SDG#9:	**KeV1:** Reduction of cost and time in decision making **KeV2:** Early detection of issues	**KVI1:** Reduced operational and maintenance costs through early AI-enabled fault	**KPI#2.4.1:** Reduction of CHE idle time due to failures	20-30%	YES



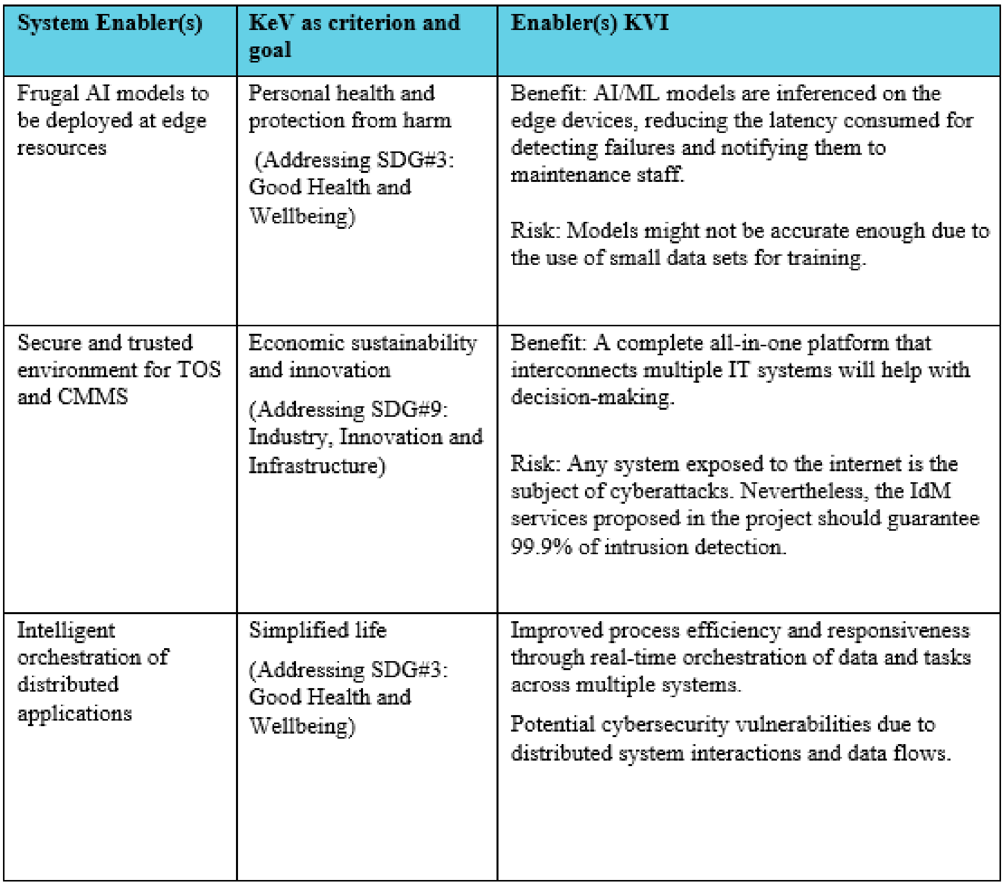
	KeV6: Reduction in resource consumption (lubricants, oils etc) & waste as maintenance occurs only when its necessary **KeV7:** Reduction in unexpected breakdown should result in reduction in clean ups	CHE's failures technical personnel to have clarity on the potential issue to plan a targeted response.	**KPI#2.4.5:** Performance evaluation metrics of regression AI models (MAE/RMSE) for predictive maintenance of CHEs	20%	YES
Simplified life (Addressing SDG#3: Good Health and Wellbeing)	**KeV1:** Container seals and damages autonomously detected, without requiring human intervention **KeV2:** Reduction in safety hazards as for checking seals personnel needs to be in close proximity to heavy machinery **Stakeholder:** ECTL, Stevedores, Terminal Customers, Cargo Owners **Effect on:** Process, Customer Experience	**KVI1:** Reduced need for human intervention in seal inspection, leading to greater automation and operational simplicity **KVI2:** Enhanced detection accuracy of damaged containers through AI, reducing reliance on manual error-prone inspections **KVI3:** Reduced need for human intervention in seal inspection reduces incident risk, as seal check needs takes place in heavy machinery operations areas	**KPI#2.4.3:** Increase of number of actual damaged containers (manually reported by staff vs automatic system reports)	30-40%	YES

Figure 31: KVI analysis of Pilot 4



Technical Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Analytics and AI tools	Personal health and protection from harm (Addressing SDG#3: Good Health and Wellbeing)	Reduction in human error through automated fault detection and predictive maintenance recommendations Risk of algorithmic inaccuracy if models are not properly trained or updated, due to the lack of labelled datasets for validation.
Analytics and AI tools	Economic sustainability and innovation (Addressing SDG#9: Industry, Innovation and Infrastructure)	Improvement of operational efficiency since maintenance is scheduled well in advance for any failure, without suddenly disrupting their operations. False negatives, leading to assuming there are no upcoming failures, and when occur a complete port call is delayed until CHE is properly fixed.
Analysing video streams at the edge	Simplified life (Addressing SDG#3: Good Health and Wellbeing)	Limiting the risk of accidents at the dock of the port with any container wrongly sealed. In case the model detects false positives, but there is not any seal the workforce labor might still need to check manually in Heavy machinery areas with associated risk.
Self-orchestrated IoT edge-cloud continuum		Easiness and human-friendly management of a Cloud-Edge-IoT continuum platform. A large knowledge about CEI is needed, as well as virtualization and containerization. In case there are no experts available, the system might fail without a quick countermeasure procedure on place.

Figure 32: Technical enablers of Pilot 4



System Enabler(s)	KeV as criterion and goal	Enabler(s) KVI
Frugal AI models to be deployed at edge resources	Personal health and protection from harm (Addressing SDG#3: Good Health and Wellbeing)	Benefit: AI/ML models are inferenced on the edge devices, reducing the latency consumed for detecting failures and notifying them to maintenance staff. Risk: Models might not be accurate enough due to the use of small data sets for training.
Secure and trusted environment for TOS and CMMS	Economic sustainability and innovation (Addressing SDG#9: Industry, Innovation and Infrastructure)	Benefit: A complete all-in-one platform that interconnects multiple IT systems will help with decision-making. Risk: Any system exposed to the internet is the subject of cyberattacks. Nevertheless, the IdM services proposed in the project should guarantee 99.9% of intrusion detection.
Intelligent orchestration of distributed applications	Simplified life (Addressing SDG#3: Good Health and Wellbeing)	Improved process efficiency and responsiveness through real-time orchestration of data and tasks across multiple systems. Potential cybersecurity vulnerabilities due to distributed system interactions and data flows.

Figure 33: System enablers of Pilot 4

7.1.5. Pilot 5 Energy Efficient, health safe and sustainable smart buildings

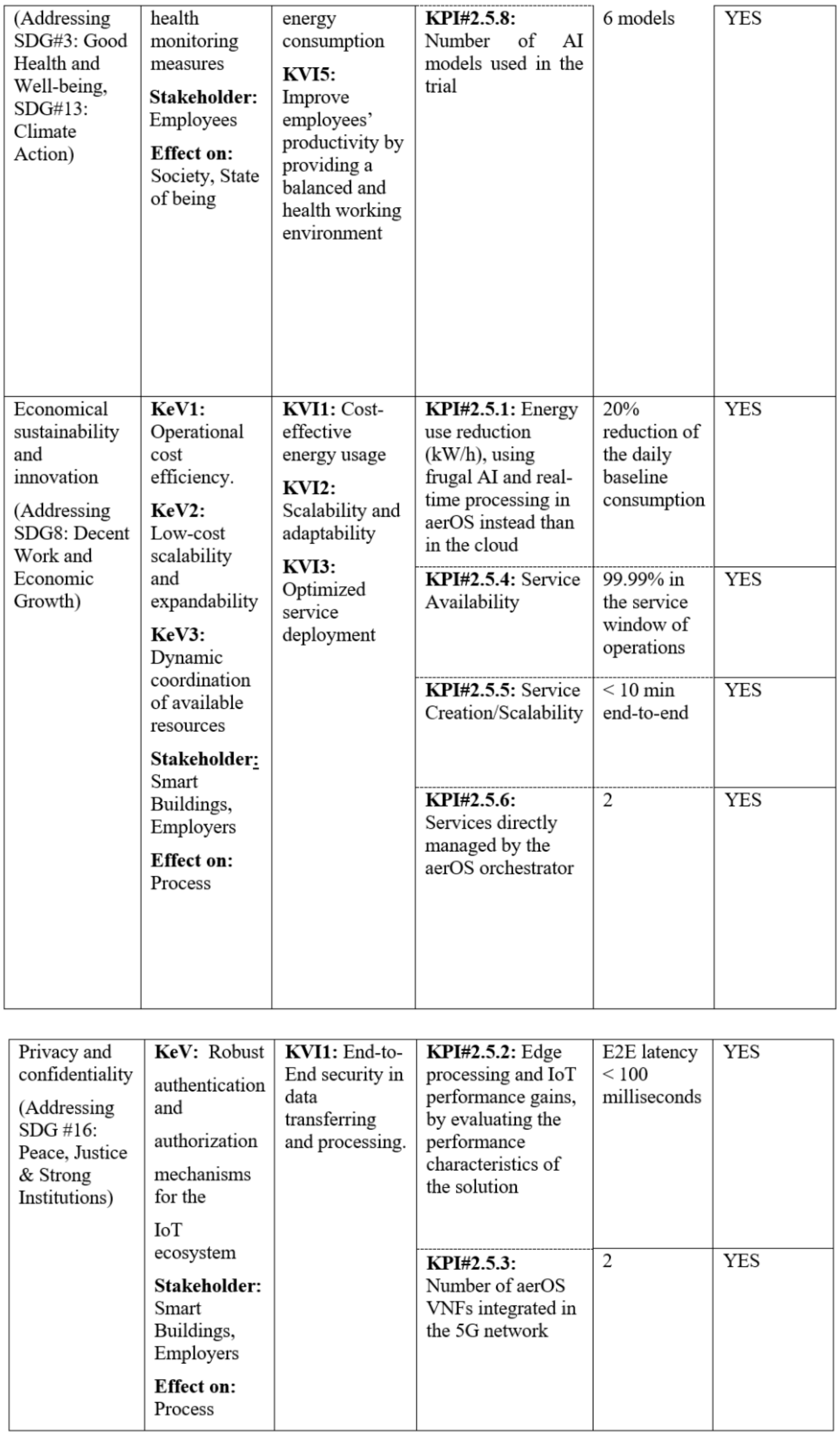
Use case: Energy Efficient, Health Safe & Sustainable Smart Buildings

As it was described in Deliverable 5.5, Pilot 5 addresses the complex challenges created by the COVID-19 pandemic in the context of smart buildings, where traditional methods such as maximizing workspace occupancy are no longer acceptable without careful attention to safety, health, and energy efficiency. The pilot demonstrates the use of the aerOS platform, integrating real-time data processing, AI, 5G, and IoT technologies to autonomously optimize building operations. It delivers measurable improvements across multiply sustainability domains and beyond: environmental, societal and personal health, economic, and data privacy.

The pilot contributes to **environmental sustainability** through the reduction of energy consumption, resource wastage and enhancing real-time edge processing capabilities. The KeVs under this domain include energy efficiency and processing responsiveness. The KVIs measure operational aspects such as working area energy consumption, utilization efficiency in far-edge nodes, and reductions in communication latency. Performance is measured through **KPI#2.5.1** achieving a 20% reduction in daily energy consumption via frugal AI and edge processing. In addition**, KPI#2.5.2** measures the communication latency at the edge with a goal of staying below 100 milliseconds, highlighting improvements in processing speed (20% less memory utilisation) and service resilience (90% increase in recovery time).

Societal sustainability focuses on improving the health, well-being, and productivity of employees within smart building environments. Also, the domain of Personal health and protection from environmental hazards is addressed by the same KeVs, KVIs and KPIs. This includes ensuring air quality, minimizing health-related workplace disruptions, and supporting a balanced working environment. The KeVs include reduced health incidents, maintaining air quality, and deploying health monitoring systems. KVIs track metrics such as an improvement in air quality, decreases in disruptions, health monitoring index, CO2 emission cuts, and improvements in productivity. This plot adopts **KPI#2.5.1** from the environmental sector, citing once more the 20% energy consumption drop. Besides, **KPI#2.5.7** ensures air quality by restricting CO_2 concentration to below 600 ppm, a threshold compatible with health safety policies and **KPI#2.5.8** assessing seven AI models deployment during the trial to ensure adaptive and intelligent monitoring of the environment.

KeV as criterion and goal	KeV as outcome	Use case KVIs	KPIs	Target Value	Achieved
Environmental sustainability (Addressing SDG#13: Climate Action)	**KeV1:** Reduce energy consumption and resource wastage **KeV2:** Accurate and quick edge processing **Stakeholder:** Smart Buildings, Employers **Effect on:** Environment	**KVI1:** Working area energy consumption **KVI2:** Efficient resource utilization in far-edge nodes **KVI3:** Reduction in communication latency	**KPI#2.5.1:** Energy use reduction (kW/h), using frugal AI and real-time processing in aerOS instead than in the cloud	20% reduction of the daily baseline consumption	YES
			KPI#2.5.2: Edge processing and IoT performance gains, by evaluating the performance characteristics of the solution	E2E latency < 100 milliseconds	YES
Societal sustainability (Addressing SDG#3: Good Health and Well-being, SDG#11: Sustainable cities & Communities)	**KeV1:** Reduced health incidents in workplaces **KeV2:** Maintain air quality levels **KeV3:** Reduce energy consumption and resource wastage **KeV4:** Employ	**KVI1:** Improvement of air quality **KVI2:** Reduction in workplace disruptions **KVI3:** Measure and monitor a Health index for the employees **KVI4:** Reduction CO2 emissions and	**KPI#2.5.1:** Energy use reduction (kW/h), using frugal AI and real-time processing in aerOS instead than in the cloud	20% reduction of the daily baseline consumption	YES
Personal health and protection from harm			**KPI#2.5.7:** Improvement of air quality	Max CO2 lower than 600 ppm	YES



(Addressing SDG#3: Good Health and Well-being, SDG#13: Climate Action)	health monitoring measures **Stakeholder:** Employees **Effect on:** Society, State of being	energy consumption **KVI5:** Improve employees' productivity by providing a balanced and health working environment	**KPI#2.5.8:** Number of AI models used in the trial	6 models	YES
Economical sustainability and innovation (Addressing SDG8: Decent Work and Economic Growth)	**KeV1:** Operational cost efficiency. **KeV2:** Low-cost scalability and expandability **KeV3:** Dynamic coordination of available resources **Stakeholder:** Smart Buildings, Employers **Effect on:** Process	**KVI1:** Cost-effective energy usage **KVI2:** Scalability and adaptability **KVI3:** Optimized service deployment	**KPI#2.5.1:** Energy use reduction (kW/h), using frugal AI and real-time processing in aerOS instead than in the cloud	20% reduction of the daily baseline consumption	YES
			KPI#2.5.4: Service Availability	99.99% in the service window of operations	YES
			KPI#2.5.5: Service Creation/Scalability	< 10 min end-to-end	YES
			KPI#2.5.6: Services directly managed by the aerOS orchestrator	2	YES
Privacy and confidentiality (Addressing SDG #16: Peace, Justice & Strong Institutions)	**KeV:** Robust authentication and authorization mechanisms for the IoT ecosystem **Stakeholder:** Smart Buildings, Employers **Effect on:** Process	**KVI1:** End-to-End security in data transferring and processing.	**KPI#2.5.2:** Edge processing and IoT performance gains, by evaluating the performance characteristics of the solution	E2E latency < 100 milliseconds	YES
			KPI#2.5.3: Number of aerOS VNFs integrated in the 5G network	2	YES

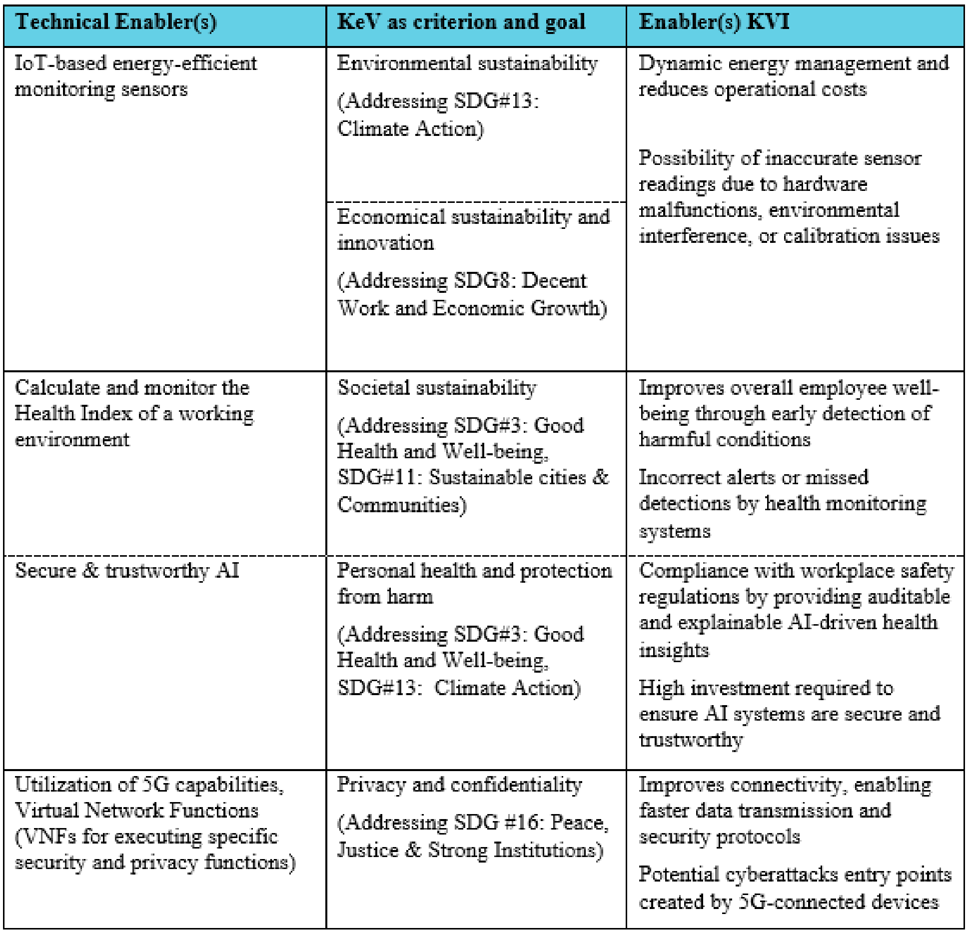
Figure 34: KVI analysis of Pilot 5

Economic sustainability focuses on creating operational efficiency, scalability, and coordination of resources in operations within buildings. KeVs in this domain are focused on enabling cost-low operations, enabling seamless scaling of services, and making deployment of digital services simple. The KVIs are all relevant to energy saving, flexibility, and maximized service orchestration. Along with reuse of **KPI#2.5.1** on energy efficiency, this category includes **KPI#2.5.4** (target: 99.99% availability of service), **KPI#2.5.5** (target: <10 minutes for deploying service), and **KPI#2.5.6**, which counts the number of services under direct control by the aerOS orchestrator with a target of three services autonomously managed.

Finally, Pilot 5 covers **Privacy and confidentiality** domain by applying strong authentication and authorization in the IoT environment. The benefit here is end-to-end security during data transfer and processing. Relevant are **KPI#2.5.2**, sustaining latency of below 100 milli seconds in order to enable safe and timely data exchange, and **KPI#2.5.3**, ensuring the integration of three aerOS Virtual Network Functions (VNFs) into the 5G network. These VNFs are contributing to as robust and secure network infrastructure.

From technical enablers perspective, the **Environmental and Economic sustainability** can be addressed through IoT-based monitoring sensors, These sensors contribute to a dynamic and energy-efficient environmental management. These sensors monitor parameters such as temperature, occupancy, and power consumption in real-time, enabling real-time energy optimization. This leads to dynamic energy management and reduction of operational costs. That can have a direct impact on lowered energy consumption and wiser utilization of resources. However, accuracy must be a priority since the possibility of hardware errors, environmental interference, or calibration errors may lead to inaccurate sensor readings, which could affect decision-making processes.

The next technical enabler is aligned with **Societal sustainability** as well as with **Personal health and protection** from harm. More specifically, calculating and continuously monitoring the Health Index of indoor environments is very important to ensure employee well-being. This involves collecting sensor data for air quality, humidity, temperature etc. to determine the suitability of a workspace. A room's health index, as a metric, reflects the overall quality of the indoor environment with regard to human comfort and safety. By combining important environmental factors including temperature, humidity, CO_2 levels, and PM into a single easily comprehensible number between 0 and 100, this score contributes to the larger objective of encouraging healthier and more energy-efficient workplaces. This can lead to an improved workplace health and well-being since it can provide early detection of harmful environmental conditions. However incorrect alerts or missed detections from health monitoring systems could lead to false alarms or a failure to act when intervention is actually required.



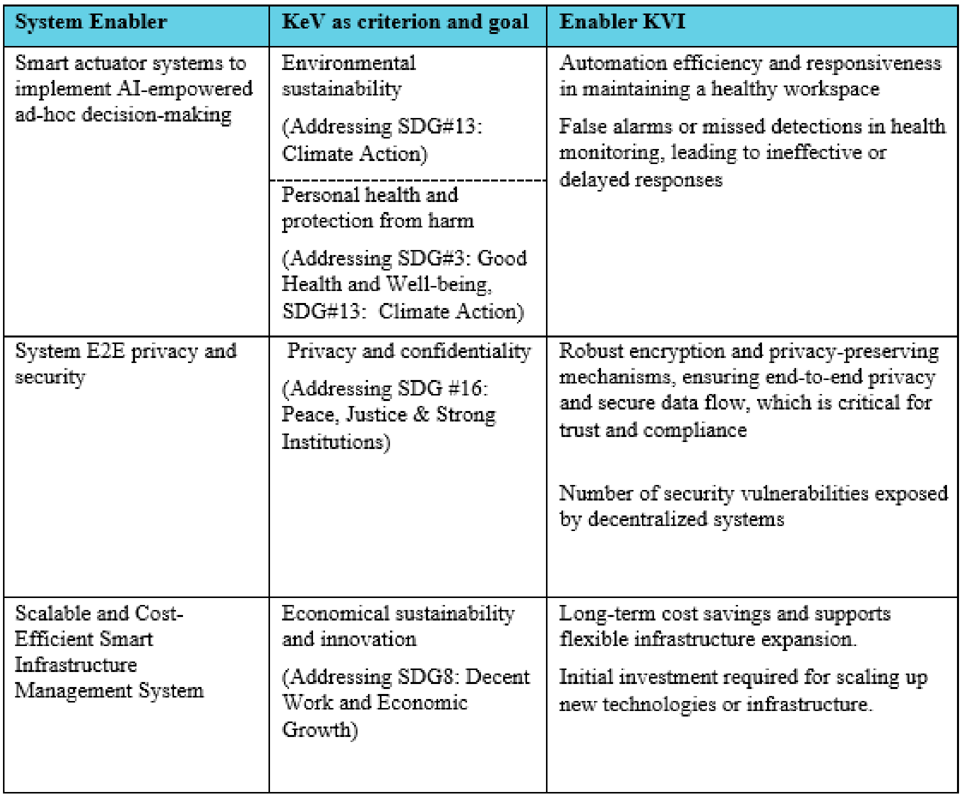
Technical Enabler(s)	**KeV as criterion and goal**	**Enabler(s) KVI**
IoT-based energy-efficient monitoring sensors	Environmental sustainability (Addressing SDG#13: Climate Action)	Dynamic energy management and reduces operational costs Possibility of inaccurate sensor readings due to hardware malfunctions, environmental interference, or calibration issues
	Economical sustainability and innovation (Addressing SDG8: Decent Work and Economic Growth)	
Calculate and monitor the Health Index of a working environment	Societal sustainability (Addressing SDG#3: Good Health and Well-being, SDG#11: Sustainable cities & Communities)	Improves overall employee well-being through early detection of harmful conditions Incorrect alerts or missed detections by health monitoring systems
Secure & trustworthy AI	Personal health and protection from harm (Addressing SDG#3: Good Health and Well-being, SDG#13: Climate Action)	Compliance with workplace safety regulations by providing auditable and explainable AI-driven health insights High investment required to ensure AI systems are secure and trustworthy
Utilization of 5G capabilities, Virtual Network Functions (VNFs for executing specific security and privacy functions)	Privacy and confidentiality (Addressing SDG #16: Peace, Justice & Strong Institutions)	Improves connectivity, enabling faster data transmission and security protocols Potential cyberattacks entry points created by 5G-connected devices

Figure 35: Technical enablers of Pilot 5

In the context of **Personal health and protection from harm**, the development of an AI algorithm has been conducted. This algorithm allows intelligent analysis of collected data and supports explainable decision-making for health and safety. Ensuring that there is a secure and trustworthy AI, which can make the choices not only correct but also traceable, it is especially critical to ensure compliance with workplace safety regulations in a compliant and transparent way. These attributes can provide assurance to stakeholders that health-related interventions are based on explainable logic. The implementation of a trustworthy AI system may come with high investment cost such as training data, cybersecurity, and system validation.

Last but not least, from the network and connectivity perspective, 5G technology and Virtual Network Functions (VNFs) can contribute to a stable, reliable and flexible connectivity in smart buildings. These networks and components allow faster data transmission and support real-time analytics at the edge. They can also enable dynamic execution of security protocols as well as improve trust among the stakeholders, supporting **Privacy & Confidentiality**. Beside all these benefits, careful management of access controls, encryption, and anomaly detection is essential since 5G introduces a new attack surface since connected devices can act as cyber-attack entry points.

Smart actuator systems are a vital connection between digital decision-making and physical action. Stimulated by AI, they enable real-time autonomous responses to changes in the building environment e.g., ventilation, lighting, or temperature control on the basis of sensor data. In that way, it maintains an indoor setting healthy and adaptable without any intervention from humans. This system enabler serves both **health and environmental** goals by reducing energy waste and encouraging occupant safeguarding. However, it is possible to have false alarms or detection failure in upstream health monitoring systems.



System Enabler	KeV as criterion and goal	Enabler KVI
Smart actuator systems to implement AI-empowered ad-hoc decision-making	Environmental sustainability (Addressing SDG#13: Climate Action) Personal health and protection from harm (Addressing SDG#3: Good Health and Well-being, SDG#13: Climate Action)	Automation efficiency and responsiveness in maintaining a healthy workspace False alarms or missed detections in health monitoring, leading to ineffective or delayed responses
System E2E privacy and security	Privacy and confidentiality (Addressing SDG #16: Peace, Justice & Strong Institutions)	Robust encryption and privacy-preserving mechanisms, ensuring end-to-end privacy and secure data flow, which is critical for trust and compliance Number of security vulnerabilities exposed by decentralized systems
Scalable and Cost-Efficient Smart Infrastructure Management System	Economical sustainability and innovation (Addressing SDG8: Decent Work and Economic Growth)	Long-term cost savings and supports flexible infrastructure expansion. Initial investment required for scaling up new technologies or infrastructure.

Figure 36: System enablers of Pilot 5

To address **Privacy and confidentiality**, a secure end-to-end system has been developed to ensure that authentication, data transmission, and processing procedures are protected through robust encryption and privacy-preserving mechanisms. Security and privacy are critical aspects in any system that collects, processes, and acts on personal or operational data. However, the number and severity of security vulnerabilities should be evaluated and reported, particularly those introduced by decentralized architectures like edge computing.

Lastly, scalability and cost-efficiency are very important aspects, especially in case of achieving long-term **Economic sustainability** in smart building deployments. This system enabler supports the centralized orchestration and decentralized deployment of smart infrastructure services. The system has the ability to scale effectively over time without exponential cost increases. However, upfront investments for platform setup, hardware upgrades, and training can be significant.

8. Conclusions

Deliverable D5.6 concludes the validation and evaluation cycle of the aerOS project by presenting all evidence from the technical, operational, and impact assessments conducted in the frame of the last project phase. Evidence from extensive cross-domain testing and pilot deployments underlines that aerOS successfully demonstrated the maturity and robustness of its Meta-Operating System approach for the Cloud-Edge-IoT continuum. Coordinated and integrated validation activities across five pilots in manufacturing, energy, agriculture, ports, and buildings confirmed that the aerOS framework enables interoperability, scalability, and secure orchestration across heterogeneous environments.

Technically, the platform reached or outperformed most of its KPIs in proving the capability of handling distributed intelligence, ensuring trust and cybersecurity, while optimizing edge-cloud resource allocation. Pilot-level KPIs further underlined the flexibility of the architecture by showing quantifiable benefits with respect to operational efficiency, energy consumption reduction, CO_2 emission mitigation, and automation. These were supplemented by the impact KPIs, which showed that significant advancement has been achieved in dissemination, standardization, and industrial commitment, confirming the contribution of aerOS to European digital domain and open innovation.

Analysis of the requirement coverage performed and the assessment of the Key Value Indicators provide good proof of the alignment between originally set project objectives and results delivered. Indeed, practically all technical and functional requirements were met, hence underlining that the resulting aerOS solution is both technically and industrially relevant. Results prove that aerOS is ready for large-scale adoption and can serve as one of the key foundational enablers for next-generation, federated, and intelligent digital ecosystems in Europe. Overall, the results presented in this deliverable provide evidence for the project vision: designing a unified intelligent and secure Meta-OS, which will be able to connect cloud and edge infrastructures in a seamless way, enabling interoperability, efficiency, and sustainability across various sectors.

References

[1] "D5.5 – Technical evaluation, validation and assessment report (v1.0)," aerOS Project Deliverable D5.5, Aug. 31, 2024. [Online]. Available: https://aeros-project.eu/wp-content/uploads/2024/09/aerOS_D5.5_v1.0-subm.pdf

[2] G. Wikström, J. Gulliksen, M. Bergvall-Kåreborn, A. Andersson, C. Hägglund, A. Grünloh, S. Tolmie, S. Couture, and D. Laurell, "Key value indicators: A framework for values-driven next-generation ICT solutions," Telecommunications Policy, vol. 48, no. 6, article no. 102778, Jul. 2024. doi: 10.1016/j.telpol.2024.102778.

A. Appendix A. Validation Activities

The following sections describe the last activities performed by each pilot. All the **Validation activities** –including KPIs—and the pending **Integration activities**, reported on the previous D5.4. Together with these activities, descriptions of the pilots and their scenarios can be found.

Pilot 1 – Data-Driven cognitive production lines

Pilot 1 is divided into 4 Sub-pilots (1.1, 1.2, 1.3 and 1.4) being the Business Process (BP) the part of the code that references it, hence BP1 for Pilot 1.1, BP2 for Pilot 1.2, BP3 for Pilot 1.3, and BP4 for Pilot 1.4.

Demonstrates how aerOS turns four heterogeneous factory scenarios into one orchestrated, secure continuum that moves intelligence to the edge, standardizes live context with NGSI-LD, and enables resilient, multi-site operations. In P1.1, the drone line was fully connected to aerOS to deliver real-time CO_2/PCF data, footprint prediction beyond target for the two main product types, 100% product coverage, and model setup time collapsing from minutes to under a second—showing tangible gains in sustainability and data transparency. P1.2 shifted metrology from manual, on-site procedures to remote, autonomous operation: dockerized OPC-UA services and self-recovery raised accuracy, cut programming time from ~10 days to just over two, and lifted throughput beyond five parts/hour while maintaining secure access and a live digital twin of each station. P1.3 proved the reuse of containerized "skills" for AGVs—autonomous docking for wireless charging and coordination with robot-arm cells—lifting availability to 96% and usage to 84% without major hardware changes. P1.4 extended optimization across sites via federated orchestration and shared context, reducing AGV trips by ~39% and CO_2 emissions by ~39.4% for the validated flow. Collectively, Pilot 1 evidences measurable gains in sustainability, accuracy, efficiency, and productivity, validating aerOS as the connective tissue for cognitive production lines.

PILOT 1		2024											2025								
Code	Name	M19	M20	M21	M22	M23	M24	M25	M26	M27	M28	M29	M30	M31	M32	M33	M34	M35	M36	M37	M38
Pilot 1.1 - (Business Process 1) - Green Manufacturing and CO2 Footprint Monitoring		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Setup & Procurement Activities		■	■	■	■	■															
P1-BP1-SA1	Stations identification for the trial	■																			
P1-BP1-SA2	Hardware setup	■	■	■	■																
P1-BP1-SA3	Equipment configuration		■	■	■	■															
Development Activities		■	■	■	■	■															
P1-BP1-DA4	Definition of IT architecture	■	■	■	■	■															
P1-BP1-DA5	Communication infrastructure developed or adapted	■	■	■	■	■															

P1-BP1-DA6	APIs setup	■	■	■	■	■															
P1-BP1-DA7	Software configuration / development	■	■	■	■	■															
Integration Activities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
P1-BP1-IA8	Identification of place of deployment in current architecture	■	■	■	■																
P1-BP1-IA9	Requirements definition for deployment	■	■	■	■																
P1-BP1-IA10	Test energy data collection					■	■	■	■												
P1-BP1-IA11	Adjust equipment configuration according to first feedback									■	■	■									
P1-BP1-IA12	aerOS Basic components				■	■	■	■	■												
P1-BP1-IA13	aerOS Non Basic components											■	■	■	■	■					
P1-BP1-IA14	Dashboard creation with energy data												■	■	■	■					
P1-BP1-IA15	Integration of data analysis service for reports and statistics creation												■	■	■	■	■	■	■		
P1-BP1-IA16	Creation of a future prediction of the CO2 footprint for each product												■	■	■	■					
P1-BP1-IA17	Experimenting with Gaia-X services												■	■	■	■	■	■	■		
Validation Activities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
P1-BP1-VA18	Data quality verification								■	■	■	■	■	■	■	■	■	■	■		
P1-BP1-VA19	Digital Product Passport implementation										■	■	■	■	■	■	■	■	■		
P1-BP1-VA20	Improvement activities										■	■	■	■	■	■	■	■	■		
P1-BP1-VA21	KPIs validation															■	■	■	■		
P1-BP1-VA22	Qualitative validation															■	■	■	■		
P1-BP1-VA23	Evaluation and reporting																■	■	■	■	■
Pilot 1.2 - (Business Process 2) - Automotive Smart factory Zero defect manufacturing		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Setup & Procurement Activities		■	■	■																	
P1-BP2-SA1	Installation of control camera in the CMM	■	■	■																	
P1-BP2-SA2	Installation of pressure sensor in the CMM	■	■	■																	
Development Activities		■	■	■	■	■															
P1-BP2-DA5	Enable Software-based control services and IoT hub for collection and brokering for instrumentation information	■	■	■	■	■															
Integration Activities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
P1-BP2-IA6	aerOS Components								■	■	■	■	■	■	■	■	■	■	■		
Validation Activities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

P1-BP2-VA7	Remote configuration/set-up of the CMM instrumentation robotic and kinematic configuration														🟦	🟦	🟦	🟦	🟦		
P1-BP2-VA8	Remote tactile operation of CMM															🟦	🟦	🟦	🟦	🟦	
P1-BP2-VA9	aerOS assist and optimize the process of Digital Twin creation													🟦	🟦	🟦	🟦	🟦	🟦	🟦	
P1-BP2-VA10	Dynamic execution of metrology services and Data assembling												🟦	🟦	🟦	🟦	🟦	🟦	🟦		
P1-BP2-VA11	KPI 2.1.1: Production process accuracy >10% baseline																		🟦	🟦	🟦
P1-BP2-VA12	KPI 2.1.2: Digital service programming time 2 days																		🟦	🟦	🟦
P1-BP2-VA13	KPI 2.1.3: Dimensional quality control productivity 5parts/hr																		🟦	🟦	🟦
Pilot 1.3 - (Business Process 3) - Zero Ramp-up safe PLC reconfiguration for Lot-Size-1 Production		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	
Setup & Procurement Activities		■	■	■	■	■	■	■	■	■	■	■									
P1-BP3-SA3	Procurement and setup of 2 robot arm modules	🟦	🟦	🟦	🟦	🟦	🟦	🟦													
P1-BP3-SA5	Setup Tech Hall Network	🟦	🟦	🟦																	
P1-BP3-SA6	Procurement camera for robot arm	🟦	🟦	🟦	🟦	🟦	🟦	🟦													
P1-BP3-SA7	Setup AI camera detection	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦									
P1-BP3-SA8	Procurement and setup gripper for robot arm	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦									
Development Activities		■	■	■	■	■	■	■	■	■	■	■	■								
P1-BP3-DA2	AGV Fleet Manager	🟦	🟦	🟦																	
P1-BP3-DA3	Device simplified programming over low code tools	🟦	🟦	🟦	🟦	🟦	🟦	🟦													
P1-BP3-DA5	Siemens Industrial Edge configuration								🟦	🟦	🟦	🟦	🟦								
P1-BP3-DA6	Integration of TSN in a domain							🟦	🟦	🟦	🟦	🟦									
Integration Activities									■	■	■	■	■	■	■	■	■	■	■	■	
P1-BP3-IA2	Custom LLO for Siemens Industrial Edge device												🟦	🟦	🟦	🟦	🟦	🟦			
P1-BP3-IA3	Communicate domains using open ziti														🟦	🟦	🟦				
P1-BP3-IA4	aerOS Basic components								🟦	🟦	🟦	🟦	🟦	🟦							
P1-BP3-IA5	aerOS Non Basic components													🟦	🟦	🟦	🟦	🟦			
P1-BP3-IA6	Definition of data federation – and achieving such federation																	🟦	🟦	🟦	
Validation Activities																		■	■	■	
P1-BP3-VA1	KPI 2.1.7 validation: AGV usage > 80 %																	🟦	🟦	🟦	
P1-BP3-VA2	KPI 2.1.8 validation: AGV availability > 95%																	🟦	🟦	🟦	

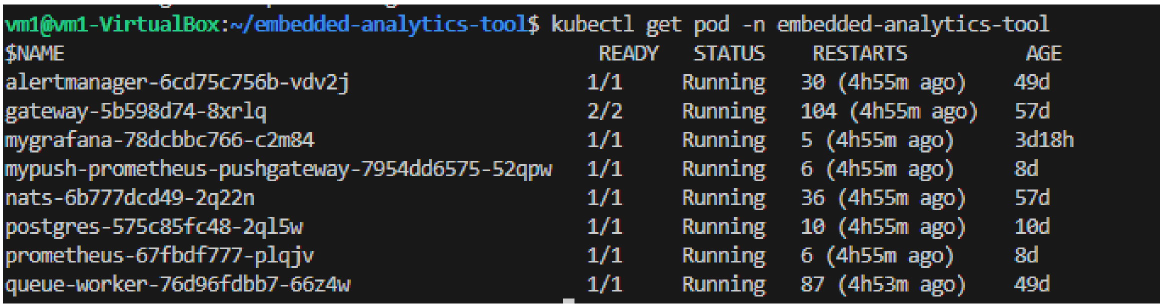
Pilot 1.4 - (Business Process 4) - AGV Swarm Zero break-down logistics for Lot-Size-1 Production																					
Setup & Procurement Activities		■	■	■	■	■	■	■	■	■											
P1-BP4-SA1	Upgrade/Update POLIMI AGV	▨																			
P1-BP4-SA2	Definition of IT architecture with aerOS	▨	▨	▨	▨	▨	▨	▨	▨	▨											
P1-BP4-SA3	Procurement and setup of Raspberry Pi POLIMI	▨																			
P1-BP4-SA4	Procurement and setup of Raspberry Pi MADE	▨																			
P1-BP4-SA5	Setup of Domain Infrastructure	▨	▨	▨	▨	▨	▨	▨	▨	▨											
P1-BP4-SA6	Setup of Dedicated network for MADE Domain	▨																			
Development Activities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■			
P1-BP4-DI1	Development/containerisation of overlay Order management application at MADE	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨										
P1-BP4-DI2	Development/containerisation of overlay order management at POLIMI	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨										
P1-BP4-DI3	Development of Synthetic order generator application							▨	▨	▨	▨	▨	▨	▨							
P1-BP4-DI4	Development of order backend persistor							▨	▨	▨	▨	▨	▨	▨							
P1-BP4-DI5	Setting up POLIMI Side AGV Navigation System												▨	▨	▨	▨	▨	▨			
P1-BP4-DI6	Development of AGV POLIMI side front end												▨	▨	▨	▨	▨	▨			
P1-BP4-DI7	Development of Communication Api between AGV Nav. And POLIMI Front end app												▨	▨	▨	▨	▨	▨			
P1-BP4-DI8	Development of AI model to optimize the number of valves							▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨			
Integration Activities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
P1-BP4-IA4	aerOS Basic components	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨		
P1-BP4-IA5	aerOS Non Basic components															▨	▨	▨	▨		
P1-BP4-IA6	Integrate Applications With Orion Context broker							▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨		
P1-BP4-IA7	APIs acces provision for existing Order/infrastructure management for MADE											▨	▨	▨	▨						
Validation Activities																	■	■	■	■	■
P1-BP4-VA1	Validation of KPI 2.1.9																▨	▨	▨	▨	▨
P1-BP4-VA2	Validation of KPI 2.1.10																▨	▨	▨	▨	▨

For sub-pilot 1.1, the activities carried out are related to the aerOS non-basic components, the integration of data analysis service for reports and statistics creation, the data quality verification, the improvement activities, the KPIs validation, the qualitative validation, and the evaluation and reporting. For sub-pilot 1.2, the activities carried out are related to the remote configuration/set-up of the CMM instrumentation robotic and kinematic configuration, the remote tactile operation of CMM, the assistance and optimisation of the process of Digital Twin creation, and the dynamic execution of metrology services and Data assembling. For sub-pilot 1.3, the activities carried out are related to the validation of KPIs 2.1.7 and 2.1.8. Finally, for sub-pilot 1.4, the activities carried out are related to the distributed order management across MADE and POLIMI domains, the AGV path planner and navigation system, the AI/ML-based outsourcing model, the Edge-first deployment of aerOS services, the integration with Orion-LD and inter-domain communication, and the validation of KPIs 2.1.5 and 2.1.9. All of these activities will be described in detail below.

Green manufacturing (zero net-energy) and CO2 footprint monitoring

Pilot 1 – Business Process 1 – Activity - 13 (P1-BP1-IA13): aerOS non-basic components

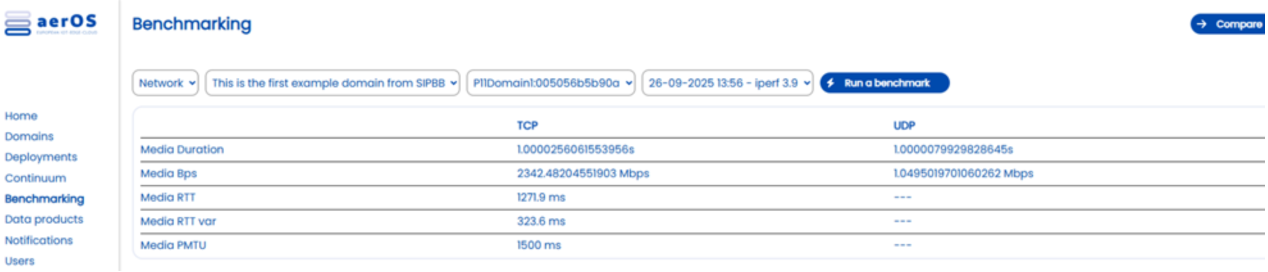
The non-basic components required for our pilot have been installed and are listed in the image below. These components are part of the embedded-analytics-tool necessary for the long-term storage of the data, function deployment and dashboard visualization.



```
vm1@vm1-VirtualBox:~/embedded-analytics-tool$ kubectl get pod -n embedded-analytics-tool
$NAME                                          READY   STATUS    RESTARTS        AGE
alertmanager-6cd75c756b-vdv2j                  1/1     Running   30 (4h55m ago)  49d
gateway-5b598d74-8xrlq                         2/2     Running   104 (4h55m ago) 57d
mygrafana-78dcbbc766-c2m84                     1/1     Running   5 (4h55m ago)   3d18h
mypush-prometheus-pushgateway-7954dd6575-52qpw 1/1     Running   6 (4h55m ago)   8d
nats-6b777dcd49-2q22n                          1/1     Running   36 (4h55m ago)  57d
postgres-575c85fc48-2ql5w                      1/1     Running   10 (4h55m ago)  10d
prometheus-67fbdf777-plqjv                     1/1     Running   6 (4h55m ago)   8d
queue-worker-76d96fdbb7-66z4w                  1/1     Running   87 (4h53m ago)  49d
```

Figure 37: Pilot 1.1. aerOS non-basic components

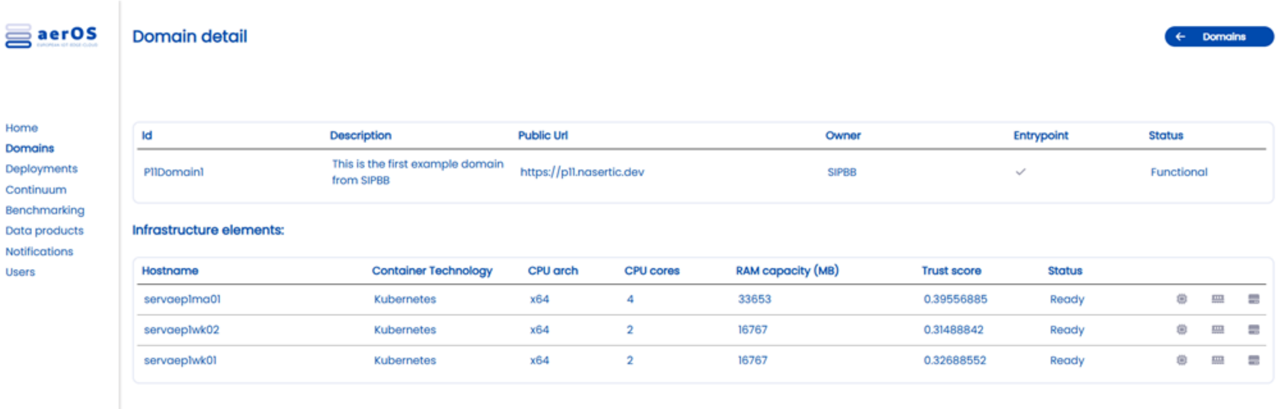
The remaining non-essential components, such as Self-Scaling, Data Catalog, Trustworthiness Score, and Benchmarking Tool, which we initially lacked, are now also installed and running.



Network | This is the first example domain from SIPBB | P1IDomain1:005056b5b90a | 26-09-2025 13:56 - iperf 3.9

	TCP	UDP
Media Duration	1.0000256061553956s	1.0000079929828645s
Media Bps	2342.48204551903 Mbps	1.0495019701060262 Mbps
Media RTT	1271.9 ms	---
Media RTT var	323.6 ms	---
Media PMTU	1500 ms	---

Figure 38: Pilot 1.1. Benchmarking Tool in the Management Portal (aerOS non-basic components)



Domain detail

Id	Description	Public Url	Owner	Entrypoint	Status
P11Domain1	This is the first example domain from SIPBB	https://p11.nasertic.dev	SIPBB	✓	Functional

Infrastructure elements:

Hostname	Container Technology	CPU arch	CPU cores	RAM capacity (MB)	Trust score	Status
servaeplma01	Kubernetes	x64	4	33653	0.39556885	Ready
servaeplwk02	Kubernetes	x64	2	16767	0.31488842	Ready
servaeplwk01	Kubernetes	x64	2	16767	0.32688552	Ready

Figure 39: Pilot 1.1. Data Catalog available in the management portal

Pilot 1 – Business Process 1 – Activity - 15 (P1-BP1-IA14): Integration of data analysis service for reports and statistics creation

The **DeepSeek-R1 1.5B LLM model** for data analysis and report generation was tested within SIPBB’s aerOS Meta OS system, deployed as an additional pod in one of aerOS domains (deployed over a Kubernetes cluster). The goal of the reporting function -that is used inside aerOS via the triggering of an Embedded Analytics Tool (EAT) function- is to support production optimization and the reduction of the overall CO_2 footprint.

The model takes production-related input data, such as machine-level energy consumption and process parameters, and generates reports that highlight patterns of energy use and their impact on emissions. These reports are designed to provide actionable insights, for example identifying which stations or drone types contribute most to CO2 output and suggesting optimization strategies such as keeping high start-up-cost machines continuously running while only activating others when needed.

However, the available hardware resources proved insufficient to achieve satisfactory performance with a locally deployed model, especially since much of the computing capacity was already occupied by other software. For practical reasons, we therefore decided to adapt our approach and integrate cloud-based LLMs via API instead.

The reporting service is now fully operational: it produces scheduled analyses and integrates with SIPBB’s existing dashboards. The reports provide tangible benefits, including anomaly detection and clearer attribution of CO_2 hotspots to specific stations and process steps. This enables faster decision-making within the production facility, reduces unplanned downtime, and supports continuous improvement toward lower energy usage and emissions.

Pilot 1 – Business Process 1 – Activity 19 (P1-BP1-VA19) - Data quality verification

Energy data completeness:

We verified that time series for each machine contain no missing timestamps and that recorded values fall within plausible ranges based on machine specifications. The verification was conducted through automated consistency checks and subsequently reviewed by a technical expert.

CO_2/PCF prediction accuracy:

We assessed the quality of CO_2 footprint predictions (product and machine level) by comparing predicted values with actuals and tracking accuracy over time. This aligns with our KPI 2.1.4 Accuracy of the CO2-footprint prediction (%), evaluated order-by-order

Pilot 1 – Business Process 1 – Activity 20 (P1-BP1-VA20) - Improvement activities

Real-time data access & monitoring :

We continuously optimized the data-pipeline code and migrated the prototype into our Kubernetes environment, improving reliability, robustness, and maintainability of the running processes.

Transparency & decision readiness:

It is now possible to retrieve CO_2/PCF (Product Carbon Footprint) data for specific products on demand via dashboards and APIs, which enhances greatly improved transparency and enables quicker, evidence-based decisions.

Pilot 1 – Business Process 1 – Activity 21 (P1-BP1-VA21) - KPIs validation

Check KPIs section for more information (Section 4).

Pilot 1 – Business Process 1 – Activity 22 (P1-BP1-VA2) - Qualitative validation

Please refer to the KeVI analysis in Section 7.

Pilot 1 – Business Process 1 – Activity 23 (P1-BP1-VA23) - Evaluation and reporting

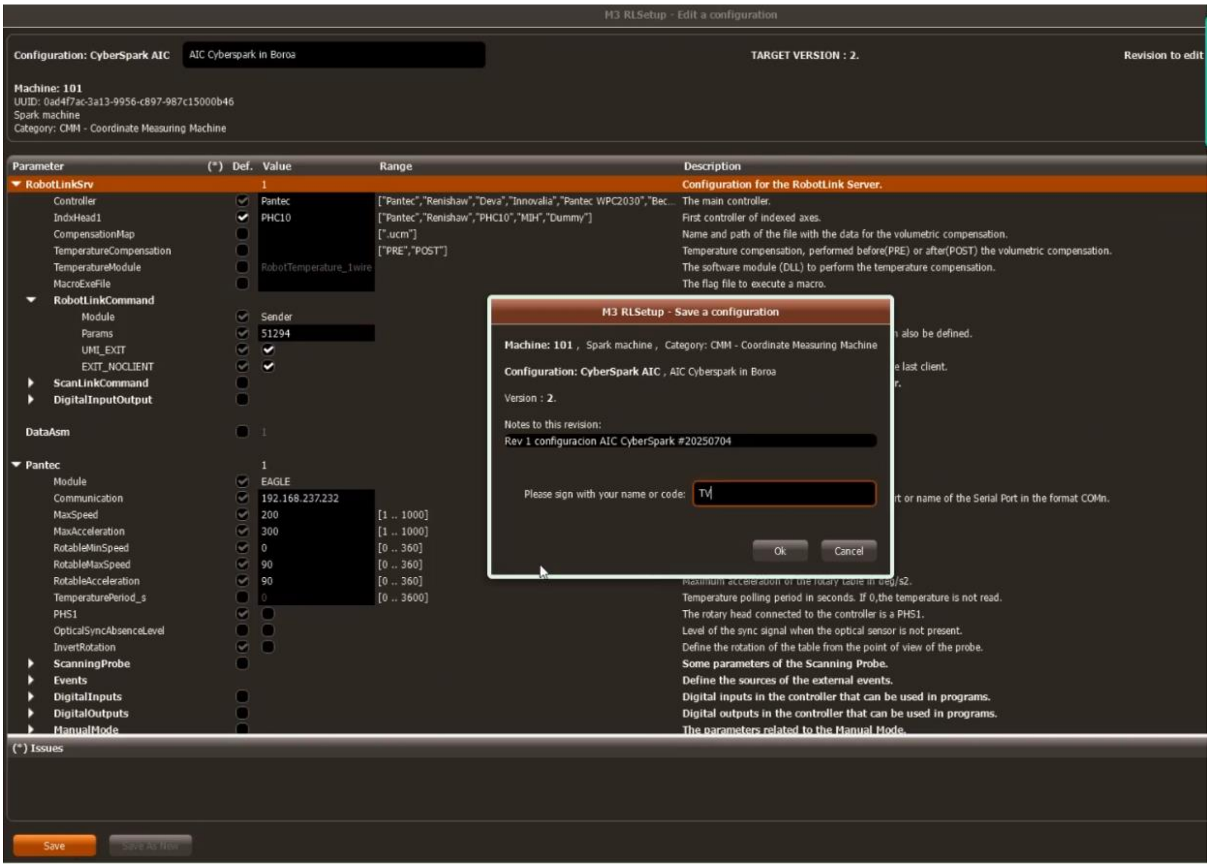
Check section 4.

Automotive Smart Factory Zero Defect Manufacturing

Pilot 1 – Business Process 2 – Activity 8 (P1-BP2-VA8) – Remote configuration/set-up of the CMM instrumentation robotic and kinematic configuration

The following images shows the configuration process and options of the RL Setup tool, which enables the remote setup of the CMM Machines via the M3 Software. The configuration allows for the customization of multiple options, from the instrumentation of the robotic parts to the kinematic—movement related variables—of the gages.

In particular the first image shows some the setup options, while the second depicts how the configuration is attached to a certain machine—in this case a CyberSpark—and saved in the system.



Configuration: CyberSpark AIC AIC Cyberspark in Boroa

TARGET VERSION : 2.

Revision to edit

Machine: 101
UUID: 0ad4f7ac-3a13-9956-c897-987c15000b46
Spark machine
Category: CMM - Coordinate Measuring Machine

Parameter	(*)	Def.	Value	Range	Description
▼ RobotLinkSrv			1		Configuration for the RobotLink Server.
Controller		☑	Pantec	["Pantec","Renishaw","Deva","Innovalia","Pantec WPC2030","Bec...	The main controller.
IndxHead1		☑	PHC10	["Pantec","Renishaw","PHC10","MIH","Dummy"]	First controller of indexed axes.
CompensationMap		☐		[".ucm"]	Name and path of the file with the data for the volumetric compensation.
TemperatureCompensation		☐		["PRE","POST"]	Temperature compensation, performed before(PRE) or after(POST) the volumetric compensation.
TemperatureModule		☐	RobotTemperature_1wire		The software module (DLL) to perform the temperature compensation.
MacroExeFile		☐			The flag file to execute a macro.
▼ RobotLinkCommand					
Module		☑	Sender		
Params		☑	51294		n also be defined.
UMI_EXIT		☑	☑		
EXIT_NOCLIENT		☑	☑		e last client.
▶ ScanLinkCommand		☐			r.
▶ DigitalInputOutput		☐			
DataAsm		☐	1		
▼ Pantec			1		
Module		☑	EAGLE		
Communication		☑	192.168.237.232		t or name of the Serial Port in the format COMn.
MaxSpeed		☑	200	[1 .. 1000]	
MaxAcceleration		☑	300	[1 .. 1000]	
RotableMinSpeed		☑	0	[0 .. 360]	
RotableMaxSpeed		☑	90	[0 .. 360]	
RotableAcceleration		☑	90	[0 .. 360]	Maximum acceleration of the rotary table in deg/s2.
TemperaturePeriod_s		☐	0	[0 .. 3600]	Temperature polling period in seconds. If 0,the temperature is not read.
PHS1		☑	☐		The rotary head connected to the controller is a PHS1.
OpticalSyncAbsenceLevel		☐	☐		Level of the sync signal when the optical sensor is not present.
InvertRotation		☑	☐		Define the rotation of the table from the point of view of the probe.
▶ ScanningProbe		☐			Some parameters of the Scanning Probe.
▶ Events					Define the sources of the external events.
▶ DigitalInputs		☐			Digital inputs in the controller that can be used in programs.
▶ DigitalOutputs		☐			Digital outputs in the controller that can be used in programs.
▶ ManualMode		☐			The parameters related to the Manual Mode.

(*) Issues

M3 RLSetup - Save a configuration

Machine: 101 , Spark machine , Category: CMM - Coordinate Measuring Machine

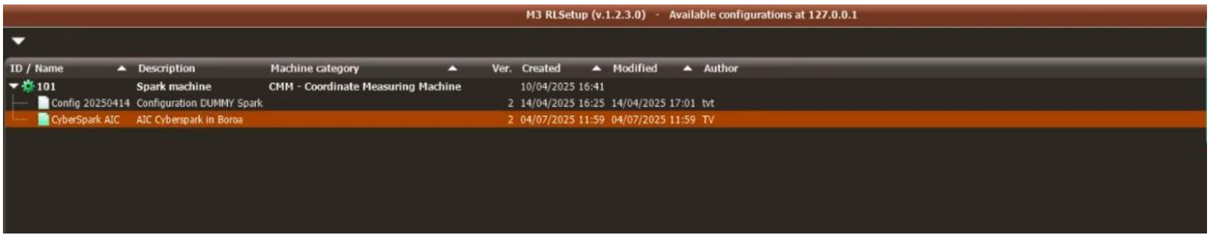
Configuration: CyberSpark AIC , AIC Cyberspark in Boroa

Version : **2.**

Notes to this revision:
Rev 1 configuracion AIC CyberSpark #20250704

Please sign with your name or code: TV

Figure 40: Pilot 1.2 Setup and Configuration of a CyberSpark


M3 RLSetup (v.1.2.3.0) - Available configurations at 127.0.0.1

ID / Name	Description	Machine category	Ver.	Created	Modified	Author
▼ 101	Spark machine	CMM - Coordinate Measuring Machine		10/04/2025 16:41		
Config 20250414	Configuration DUMMY Spark		2	14/04/2025 16:25	14/04/2025 17:01	tvt
CyberSpark AIC	AIC Cyberspark in Boroa		2	04/07/2025 11:59	04/07/2025 11:59	TV

Figure 41: Pilot 1.2 CMM CyberSpark available

Pilot 1 – Business Process 2 – Activity 9 (P1-BP2-VA9) - Remote tactile operation of CMM

aerOS MetaOS allows to remotely execute tactile operations of different CMMs. After setting up the machine with its configuration, M3 and aerOS enable the metrologist to remotely control the movement of multiple axis and tools attached of the gage. The following images displays the virtual joystick for a Spark series model and how the M3 user is able to easily control the CMM on the client premises on real-time.

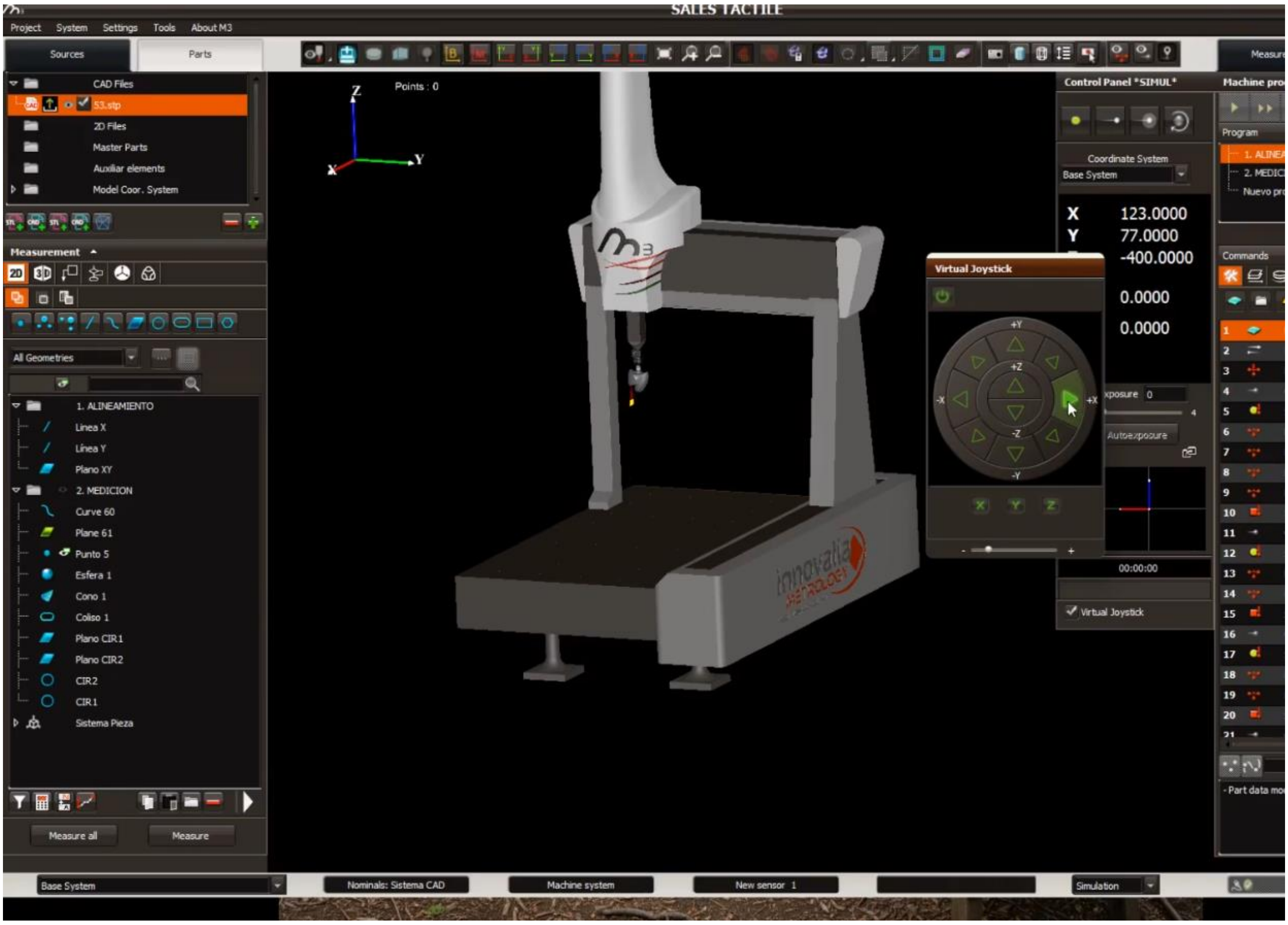


Figure 42: Pilot 1.2 Cyberspark remote control with virtual joystick

On top of that, on the M3 platform is possible to switch to another machine from the available ones within the AIC Lab, if the measurement specifications required it (e.g. higher speed, minimum accuracy, limited amount of energy). This translates into a valuable ability to operate and set-up—in a flexible manner—all these industrial services across the IoT and Edge domains.

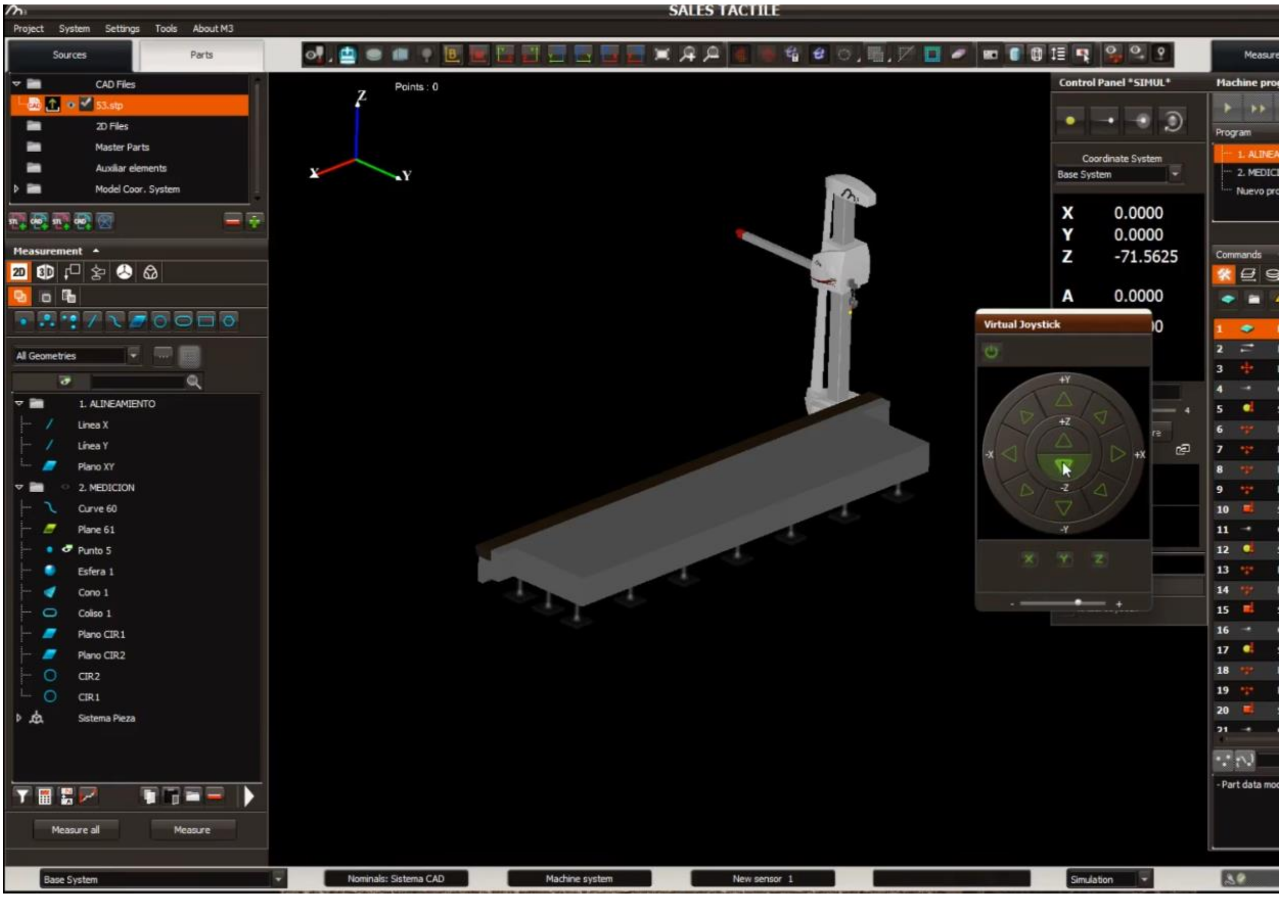


Figure 43: Secondary machine remote control with virtual joystick

Pilot 1 – Business Process 2 – Activity 10 (P1-BP2-VA10) – aerOS assist and optimize the process of Digital Twin creation.

The Human-Machine Interface (HMI) communicates directly with the M3 software, receiving real-time data streams from various sensors integrated into the metrology setup. This data includes key operational metrics such as compressed air pressure and flow, temperature, electrical current, and positional coordinates, among others.

As shown in the image below, the physical gage is digitally mirrored through a Digital Twin that runs on top of aerOS-enabled computing continuum, providing a live, virtual representation of the machine. This enables continuous, remote monitoring of its status and behaviour.

Through aerOS, all the required services for data acquisition, processing, and synchronization are deployed seamlessly across the edge infrastructure. These services ensure the consistent generation and delivery of the data needed to maintain an accurate and responsive Digital Twin of the CMMs.

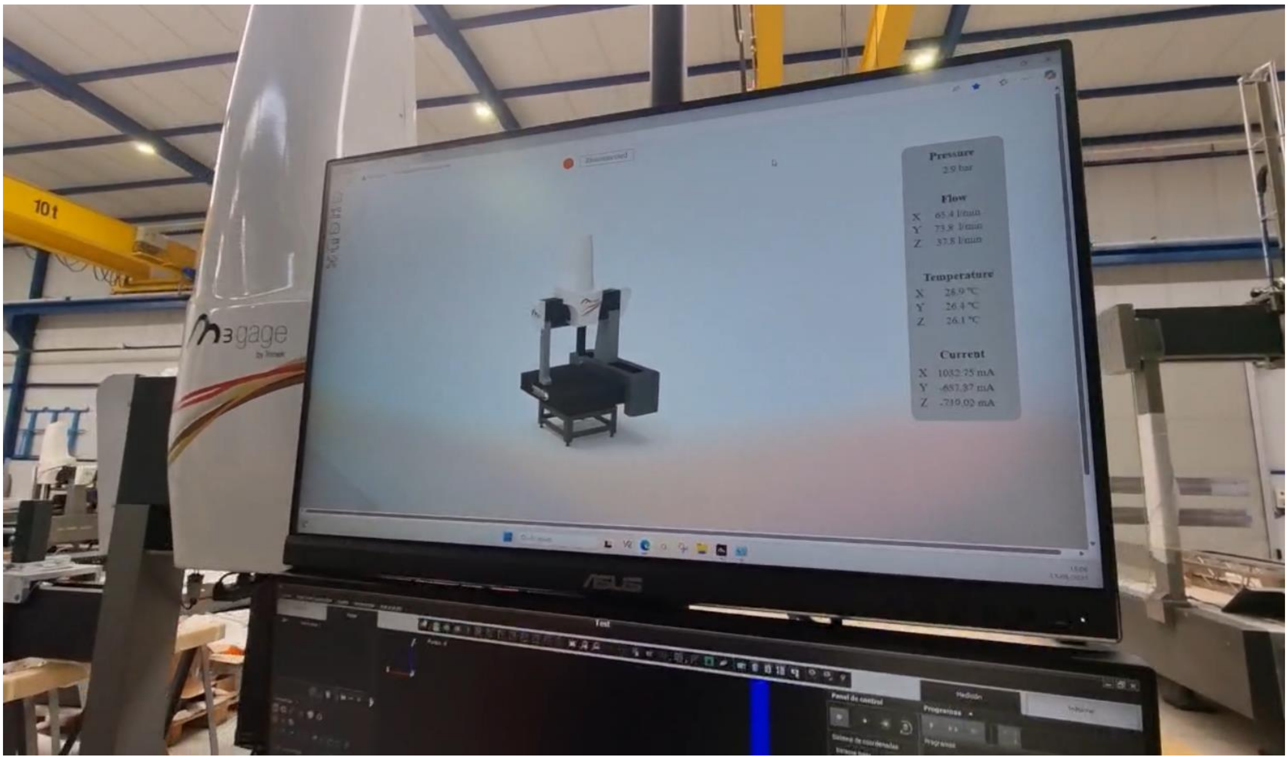


Figure 44: Pilot 1.2 HMI Digital Twin

Pilot 1 – Business Process 2 – Activity 11 (P1-BP2-VA11) – Dynamic execution of metrology services and Data assembling

The Data Assembler plays a key role in enabling real-time metrology by collecting and organizing data from multiple sources within the IoT layer. These sources include point cloud data, sensor readings, and machine axis positions, among others.

This tool integrates sensor data with machine movement parameters to produce structured, actionable information. By correlating spatial and contextual data, the Data Assembler transforms raw inputs into meaningful outputs that support precise measurement workflows.

Thanks to this component, metrology services can now be executed in a semi-automated manner, requiring only minimal intervention from the metrologist. This significantly reduces the overall duration of measurement processes by offloading repetitive and data-intensive tasks that were traditionally handled manually.

Zero Ramp-up safe PLC reconfiguration for Lot-Size-1 Production

Pilot 1 – Business Process 3 – Activity - 1 (P1-BP3-VA1): KPI 2.1.8 validation: AGV availability > 95%

Before updating the AGVs via aerOS with the new docking skill, the following situation was the case:

AGVs had to be charged manually. This meant that someone had to actively monitor the AGVs and plug a power supply into them whenever they were running low on power, as shown in the next figure. This process was not only tedious but also required manual labour. In addition, the installation and removal of the PSU (power supply unit) took up time, which could neither be used for charging nor for other productive work. A concrete example will be provided later in this document.

It is automatically detected when an AGV requires to be charged. The aerOS system then will update the skill of the AGV on the fly. After updating the AGVs via aerOS with the new docking skill, the following improvements were achieved:

The AGVs are now able to charge autonomously. They can drive independently to a free wireless charging station and dock onto it, as shown in the next figure. This marks a significant improvement because no human intervention is necessary anymore, and no time is wasted plugging devices into the AGVs. As a result, there is a considerable increase in efficiency and productivity.

The introduction of the autonomous docking skill via aerOS has led to a major improvement in the charging process and overall productivity of the AGVs. In the previous setup, AGVs required about 2 hours of charging per workday, plus roughly 30 minutes spent waiting for someone to manually plug in or unplug the power supply unit (PSU). Altogether, this meant that a maximum of 5.5 hours per 8-hour day were left for productive operation — equivalent to a 69% productivity rate.

With the new on-demand system, AGVs now require only around 20 minutes of charging per full workday and no longer depend on human involvement. This allows them to operate productively for the remaining time, resulting in a productivity level of 96%, which represents a 27% increase in daily productivity.

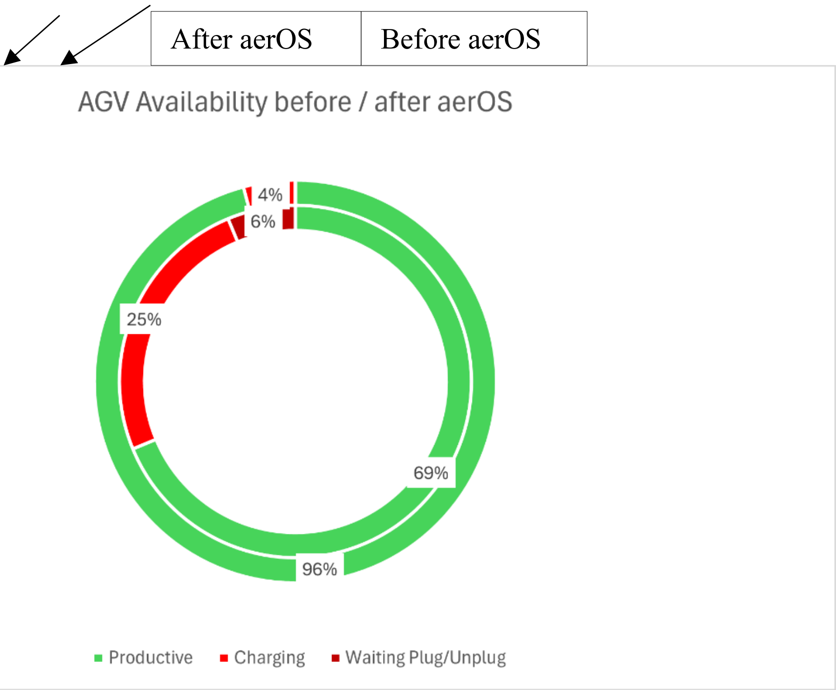


Figure 45: Pilot 1.3 AGV Availability Before & after aerOS comparison

Pilot 1 – Business Process 3 – Activity - 2 (P1-BP3-VA2): KPI 2.1.7 validation: AGV usage > 80 %

Another skill added via aerOS is the ability to communicate with the mobile robotic arm workstations. These stations can now transmit their desired location to the AGVs. As a result, it is now possible not only to move boxes with the AGVs but also to relocate the workstations themselves. The AGVs have thus gained an entirely new group of tasks without any major intervention. Before gaining this new ability, they were often idle while waiting for the next task, resulting in a usage rate of only 54%. With their newly acquired tasks, they have now increased their overall usage to over 84%, which represents a 30% increase.

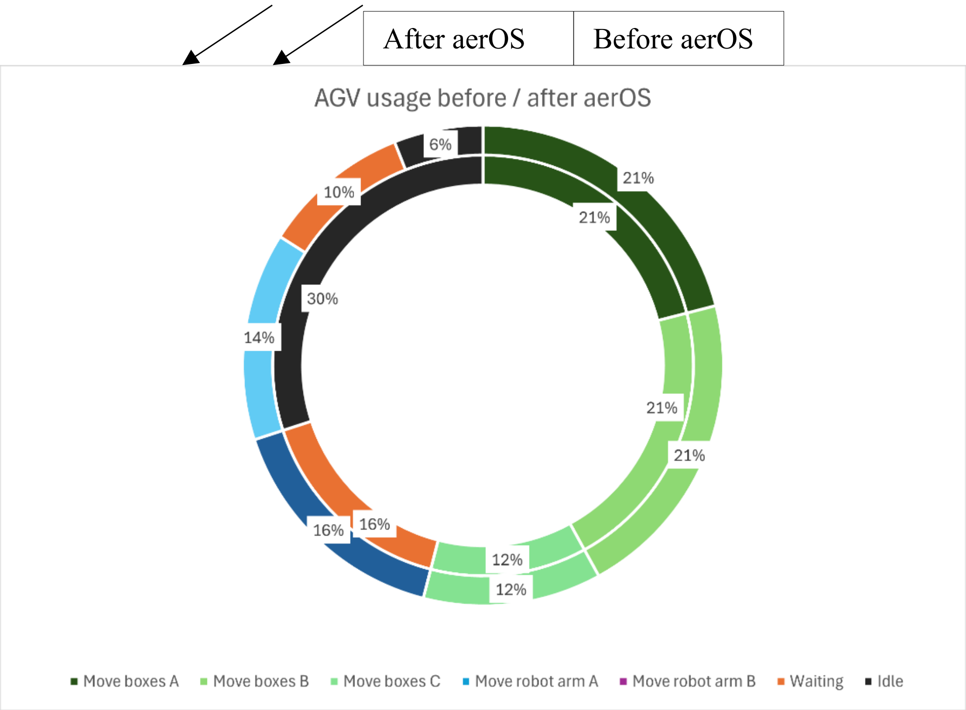


Figure 46: AGV Usage Before & After aerOS comparison

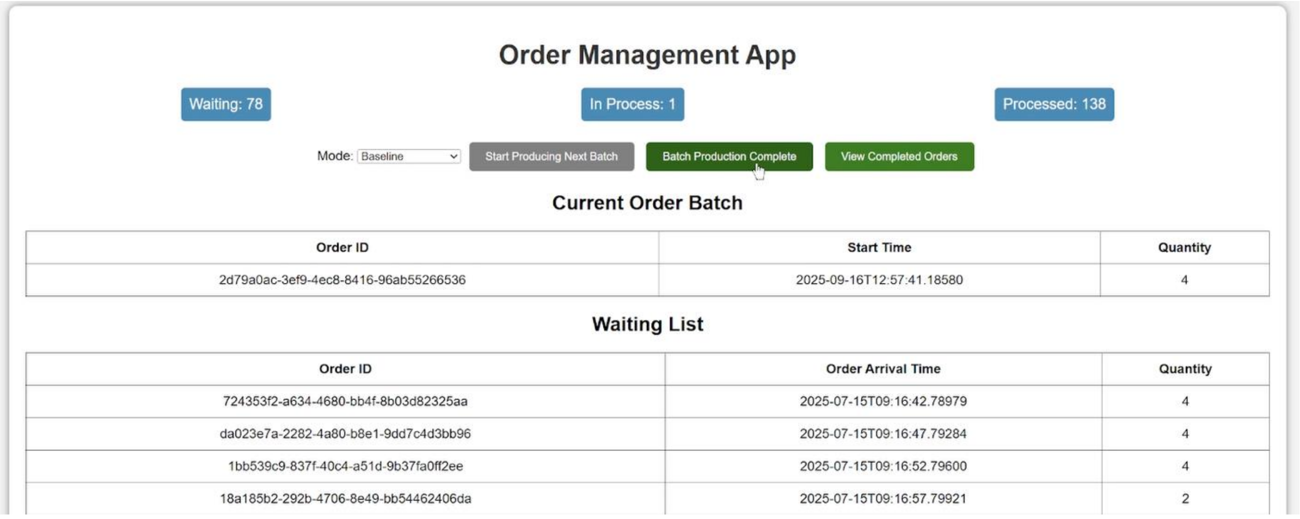
The new docking skill, which allows AGVs to autonomously approach and dock with charging stations, was packaged as a Docker container. Using the aerOS framework, this containerized skill could be efficiently transferred and deployed onto the AGVs. In this way, aerOS acted as the bridge between the software environment and the physical AGV platform, ensuring that the new skill was delivered in a standardized and reproducible manner. This approach simplified deployment and provided a reliable foundation for future skill updates.

Due to the improved charging of the AGVs, the productivity of the robotic arm stations has also increased. The Siemens Fleet Manager, in combination with aerOS, are now able to control each AGV. Whenever a robotic arm station needs to be relocated, one of the available AGVs automatically moves to the arm and transports the station to its desired destination. Before the use of aerOS, it was not possible to coordinate the AGVs in this way nor interact with the robotic arm cells. The switch to aerOS has therefore resulted in a higher efficiency for the AGVs and their usage within the flexible factory.

AGV Swarm Zero break-down logistics for Lot-Size-1 Production

Pilot 1 – Business Process 4 – Activity - 1 (P1-BP4-VA1): Distributed order management across MADE and POLIMI domains

The newly developed Order Manager applications at MADE and POLIMI were tested to validate their ability to handle aggregated orders, reduce redundant AGV travels, and ensure optimized batch handling. Compared with the legacy platform, aerOS enabled more efficient order scheduling, achieving measurable reductions in unnecessary AGV trips and idle time, the numerical results can be observed in (KPI 2.1.9)



Order Management App

Waiting: 78 In Process: 1 Processed: 138

Mode: Baseline

Current Order Batch

Order ID	Start Time	Quantity
2d79a0ac-3ef9-4ec8-8416-96ab55266536	2025-09-16T12:57:41.18580	4

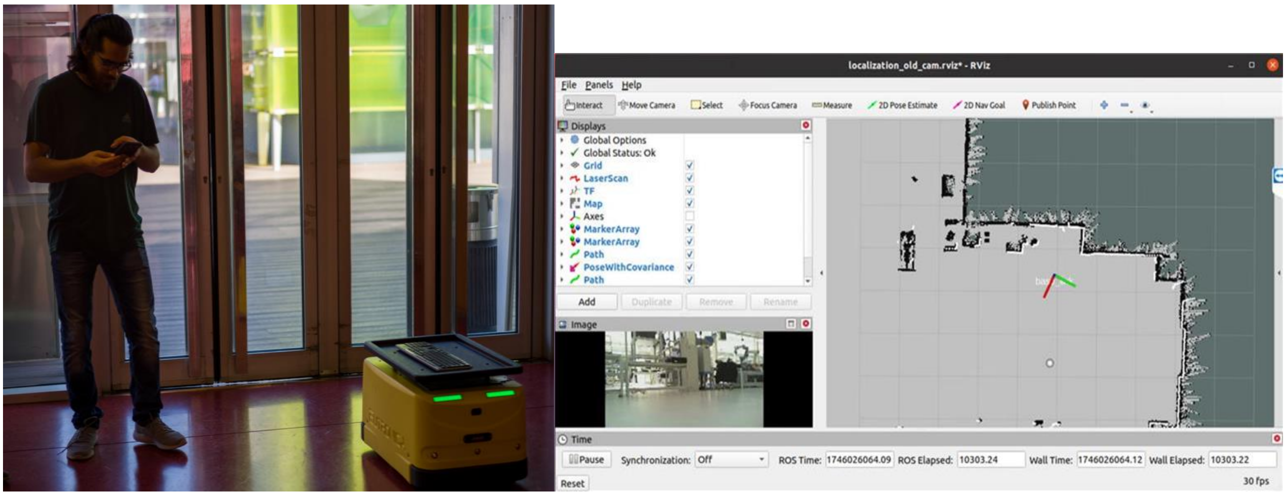
Waiting List

Order ID	Order Arrival Time	Quantity
724353f2-a634-4680-bb4f-8b03d82325aa	2025-07-15T09:16:42.78979	4
da023e7a-2282-4a80-b8e1-9dd7c4d3bb96	2025-07-15T09:16:47.79284	4
1bb539c9-837f-40c4-a51d-9b37fa0ff2ee	2025-07-15T09:16:52.79600	4
18a185b2-292b-4706-8e49-bb54462406da	2025-07-15T09:16:57.79921	2

Figure 47: Order Manager App: Batching

Pilot 1 – Business Process 4 – Activity - 2 (P1-BP4-VA2): AGV path planner and navigation system

The ROS-based navigation system was validated within the POLIMI domain. Validation confirmed that AGVs could reliably follow pre-defined linear paths with collision detection and stop mechanisms, ensuring predictability and safety in a factory environment. The tests demonstrated the improved perceived safety of AGVs compared to conventional dynamic obstacle-avoidance methods.



Synchronization: Off ROS Time: 1746026064.09 ROS Elapsed: 10303.24 Wall Time: 1746026064.12 Wall Elapsed: 10303.22

Figure 48: ROS Based Navidation system

Pilot 1 – Business Process 4 – Activity - 3 (P1-BP4-VA3): AI/ML-based outsourcing model

The AI-driven outsourcing module was validated in scenarios where MADE's line was saturated.

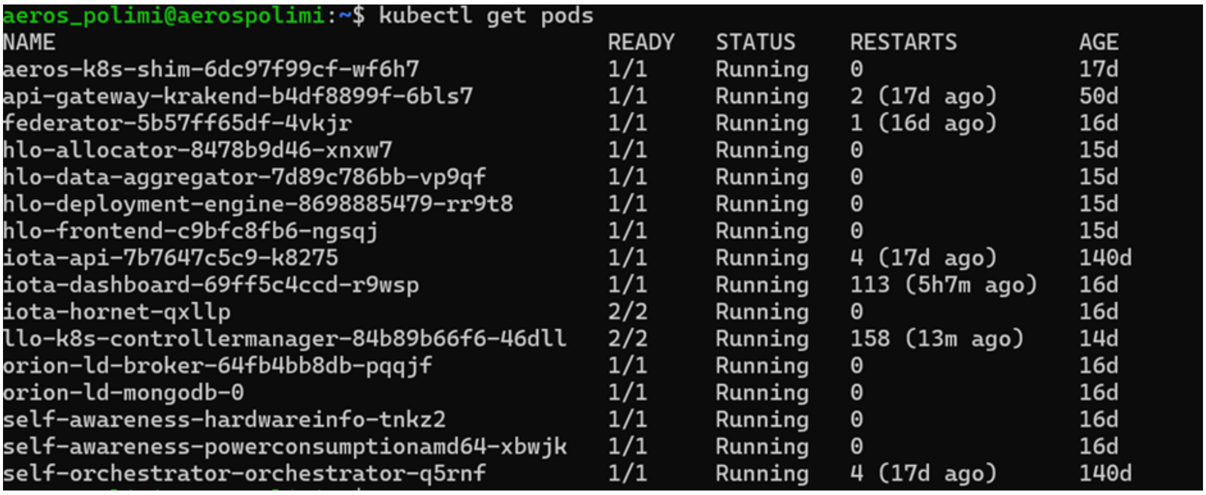
Prior to aerOS, functionality was being accomplished by a cloud-based platform called LEA, the system was quite rigid and would start the line as soon as a new order entered the system. This was quite inefficient in terms of both energy and production efficiency particularly the number of AGV travels. This was because the system does not allow the combination of multiple orders. Therefore, even though the line is capable of handling a batch of 5 valves, such combination was not feasible with the existing system.

The model, based on random forest classification, correctly predicted outsourcing decisions with high reliability, ensuring that POLIMI could take over orders dynamically. This confirmed aerOS's ability to support resilience and continuity in production chains.

Pilot 1 – Business Process 4 – Activity - 4 (P1-BP4-VA4): Edge-first deployment of aerOS services

The experiment confirmed the successful migration of nearly all pilot applications to the edge, supported by dual-architecture containerization. Out of 13 deployed applications, 12 (92%) now run on edge nodes, while only 8% remain cloud-dependent.

	Cloud	Edge
Before aerOS	All Cloud based	
After aerOS	Order Manager	@Context (MADE), Order Generator, Orion-LD (MADE), MongoDB (MADE), OM-MADE, ML Module, @Context (POLIMI), Orion-LD (POLIMI), MongoDB (POLIMI), OM-POLIMI, Button, AGV Path planner

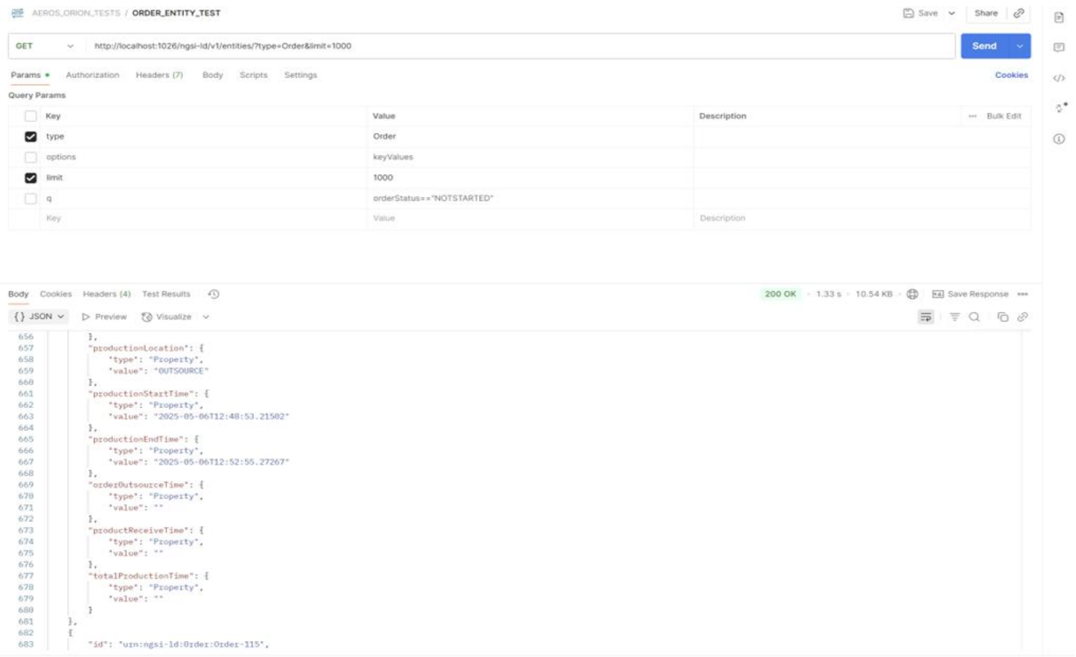


```
aeros_polimi@aerospolimi:~$ kubectl get pods
NAME                                          READY   STATUS    RESTARTS         AGE
aeros-k8s-shim-6dc97f99cf-wf6h7               1/1     Running   0                17d
api-gateway-krakend-b4df8899f-6bls7           1/1     Running   2 (17d ago)      50d
federator-5b57ff65df-4vkjr                    1/1     Running   1 (16d ago)      16d
hlo-allocator-8478b9d46-xnxw7                 1/1     Running   0                15d
hlo-data-aggregator-7d89c786bb-vp9qf          1/1     Running   0                15d
hlo-deployment-engine-8698885479-rr9t8        1/1     Running   0                15d
hlo-frontend-c9bfc8fb6-ngsqj                  1/1     Running   0                15d
iota-api-7b7647c5c9-k8275                     1/1     Running   4 (17d ago)      140d
iota-dashboard-69ff5c4ccd-r9wsp               1/1     Running   113 (5h7m ago)   16d
iota-hornet-qxllp                             2/2     Running   0                16d
llo-k8s-controllermanager-84b89b66f6-46dll    2/2     Running   158 (13m ago)    14d
orion-ld-broker-64fb4bb8db-pqqjf              1/1     Running   0                16d
orion-ld-mongodb-0                            1/1     Running   0                16d
self-awareness-hardwareinfo-tnkz2             1/1     Running   0                16d
self-awareness-powerconsumptionamd64-xbwjk    1/1     Running   0                16d
self-orchestrator-orchestrator-q5rnf          1/1     Running   4 (17d ago)      140d
```

Figure 49: POLIMI Domain

Pilot 1 – Business Process 4 – Activity - 5 (P1-BP4-VA5): Integration with Orion-LD and inter-domain communication

The Orion-LD context broker was validated as the backbone for semantic interoperability across MADE and POLIMI. Tests confirmed that order and status entities were consistently synchronized between domains. This validated interoperability and standardized data exchange across the continuum.



GET http://localhost:1026/ngsi-ld/v1/entities/?type=Order&limit=1000

Query Params

	Key	Value	Description
☑	type	Order	
☐	options	keyValues	
☑	limit	1000	
☐	q	orderStatus=="NOTSTARTED"	
	Key	Value	Description

```
            },
            "productionLocation": {
                "type": "Property",
                "value": "OUTSOURCE"
            },
            "productionStartTime": {
                "type": "Property",
                "value": "2025-05-06T12:48:53.21502"
            },
            "productionEndTime": {
                "type": "Property",
                "value": "2025-05-06T12:52:55.27267"
            },
            "orderOutsourceTime": {
                "type": "Property",
                "value": ""
            },
            "productReceiveTime": {
                "type": "Property",
                "value": ""
            },
            "totalProductionTime": {
                "type": "Property",
                "value": ""
            }
        },
        {
            "id": "urn:ngsi-ld:Order:Order-115",
```

Figure 50: Request to Orion-LD to retrieve entities of type Order

Pilot 1 – Business Process 4 – Activity - 6 (P1-BP4-VA6): KPIs 2.1.5 & 2.1.9

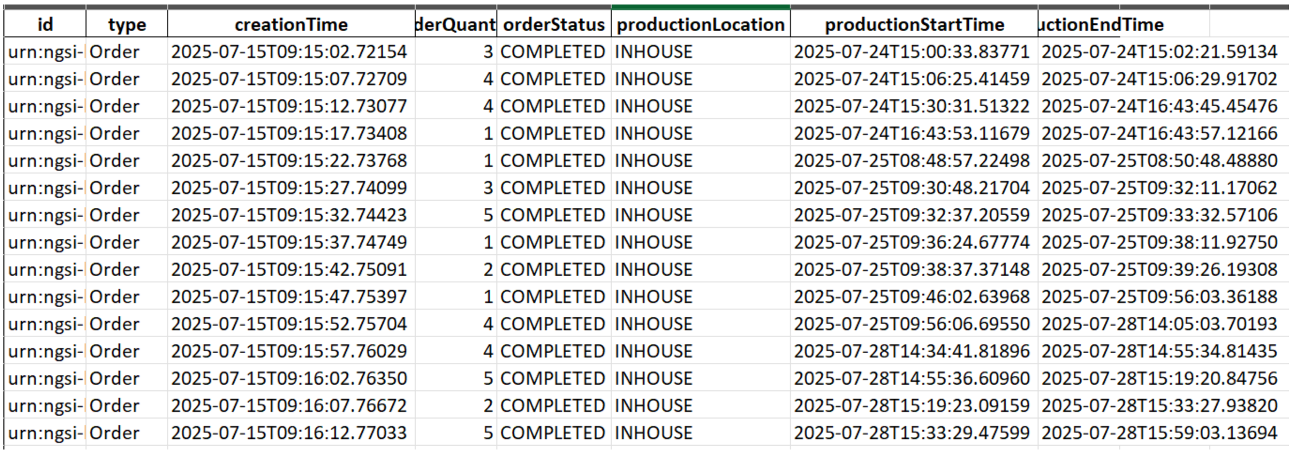
In order to calculate these KPI, POLIMI and MADE first ran the production line in baseline state to collect data for baseline generation. In order to maintain the data storage modalities, the team also persisted this data to the Orion-LD but in this case the optimization was bypassed by creating a dedicated docker image where the orders coming from the order generator are directly passed to the MADE LEA system without any changes or clubbing of orders.

In the second phase the optimization was turned on. Again, in this case also all the production data like order receive times, production start and end times etc. was captured inside Orion-LD. Finally, we used a Python script to query all the data and save this in an excel format.

```
import requests
import pandas as pd
base_url = "http://localhost:1026/ngsi-ld/v1/entities"
query_url = f"{base_url}/?type=Order&limit=1000&q=productionStartTime!=\"timestamp\""
headers = {
    'Link': '<http://context:5051/ngsi-context.jsonld>; rel="http://www.w3.org/ns/json-ld#context"
}

response = requests.get(query_url, headers=headers)
if response.status_code == 200:
    entities = response.json()
    print(f"Fetched {len(entities)} entities")
    data = []
    for e in entities:
        row = {
            "id": e.get("id"),
            "type": e.get("type"),
            "creationTime": e.get("creationTime", {}).get("value", ""),
            "orderQuantity": e.get("orderQuantity", {}).get("value", ""),
            "orderStatus": e.get("orderStatus", {}).get("value", ""),
            "productionLocation": e.get("productionLocation", {}).get("value", ""),
            "productionStartTime": e.get("productionStartTime", {}).get("value", ""),
            "productionEndTime": e.get("productionEndTime", {}).get("value", ""),
            "orderOutsourceTime": e.get("orderOutsourceTime", {}).get("value", ""),
            "productReceiveTime": e.get("productReceiveTime", {}).get("value", ""),
            "totalProductionTime": e.get("totalProductionTime", {}).get("value", "")
        }
        data.append(row)
    df = pd.DataFrame(data)
    df.to_excel("orders.xlsx", index=False)
    print("Saved entities to orders.xlsx")
else:
    print(f"Failed to fetch entities: {response.status_code} {response.text}")
```

Figure 51: Pilot 1.4 Python script query all the data and save in an excel format.

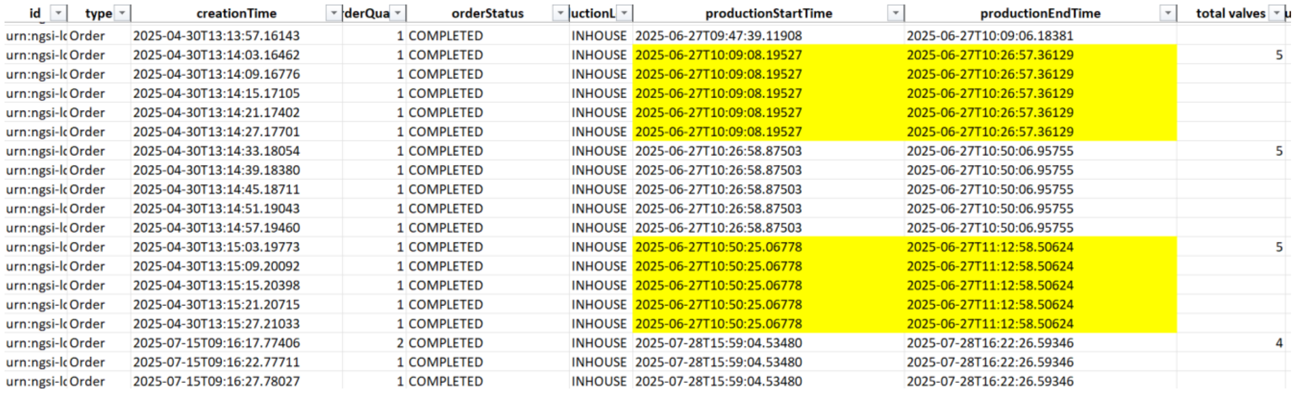
For baseline case, it is possible to directly use the order quantity to calculate the average number of valves per travel from the order quantity.



id	type	creationTime	derQuant	orderStatus	productionLocation	productionStartTime	uctionEndTime
urn:ngsi-	Order	2025-07-15T09:15:02.72154	3	COMPLETED	INHOUSE	2025-07-24T15:00:33.83771	2025-07-24T15:02:21.59134
urn:ngsi-	Order	2025-07-15T09:15:07.72709	4	COMPLETED	INHOUSE	2025-07-24T15:06:25.41459	2025-07-24T15:06:29.91702
urn:ngsi-	Order	2025-07-15T09:15:12.73077	4	COMPLETED	INHOUSE	2025-07-24T15:30:31.51322	2025-07-24T16:43:45.45476
urn:ngsi-	Order	2025-07-15T09:15:17.73408	1	COMPLETED	INHOUSE	2025-07-24T16:43:53.11679	2025-07-24T16:43:57.12166
urn:ngsi-	Order	2025-07-15T09:15:22.73768	1	COMPLETED	INHOUSE	2025-07-25T08:48:57.22498	2025-07-25T08:50:48.48880
urn:ngsi-	Order	2025-07-15T09:15:27.74099	3	COMPLETED	INHOUSE	2025-07-25T09:30:48.21704	2025-07-25T09:32:11.17062
urn:ngsi-	Order	2025-07-15T09:15:32.74423	5	COMPLETED	INHOUSE	2025-07-25T09:32:37.20559	2025-07-25T09:33:32.57106
urn:ngsi-	Order	2025-07-15T09:15:37.74749	1	COMPLETED	INHOUSE	2025-07-25T09:36:24.67774	2025-07-25T09:38:11.92750
urn:ngsi-	Order	2025-07-15T09:15:42.75091	2	COMPLETED	INHOUSE	2025-07-25T09:38:37.37148	2025-07-25T09:39:26.19308
urn:ngsi-	Order	2025-07-15T09:15:47.75397	1	COMPLETED	INHOUSE	2025-07-25T09:46:02.63968	2025-07-25T09:56:03.36188
urn:ngsi-	Order	2025-07-15T09:15:52.75704	4	COMPLETED	INHOUSE	2025-07-25T09:56:06.69550	2025-07-28T14:05:03.70193
urn:ngsi-	Order	2025-07-15T09:15:57.76029	4	COMPLETED	INHOUSE	2025-07-28T14:34:41.81896	2025-07-28T14:55:34.81435
urn:ngsi-	Order	2025-07-15T09:16:02.76350	5	COMPLETED	INHOUSE	2025-07-28T14:55:36.60960	2025-07-28T15:19:20.84756
urn:ngsi-	Order	2025-07-15T09:16:07.76672	2	COMPLETED	INHOUSE	2025-07-28T15:19:23.09159	2025-07-28T15:33:27.93820
urn:ngsi-	Order	2025-07-15T09:16:12.77033	5	COMPLETED	INHOUSE	2025-07-28T15:33:29.47599	2025-07-28T15:59:03.13694

Figure 52: Pilot 1.4 Number of valves per travel from the order quantity

For the Improved case, it was needed to first club the orders in which production start time and end time are identical (meaning these were clubbed together) to create a new column showing the total number of valves carried in a trip. Figure below shows a snapshot of this process



id	type	creationTime	derQua	orderStatus	uctionL	productionStartTime	productionEndTime	total valves	u
urn:ngsi-lc	Order	2025-04-30T13:13:57.16143	1	COMPLETED	INHOUSE	2025-06-27T09:47:39.11908	2025-06-27T10:09:06.18381		
urn:ngsi-lc	Order	2025-04-30T13:14:03.16462	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129	5	
urn:ngsi-lc	Order	2025-04-30T13:14:09.16776	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129		
urn:ngsi-lc	Order	2025-04-30T13:14:15.17105	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129		
urn:ngsi-lc	Order	2025-04-30T13:14:21.17402	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129		
urn:ngsi-lc	Order	2025-04-30T13:14:27.17701	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129		
urn:ngsi-lc	Order	2025-04-30T13:14:33.18054	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755	5	
urn:ngsi-lc	Order	2025-04-30T13:14:39.18380	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755		
urn:ngsi-lc	Order	2025-04-30T13:14:45.18711	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755		
urn:ngsi-lc	Order	2025-04-30T13:14:51.19043	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755		
urn:ngsi-lc	Order	2025-04-30T13:14:57.19460	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755		
urn:ngsi-lc	Order	2025-04-30T13:15:03.19773	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624	5	
urn:ngsi-lc	Order	2025-04-30T13:15:09.20092	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624		
urn:ngsi-lc	Order	2025-04-30T13:15:15.20398	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624		
urn:ngsi-lc	Order	2025-04-30T13:15:21.20715	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624		
urn:ngsi-lc	Order	2025-04-30T13:15:27.21033	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624		
urn:ngsi-lc	Order	2025-07-15T09:16:17.77406	2	COMPLETED	INHOUSE	2025-07-28T15:59:04.53480	2025-07-28T16:22:26.59346	4	
urn:ngsi-lc	Order	2025-07-15T09:16:22.77711	1	COMPLETED	INHOUSE	2025-07-28T15:59:04.53480	2025-07-28T16:22:26.59346		
urn:ngsi-lc	Order	2025-07-15T09:16:27.78027	1	COMPLETED	INHOUSE	2025-07-28T15:59:04.53480	2025-07-28T16:22:26.59346		

Finally, from both these values, the average values are taken for both the conditions and then calculate the reduction in terms of travels per valve. Which is reported in the following table. This same reduction directly impacts the CO2 production.

Table 16: Pilot 1.4 Reduction directly impact the CO2 production.

Parameter	Final	Initial	delta	Percentage reduction
Valves / travel	4.95238	3	1.95238	
Travels / valves	0.201923	0.333333	-0.13141	-39.42%
CO2 emissions reduction (Kg/year)	61,56	101,632	- 40,06	-39.42%

All the underlying data is available in both the Orion-LD as well as the MADE LEA System and it is possible to verify the same on request.

Pilot 2 – Data-Driven cognitive production lines

Pilot 2 has validated aerOS scheduling and federation with node pools and semi-automatic placement across edge and cloud for Earth Observation and energy workloads, tracking gains in energy use, CPU efficiency, and job completion while proving a hardened stack (monitoring, autoscaling, certificates, storage, CNI) on bare-metal edge nodes.

Pilot 2		2024											2025								
Code	**Name**	**M19**	**M20**	**M21**	**M22**	**M23**	**M24**	**M25**	**M26**	**M27**	**M28**	**M29**	**M30**	**M31**	**M32**	**M33**	**M34**	**M35**	**M36**	**M37**	**M38**
Business Process 1 (Scenario 1) - Green Edge Processing		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Setup & Procurement Activities		■	■	■	■	■	■	■	■												
P2-BP1-SA1	A1- Obtaining servers and switches	■																			
P2-BP1-SA2	A2 - Obtaining RACKs, servers and switches for second container		■	■	■																
P2-BP1-SA3	A3 - Preparation for un-trusted workloads								■												
Development Activities			■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■			
P2-BP1-DA4	A4 - HW installation and run test in container		■																		
P2-BP1-DA5	A5 - K8s setup and test			■																	
P2-BP1-DA6	A6 - HW installation and run test in the LAB			■																	
P2-BP1-DA7	A7 - Containerized Edge Node Integration with Electrum components			■																	
P2-BP1-DA8	A8 - Containerized Edge Node Integration with SRIPAS components			■																	
P2-BP1-DA9	A9 - Lab Edge Node Integration with aerOS components				■																
P2-BP1-DA10	A10 - Autoscaler monitor Development										■	■									
P2-BP1-DA11	A11 - Development of future Energy Price microservices												■	■	■	■					

P2-BP1-DA12	A12 - Development of DF the data sources connectors														▨	▨	▨	▨			
Integration Activities				■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■		
P2-BP2-IA10	A10 - First Containerized Edge Node Integration with Electrum components			▨																	
P2-BP2-IA11	A11 - First Containerized Edge Node Integration with SRIPAS components				▨																
P2-BP2-IA12	A12 - Container deployment					▨															
P2-BP2-IA13	A13 - Container connection to PV					▨															
P2-BP2-IA15	A15- aerOS Basic components								▨	▨	▨	▨									
P2-BP2-IA16	A16 - aerOS Non Basic components											▨	▨								
P2-BP2-IA17	A17 - Energy Price microservices integration														▨	▨	▨	▨			
P2-BP2-IA18	A18 - Container deployment in new location												▨	▨							
P2-BP2-IA19	A19 - Container connection to PV (second green energy source)														▨	▨	▨	▨			
P2-BP2-IA20	A20 - aerOS Basic components(up-to-date version)															▨	▨	▨	▨		
P2-BP2-IA21	A21 - Integration in the DF the next data sources															▨	▨	▨	▨		
Validation Activities				■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
P2-BP2-VA17	A17 - First Containerized Edge Node test				▨																
P2-BP2-VA18	A18 - K8s setup and test			▨																	
P2-BP2-VA19	A19 - Second Containerized Edge Node test				▨																
P2-BP2-VA20	A20 - Both Containerized Edge Node run test with aerOS				▨																
P2-BP2-VA21	A21 - HW installation and run test in container					▨															

P2-BP2-VA22	A22 - Scenario 1 deployment and test						▨										▨	▨	▨		
P2-BP2-VA23	A23 - Scenario 1 lessons learned						▨												▨	▨	
P2-BP2-VA27	A27 - KPI validation (1st version)																▨	▨	▨		
P2-BP2-VA28	A28 - KPI validation (final)																		▨	▨	
Business Process 2 (Scenario 2) - Secure Federation of edge/cloud																					
Setup & Procurement Activities																					
Development Activities																					
Integration Activities								■	■	■											
P2-BP2-IA14	A14 - Inter-cloud integration							▨													
Validation Activities								■	■	■											
P2-BP2-VA24	A24 - Configuration Validation test								▨												
P2-BP2-VA25	A25 - Scenario 2 deployment and test								▨												
P2-BP2-VA26	A26 - Scenario 2 lessons learned									▨											

Pilot 2 – Business Process 1 – Activity - 17 (P2-BP1-VA17): First Containerized Edge Node test

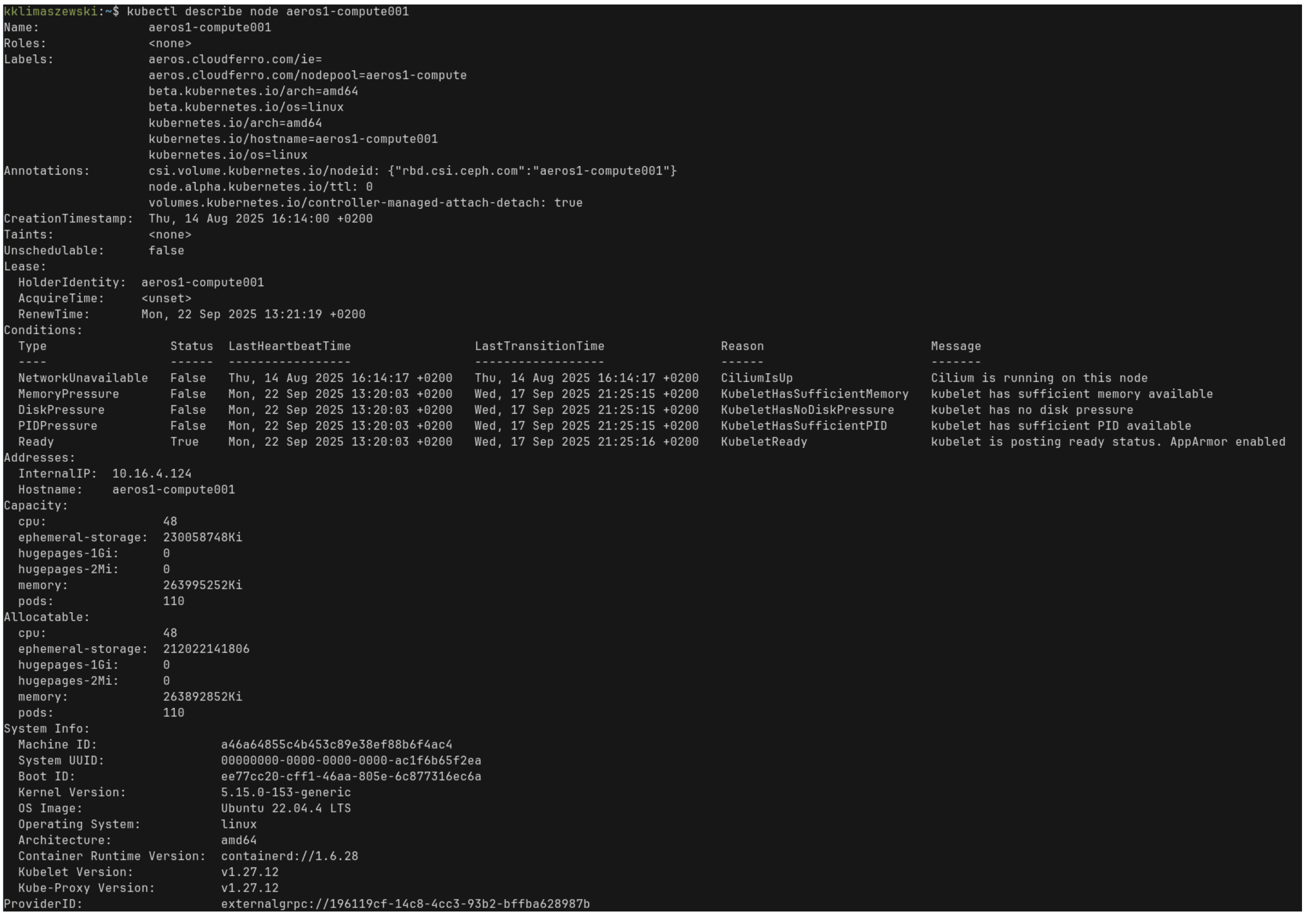
Power and temperature parameters were checked and successfully verified.

Figure 53: RackChiller Rear Door Display

Pilot 2 – Business Process 1 – Activity - 18 (P2-BP1-VA18): K8s setup and test

K8s installation was verified as a stable installation for both clusters. All nodes set up as workers joined the appropriate cluster thanks to *kubeadm* and aerOS flexibility with correct name and parameters. There were no connectivity issues beteween the nodes in each cluster. Load balancer services (including ingress) deployed in edge clusters are accessible in cloud/entrypoint cluster.

Below picture shows an example of aerOS Meta OS validated (properly functioning and deploying expected workloads) in the cluster. Fields to note are name – based on hostname configured for server, *aeros.cloud-ferro.com/nodepool=aeros1-compute* label – identifier of node kind for purposes of autoscaling, and ProviderID field – used to identify node in ironic service (which is responsible for provisioning servers and managing their powerstate in response to autoscaler decisions).



```
kklimaszewski:~$ kubectl describe node aeros1-compute001
Name:               aeros1-compute001
Roles:              <none>
Labels:             aeros.cloudferro.com/ie=
                    aeros.cloudferro.com/nodepool=aeros1-compute
                    beta.kubernetes.io/arch=amd64
                    beta.kubernetes.io/os=linux
                    kubernetes.io/arch=amd64
                    kubernetes.io/hostname=aeros1-compute001
                    kubernetes.io/os=linux
Annotations:        csi.volume.kubernetes.io/nodeid: {"rbd.csi.ceph.com":"aeros1-compute001"}
                    node.alpha.kubernetes.io/ttl: 0
                    volumes.kubernetes.io/controller-managed-attach-detach: true
CreationTimestamp:  Thu, 14 Aug 2025 16:14:00 +0200
Taints:             <none>
Unschedulable:      false
Lease:
  HolderIdentity:  aeros1-compute001
  AcquireTime:     <unset>
  RenewTime:       Mon, 22 Sep 2025 13:21:19 +0200
Conditions:
  Type                 Status  LastHeartbeatTime                 LastTransitionTime                Reason                       Message
  ----                 ------  -----------------                 ------------------                ------                       -------
  NetworkUnavailable   False   Thu, 14 Aug 2025 16:14:17 +0200   Thu, 14 Aug 2025 16:14:17 +0200   CiliumIsUp                   Cilium is running on this node
  MemoryPressure       False   Mon, 22 Sep 2025 13:20:03 +0200   Wed, 17 Sep 2025 21:25:15 +0200   KubeletHasSufficientMemory   kubelet has sufficient memory available
  DiskPressure         False   Mon, 22 Sep 2025 13:20:03 +0200   Wed, 17 Sep 2025 21:25:15 +0200   KubeletHasNoDiskPressure     kubelet has no disk pressure
  PIDPressure          False   Mon, 22 Sep 2025 13:20:03 +0200   Wed, 17 Sep 2025 21:25:15 +0200   KubeletHasSufficientPID      kubelet has sufficient PID available
  Ready                True    Mon, 22 Sep 2025 13:20:03 +0200   Wed, 17 Sep 2025 21:25:16 +0200   KubeletReady                 kubelet is posting ready status. AppArmor enabled
Addresses:
  InternalIP:  10.16.4.124
  Hostname:    aeros1-compute001
Capacity:
  cpu:                48
  ephemeral-storage:  230058748Ki
  hugepages-1Gi:      0
  hugepages-2Mi:      0
  memory:             263995252Ki
  pods:               110
Allocatable:
  cpu:                48
  ephemeral-storage:  212022141806
  hugepages-1Gi:      0
  hugepages-2Mi:      0
  memory:             263892852Ki
  pods:               110
System Info:
  Machine ID:                 a46a64855c4b453c89e38ef88b6f4ac4
  System UUID:                00000000-0000-0000-0000-ac1f6b65f2ea
  Boot ID:                    ee77cc20-cff1-46aa-805e-6c877316ec6a
  Kernel Version:             5.15.0-153-generic
  OS Image:                   Ubuntu 22.04.4 LTS
  Operating System:           linux
  Architecture:               amd64
  Container Runtime Version:  containerd://1.6.28
  Kubelet Version:            v1.27.12
  Kube-Proxy Version:         v1.27.12
ProviderID:                   externalgrpc://196119cf-14c8-4cc3-93b2-bffba628987b
```

Figure 54: Example node on Pilot 2's edge cluster

Pilot 2 – Business Process 1 – Activity - 19 (P2-BP1-VA19): Second Containerized Edge Node test

Power and temperature parameters were checked and successfully verified in the testing environment. Testing was conducted in the LAB environment because there was an accident with a car crashing the container.

Pilot 2 – Business Process 1 – Activity - 20 (P2-BP1-VA20): Both Containerized Edge Node run test with aerOS

aerOS component listed in installation guide are installed in entrypoint (cloud) and other (edge) domains. Service Component is successfully scheduled through management portal which resulted in corresponding pod starting on one of the servers in one of the nodes. This works for both edge nodes.

After changes to LLO and introduction of autoscaler-monitor it is possible to create Service Components with nodepools as target IEs – in such case scheduling within a nodepool/domain is done by kube-scheduler.

Edge domains have been federated with entrypoint domain and are visible in management portal. They can also be queried from krakend API gateway.

Pilot 2 – Business Process 1 – Activity - 21 (P2-BP1-VA21): HW installation and run test in container

The containers were installed on CF site. Second batch of hardware was mounted and connected inside.

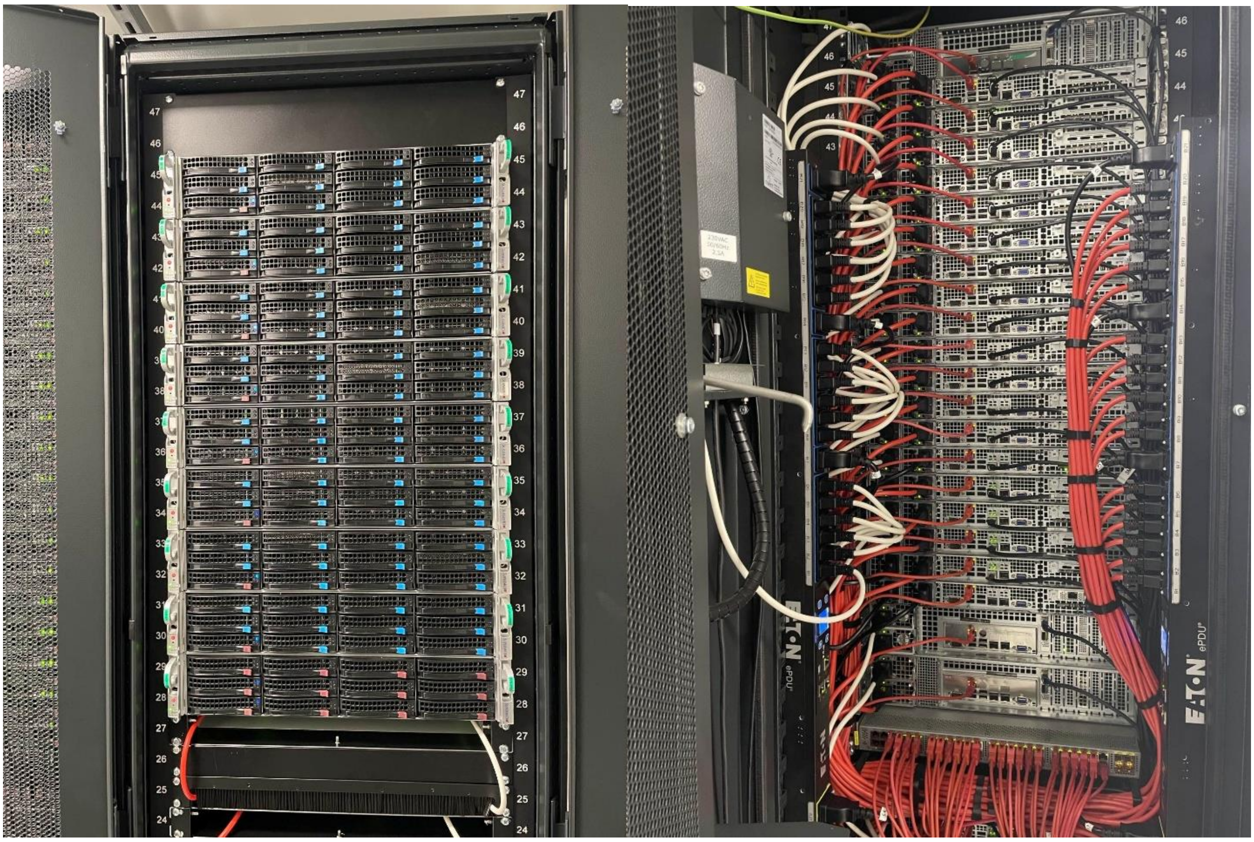


Figure 55: Hardware within the container

Pilot 2 – Business Process 1 – Activity - 22 (P2-BP1-VA22): Scenario 1 deployment and test

Hardware (metal containers with PV power source) and software (aerOS components) environments are prepared.

Pilot 2 – Business Process 1 – Activity - 23 (P1-BP1-VA23): Scenario 1 lessons learned

Gained hands-on experience with Kubernetes as a container orchestration tool, including cluster setup and automated persistent storage allocation. Learned how to configure services using **Nginx Ingress Controller** for internal communication. Developed and deployed **Helm manifests** to define and manage application resources declaratively. Improved understanding of service configuration, device monitoring integration, and overall Kubernetes-based infrastructure management.

Understanding of all components and tasks required for setting up a containerized private site, including cooling, power supply, fire protection, and racks suitable for containers.

Pilot 2 – Business Process 1 – Activity - 27 (P2-BP1-VA27): KPI validation (1st version)

The KPI definitions had been completed and the specifications for measuring them had also been finished. However, the validation (measuring the values) was still pending because the installation of all aerOS components was required. Results of this process of KPI validation can be observed in Section 4 and Appendix C of this deliverable.

A version of management portal allowing scheduling in semi-automatic mode was installed. This feature was necessary to test most of Pilot 2's KPIs. Workloads need to be set to be scheduled to one of a selected group of IEs. In this case, these selected IEs would be nodepool IEs. Such scheduling would make aerOS choose an appropriate domain for the workload and kube-scheduler would handle scheduling inside each domain.

With semi-automatic functionality of aerOS validated, Pilot 2 team was able to proceed with KPI validation. We prepared Earth Observation data workloads and deployed them through aerOS. We specified aeros1 and aeros2 nodepool IEs as vali IEs for them (we could specify both thanks to semi-automatic mode). This meant that HLO made the decision to which domain the workload should be assigned and kube-scheduler decided on which node in the cluster the workload was put. Our workloads were long-running pods that fetched jobs to run from a RabbitMQ queue in entrypoint domain, executed them and returned the results. We monitored:

- Energy consumed (KPI-2.2.1) - based on data from energy meters gathered by Electrum
- The number of tasks finished (KPI-2.2.3) - by amount of aerOS service components that are Finished
- CPU Utilization efficiency (KPI-2.2.4) – by a script gathering CPU usage from node exporter metric on prometheus at an interval
- CPU Utilization efficiency (KPI-2.2.4) – by a script gathering CPU usage from node exporter metric on prometheus at an interval
- The number of jobs finished (KPI-2.2.7) as amount of queue items processed.

Remaining KPIs was determined based on static data.

Pilot 2 – Business Process 2 – Activity - 24 (P2-BP2-VA24): Configuration Validation test

Validation tests were conducted: network connection, overall health checks.

The following necessary components running on edge clusters (besides aerOS) were validated to be running correctly:

- Prometheus in edge and cloud clusters and grafana in cloud cluster – metrics were validated to be collected from clusters and displayed in grafana
- Ironic, custom autoscaler provider, cluster-autoscaler in edge clusters – nodes were apropriately provisioned, configured according to given parameters and could be scaled down and up by cluster autoscaler
- Metallb for loadbalancers on bare-metal clusters in edge clusters – load balancers created on edge nodes were accessible outside the cluster
- Vault instance in cloud cluster and its usage in cloud and edge clusters for secrets and certificates – secrets created through vault UI were accessible in edge clusters’ secrets and pod environment variables, cert manager could create TLS certificates with vault root CA.
- Cert-manager + trust-manager for TLS certificate management for pods in edge and cloud clusters – cert manager could create certificates with vault integration or self-signed, trust manager properly created trust bundles for use by pods
- CEPH as storage solution with ceph-csi for attaching storage to pods – PVC created on edge clusters are handled by ceph-csi and corresponding CEPH volumes are created, that can be then attached to pods
- DNSmasq as DNS and DHCP manager for bare metal nodes and provisioning networks – bare-metal nodes receive DHCP configuration both in standard and provisioning networks, their DNS queries are handled correctly by dnsmasq, with custom handling of directly configured records like those of edge ingresses
- Custom CoreDNS configuration for pods DNS – same as in DNSmasq, custom handling of directly configured records works correctly
- Cilium as CNI solution for edge nodes – there is connectivity to/between pods and to services, there are no issues reported by Kubelets regarding CNI
- Kepler for collection of energy related metrics – metrics are reported in grafana
- NGINX Ingress controller for enabling access to http services outside the clusters – created ingressses are accessible outside the clusters

Pilot 2 – Business Process 2 – Activity - 25 (P2-BP2-VA25): Scenario 2 deployment and test

There were several challenges related to handling mixed workloads in aerOS.

First, due to high amount of workloads wished to be created it was not really feasible to schedule them through management portal and using API directly was not simple. It must be reminded that aerOS has been designed to reach a TRL of 5 / 6, therefore massive scaling tests have been not performed yet. This should not be considered a drawback but an opportunity for improvement.

To handle this, a simple python client was created, being it a wrapper around the API that can list, get, delete and create aerOS Service Components. Creation is handled by constructing appropriate TOSCA document and sending it to HLO FE Krakend endpoint. As an example, execution with following arguments:

```
python3 manage_services/manage_services.py create
    --service-name test-2025-09-22-001 --service-component-name sc
    --image aeros/workload-images/test_job:master
    --repository registry.cloudferro.com --job --requested-cpu-cores 32
    --requested-memory-mb 8192 --cli-arg "3000=" --use-private-registry
    --allowed-on-ids "urn:ngsi-ld:InfrastructureElement:aeros1:aeros1-compute"
```

Creates service with a following TOSCA:

```
description: test-2025-09-22-001
node_templates:
  sc:
    artifacts:
      application_image:
        file: aeros/workload-images/test_job:master
        is_private: true
        password: ************
        repository: registry.cloudferro.com
        type: tosca.artifacts.Deployment.Image.Container.Docker
        username: robot
    interfaces:
      Standard:
        create:
          implementation: application_image
          inputs:
            cliArgs:
            - '3000': ''
            envVars:
            - llo.resources.requests.cpu: '32.0'
            - llo.resources.requests.memory: 8192Mi
    isJob: true
    requirements:
    - host:
        node_filter:
          properties:
            id:
            - urn:ngsi-ld:InfrastructureElement:aeros1:aeros1-compute
    type: tosca.nodes.Container.Application
serviceOverlay: false
tosca_definitions_version: tosca_simple_yaml_1_3
```

Figure 56: TOSCA created by manage_services.py script

Such client allowed us to create large amount of Service Components and control if they are running easily.

Another challenge was that due to wanting to maximize CPU efficiency we needed to only use nodes if they were needed. Solution to this consisted of three parts:

1. We used cluster autoscaler to shut down nodes if other nodes could handle all the workloads.
2. For scheduling aerOS workloads to nodepools instead of directly to nodes we modified LLO and created autoscaler-monitor component

3. Since aerOS did not pass resource requirements specified by user in TOSCA to Kubernetes (as resource.requests and resource.limits in deployment) we created custom handling of some environment variables specified by the user in LLO.

The third point can be seen in the above request. TOSCA includes environment variables set on SC service Component. Our custom LLO version recognizes them and sets appropriate values on created Deployment/Job. In the same way user can specify that they want a volume of specific size attached to the pod.

Pilot 2 – Business Process 2 – Activity - 26 (P2-BP2-VA26): Scenario 2 lessons learned

Creating the future energy price prediction microservice, pilot 2 team learned about the volatility of the energy price market. It was developed an infrastructure for continuous evaluation and model training. We delivered a lightweight API that scrapes TGE RDN data and serves point forecasts from a trained LSTM. Building it underscored how non-stationary the market has become: rapid regime shifts and recurring negative prices. The LSTM performs well in “normal” periods but degrades when the training window goes stale, so retraining cadence and data hygiene proved as important as architecture. Key lessons:

- Accuracy drops with time, probably after grid modifications, which are not directly considered as a feature of the model; continuous, periodic retraining with increasing drift is inevitable
- Negative prices require careful scaling/normalization and loss selection; careful hyperparameter/loss selection is crucial
- Use versioned models, a last-known-good fallback, health checks, and alerts on data freshness and large forecast errors.

We decided to change Scenario 2 from inter-cloud integration to mixed cloud integration because of very high networking egress cost.

Pilot 3 – High Performance Computing Platform for Connected and Cooperative Mobile Machinery to improve CO2 footprint

Pilot 3 has integrated aerOS with field robots and tractors to run AI vision and coordination across the continuum; lab and field tests improved real-time processing and used AI-derived prescription maps to drive targeted spraying, yielding about 40% CO₂ reduction for diesel and electric platforms when operating in coordinated swarms.

PILOT 3		2024											2025								
Code	**Name**	M19	M20	M21	M22	M23	M24	M25	M26	M27	M28	M29	M30	M31	M32	M33	M34	M35	M36	M37	M38
Business Process 1 (Scenario 1) - Cooperation large Scale Production		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■				
Setup & Procurement Activities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■				
P3-BP1-SA2	Setup Ethernet-based ECU platform prototype	▨	▨	▨	▨	▨	▨	▨	▨												
P3-BP1-SA3	Setup SESAM 2							▨	▨	▨	▨	▨	▨	▨	▨	▨					
P3-BP1-SA4	Setup self-propelled sprayer	▨	▨	▨																	
P3-BP1-SA8	Setup Ethernet Camera	▨	▨	▨																	
P3-BP1-SA9	Procurement Computing Node (on premise domain)								▨	▨	▨	▨									
P3-BP1-SA10	Setup Computing Node (on premise domain)								▨	▨	▨	▨	▨	▨	▨	▨	▨				
P3-BP1-SA11	Setup Extended SESAM 2 with consumption analytics (iterative with A2.1.3)								▨	▨	▨	▨	▨	▨	▨	▨	▨				
P3-BP1-SA12	Setup Adapted Computing Node (iterative with A2.1.11)	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨									
Development Actvities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■			
P3-BP1-DA1	Development Image Processing Tool	▨	▨																		
P3-BP1-DA2	Development Spraying Adaptation Application	▨	▨																		
P3-BP1-DA3	Development Tracking and Navigation Application			▨	▨	▨	▨	▨	▨	▨	▨	▨									
P3-BP1-DA4	Ethernet-based ECU Platform Prototype setup and integration											▨	▨	▨	▨	▨	▨	▨			
Integration Actvities												■	■	■	■	■	■	■	■	■	■

P3-BP1-IA2	aerOS Basic components														▨	▨	▨	▨			
P3-BP1-IA3	aerOS Non Basic components														▨	▨	▨	▨			
P3-BP1-IA4	Integrating AI Models for field operation and orchestration (using AI for weed detection)																		▨	▨	▨
Validation Activities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
P3-BP1-VA1	*KPI Validation (Lab)*	░	░	░	░	░	░	░	░	░	░	░	░	░	░	░	░	░	▨	▨	▨
P3-BP1-VA2	*KPI Validation (Real)*	░	░	░	░	░	░	░	░	░	░	░	░	░	░	░	░	░	▨	▨	▨
P3-BP1-VA3	*KPI TTC Validation*	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨
Business Process 2 (Scenario 2) - Basis for CO2 neutral intelligent operations									■	■	■	■	■	■	■	■	■	■	■	■	■
Setup & Procurement Activities									■	■	■	■	■	■	■	■	■	■			
P3-BP2-SA1	Setup Extended SESAM 2 with consumption analytics (iterative with A2.1.3)								▨	▨	▨	▨	▨	▨	▨	▨	▨				
P3-BP2-SA2	Setup Adapted Computing Node (iterative with A2.1.11)								▨	▨	▨	▨	▨	▨	▨	▨	▨	▨			
Development Actvities									■	■	■	■	■	■	■	■	■	■	■	■	
P3-BP2-DA1	Vehicle Configuration Adaptation Tool								▨	▨	▨	▨	▨	▨	▨	▨	▨	▨			
P3-BP2-DA2	Machine Analysis AI Engine Application															▨	▨	▨	▨	▨	
Integration Actvities																■	■	■	■	■	■
P3-BP2-IA1	Integrating AI Models for field operation and orchestration (using AI for weed detection)															▨	▨	▨	▨	▨	
P3-BP2-IA2	Simulate data orchestration for CO2 reduction (by integrating low latency networks)																			▨	▨
Validation Activities																				■	■
P3-BP2-VA1	*KPI Validation (Lab)*																			▨	▨
P3-BP2-VA2	*KPI Validation (Real)*																			▨	▨

For Pilot 3, the activities carried out are related to the KPI validation. This activity will be described in detail below.

Pilot 3 has undergone a revision in its implementation timeline due to the seasonal constraints of agricultural operations. Specifically, the **field validation activities** have been rescheduled to take place in **August and September**, aligning with the operational window of the demonstration field. This adjustment was necessary because certain critical activities in the field can only be performed during this period.

The key activities dependent on this timeframe include:

- **Setup and calibration of the autonomous sprayer system**
- **Execution of AI-based weed detection under real field conditions**
- **Validation of the georeferenced application map through live spraying trials**

Pilot 3 has successfully executed the setup and procurement activities, including the configuration of the Sesam Tractor and the deployment of the computing node. In addition, the pilot has driven the development of key applications such as the spraying application and the tracking and navigation application, as well as the integration of both aerOS basic components and non-basic components, as outlined in Report 5.4 for the two defined business processes.

This work supports Scenario 1: Cooperation in Large-Scale Production and Scneario 2: CO_2-Neutral Intelligent Operation, encompassing the physical hardware setup and the integration of all necessary aerOS components. The following chapter presents the results of KPI validation for the pilot, covering both controlled laboratory conditions and real-world field scenarios.

This validation report addresses critical Key Performance Indicators (KPIs) within the scope of the aerOS deployment for connected mobile machinery:

- **KPI 2.3.1.a** evaluates the **computational performance of edge devices** in mobile machinery **without AI-supported applications**, with a target of achieving **more than 20% improvement** over baseline capabilities.
- **KPI 2.3.1.b** assesses the **connectivity performance** of vehicles operating in rural or GPS-dead zones, aiming to establish **high-bandwidth communication** (e.g., 4G/5G) via temporary network infrastructure.
- **KPI 2.3.2** Performance using AI-supported application(s) to monitor and optimize the integration of AI-based solutions to enhance vehicle efficiency, and overall performance. This KPI helps to identify areas for improvement, to fine-tune the AI algorithms, and ensure a seamless operation experience for end users.
- **KPI 2.3.3** CO2 indicators to measure and track the CO2 emissions and subsequent reduction due to the utilization of electric tractors and the aerOS services. Here in particular for the spraying activity during pesticide applications.

Pilot 3 – Business Process 1 – Activity - 23 (P3-BP1-VA1): KPI validation (Lab)

Lab Validation KPI 2.3.1.a and KPI 2.3.1.b

Modern agricultural and construction machinery increasingly rely on edge computing to support complex applications under harsh environmental conditions. KPI 2.3.1.a reflects the need to quantify the engineering effort required to deliver robust computational platforms capable of handling demanding workloads without AI acceleration.

KPI 2.3.1.b supports the broader aerOS objective of enabling an **edge-to-cloud continuum**, where reliable connectivity is essential for real-time data exchange, remote orchestration, and distributed intelligence. Measuring sustained network availability and bandwidth in rural settings provides insight into infrastructure readiness and system resilience.

Target Performance Metrics

- **Computational Targets** KPI 2.3.1.a:
    - GPU: 12.6 FP16 TFLOPS
    - CPU: SPEC int 2k6: 22; SPEC int rate: 140 Gflops
- **Connectivity Target** KPI 2.3.1.b:
    - Availability of 4G/5G network in rural test environments

Validation Setup and Prerequisites

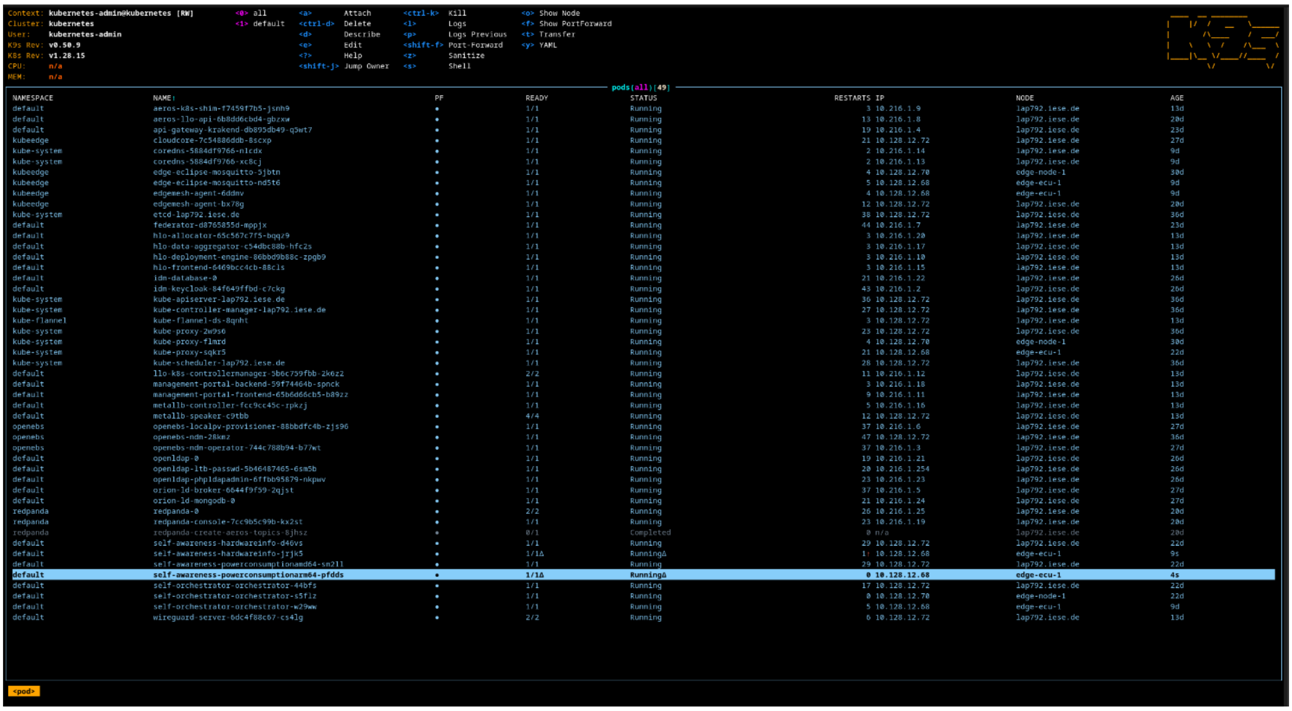
To validate these KPIs, the following prerequisites were fulfilled:

- Assembly and testing of prototype hardware platforms
- Integration of operating systems and required libraries
- Establishment of interfaces between hardware and prototype vehicles
- Deployment of aerOS components including Service Fabric, Data Fabric, Federated Orchestration, and auxiliary services

Evaluation Methodology

Performance Evaluation was conducted using the TTControl platform and HPCP (High-Performance Computing Platform) prototypes enhanced with NVIDIA-based packages. These systems were deployed on prototype John Deere machines running aerOS software.

Figure 57: TTTech Hardware build in the tractor



```
Context: kubernetes-admin@kubernetes [RW]   <0> all       <a>       Attach        <ctrl-k>  Kill           <o> Show Node
Cluster: kubernetes                         <1> default   <ctrl-d>  Delete        <l>       Logs           <f> Show PortForward
User:    kubernetes-admin                                 <d>       Describe      <p>       Logs Previous  <t> Transfer
K9s Rev: v0.50.9                                          <e>       Edit          <shift-f> Port-Forward   <y> YAML
K8s Rev: v1.28.15                                         <?>       Help          <z>       Sanitize
CPU:     n/a                                              <shift-j> Jump Owner    <s>       Shell
MEM:     n/a
──────────────────────────────────────────────── pods(all)[49] ─────────────────────────────────────────────────
NAMESPACE     NAME↑                                         PF  READY  STATUS     RESTARTS IP            NODE             AGE
default       aeros-k8s-shim-f7459f7b5-jsnh9                ●   1/1    Running           3 10.216.1.9    lap792.iese.de   13d
default       aeros-llo-api-6b8dd6cbd4-gbzxw                ●   1/1    Running          13 10.216.1.8    lap792.iese.de   20d
default       api-gateway-krakend-db895db49-q5wt7           ●   1/1    Running          19 10.216.1.4    lap792.iese.de   23d
kubeedge      cloudcore-7c54886ddb-8scxp                    ●   1/1    Running          21 10.128.12.72  lap792.iese.de   27d
kube-system   coredns-5884df9766-nlcdx                      ●   1/1    Running           2 10.216.1.14   lap792.iese.de   9d
kube-system   coredns-5884df9766-xc8cj                      ●   1/1    Running           2 10.216.1.13   lap792.iese.de   9d
kubeedge      edge-eclipse-mosquitto-5jbtn                  ●   1/1    Running           4 10.128.12.70  edge-node-1      30d
kubeedge      edge-eclipse-mosquitto-nd5t6                  ●   1/1    Running           5 10.128.12.68  edge-ecu-1       9d
kubeedge      edgemesh-agent-6ddmv                          ●   1/1    Running           4 10.128.12.68  edge-ecu-1       9d
kubeedge      edgemesh-agent-bx78g                          ●   1/1    Running          12 10.128.12.72  lap792.iese.de   20d
kube-system   etcd-lap792.iese.de                           ●   1/1    Running          38 10.128.12.72  lap792.iese.de   36d
default       federator-d8765855d-mppjx                     ●   1/1    Running          44 10.216.1.7    lap792.iese.de   23d
default       hlo-allocator-65c567c7f5-bqqz9                ●   1/1    Running           3 10.216.1.20   lap792.iese.de   13d
default       hlo-data-aggregator-c54dbc88b-hfc2s           ●   1/1    Running           3 10.216.1.17   lap792.iese.de   13d
default       hlo-deployment-engine-86bbd9b88c-zpgb9        ●   1/1    Running           3 10.216.1.10   lap792.iese.de   13d
default       hlo-frontend-6469bcc4cb-88cls                 ●   1/1    Running           3 10.216.1.15   lap792.iese.de   13d
default       idm-database-0                                ●   1/1    Running          21 10.216.1.22   lap792.iese.de   26d
default       idm-keycloak-84f649ffbd-c7ckg                 ●   1/1    Running          43 10.216.1.2    lap792.iese.de   26d
kube-system   kube-apiserver-lap792.iese.de                 ●   1/1    Running          36 10.128.12.72  lap792.iese.de   36d
kube-system   kube-controller-manager-lap792.iese.de        ●   1/1    Running          27 10.128.12.72  lap792.iese.de   36d
kube-flannel  kube-flannel-ds-8qnht                         ●   1/1    Running           3 10.128.12.72  lap792.iese.de   13d
kube-system   kube-proxy-2w9s6                              ●   1/1    Running          23 10.128.12.72  lap792.iese.de   36d
kube-system   kube-proxy-flmrd                              ●   1/1    Running           4 10.128.12.70  edge-node-1      30d
kube-system   kube-proxy-sqkr5                              ●   1/1    Running          21 10.128.12.68  edge-ecu-1       22d
kube-system   kube-scheduler-lap792.iese.de                 ●   1/1    Running          28 10.128.12.72  lap792.iese.de   36d
default       llo-k8s-controllermanager-5b6c759fbb-2k6z2    ●   2/2    Running          11 10.216.1.12   lap792.iese.de   13d
default       management-portal-backend-59f74464b-spnck     ●   1/1    Running           3 10.216.1.18   lap792.iese.de   13d
default       management-portal-frontend-65b6d66cb5-b89zz   ●   1/1    Running           9 10.216.1.11   lap792.iese.de   13d
default       metallb-controller-fcc9cc45c-rpkzj            ●   1/1    Running           5 10.216.1.16   lap792.iese.de   13d
default       metallb-speaker-c9tbb                         ●   4/4    Running          12 10.128.12.72  lap792.iese.de   13d
openebs       openebs-localpv-provisioner-88bbdfc4b-zjs96   ●   1/1    Running          37 10.216.1.6    lap792.iese.de   27d
openebs       openebs-ndm-28kmz                             ●   1/1    Running          47 10.128.12.72  lap792.iese.de   36d
openebs       openebs-ndm-operator-744c788b94-b77wt         ●   1/1    Running          37 10.216.1.3    lap792.iese.de   27d
default       openldap-0                                    ●   1/1    Running          19 10.216.1.21   lap792.iese.de   26d
default       openldap-ltb-passwd-5b46487465-6sm5b          ●   1/1    Running          20 10.216.1.254  lap792.iese.de   26d
default       openldap-phpldapadmin-6ffbb95879-nkpwv        ●   1/1    Running          23 10.216.1.23   lap792.iese.de   26d
default       orion-ld-broker-6644f9f59-2qjst               ●   1/1    Running          37 10.216.1.5    lap792.iese.de   27d
default       orion-ld-mongodb-0                            ●   1/1    Running          21 10.216.1.24   lap792.iese.de   27d
redpanda      redpanda-0                                    ●   2/2    Running          26 10.216.1.25   lap792.iese.de   20d
redpanda      redpanda-console-7cc9b5c99b-kx2st             ●   1/1    Running          23 10.216.1.19   lap792.iese.de   20d
redpanda      redpanda-create-aeros-topics-8jhsz            ●   0/1    Completed         0 n/a           lap792.iese.de   20d
default       self-awareness-hardwareinfo-d46vs             ●   1/1    Running          29 10.128.12.72  lap792.iese.de   22d
default       self-awareness-hardwareinfo-jrjk5             ●   1/1Δ   RunningΔ         1↑ 10.128.12.68  edge-ecu-1       9s
default       self-awareness-powerconsumptionamd64-sm2ll    ●   1/1    Running          29 10.128.12.72  lap792.iese.de   22d
default       self-awareness-powerconsumptionarm64-pfdds    ●   1/1Δ   RunningΔ          0 10.128.12.68  edge-ecu-1       4s
default       self-orchestrator-orchestrator-44bfs          ●   1/1    Running          17 10.128.12.72  lap792.iese.de   22d
default       self-orchestrator-orchestrator-s5flz          ●   1/1    Running           0 10.128.12.70  edge-node-1      22d
default       self-orchestrator-orchestrator-w29ww          ●   1/1    Running           5 10.128.12.68  edge-ecu-1       9d
default       wireguard-server-6dc4f88c67-cs4lg             ●   2/2    Running           6 10.128.12.72  lap792.iese.de   13d

<pod>
```

Figure 58: Pilot 3 -List and status of service pods deployed in the Kubernetes cluster of the Laptop/ECU-based aerOS Entry/Field Domain hosting any aerOS basic and additional components relevant

Domain detail

Id	Description	Public Url	Owner	Entrypoint	Status
EntrypointDomain	aerOS Entrypoint Domain	http://10.128.12.72:31566	IESE	✓	Functional

Infrastructure elements:

Hostname	Container Technology	CPU arch	CPU cores	RAM capacity (MB)	Trust score	Status
lap792.iese.de	Kubernetes	x64	8	16477	-1	Ready
edge-ecu-1	Kubernetes	arm64	8	16591	-1	Ready

Figure 59: Pilot 3 - Entry/Field Domain details in the Management Portal

Lab and field tests were executed using real-world application scenarios, with continuous monitoring of CPU, GPU, and memory utilization. The sustainability impact was assessed through application execution metrics.

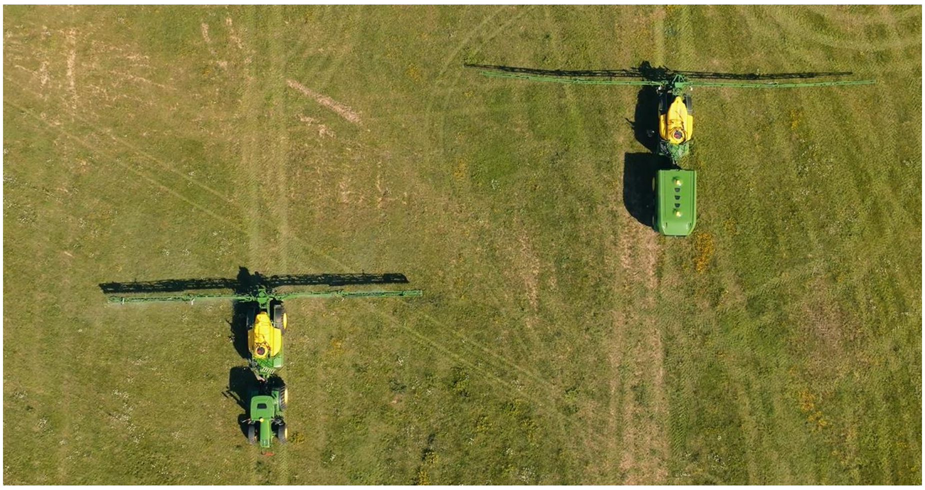
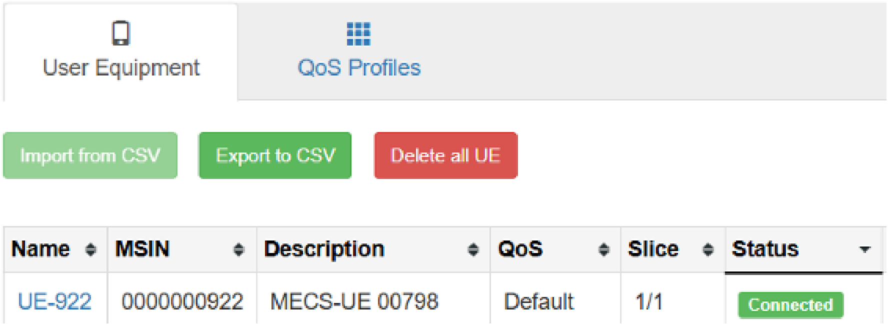


Figure 60: Machine fleet operation

Connectivity Evaluation involved testing temporary network infrastructure using aerOS and John Deere operational applications. The goal was to validate the availability and stability of high-bandwidth connectivity in Mobile coverage-dead zones. Therefore, the MECSware campusXG network was used as connectivity layer.

Figure 61: MECSware campusXG network with Application Server

In the image below it is shown the configuration of the MECSware campusXG network with the connectected use equipment (UE).



Name	MSIN	Description	QoS	Slice	Status
UE-922	0000000922	MECS-UE 00798	Default	1/1	Connected

Figure 62: MECSware campusXG network status page

A systematic latency test was conducted to prove the connectivity in the private 5G network. Results are shown in the table below:

Table 17: Results connectivity test

IP	Icmp_Seq	Ping Time
64 bytes from 192.168.4.202	77	11ms
64 bytes from 192.168.4.202	78	12.1ms
64 bytes from 192.168.4.202	79	10.5ms
64 bytes from 192.168.4.202	80	18.6ms
64 bytes from 192.168.4.202	81	13.2ms
64 bytes from 192.168.4.202	82	15.3ms

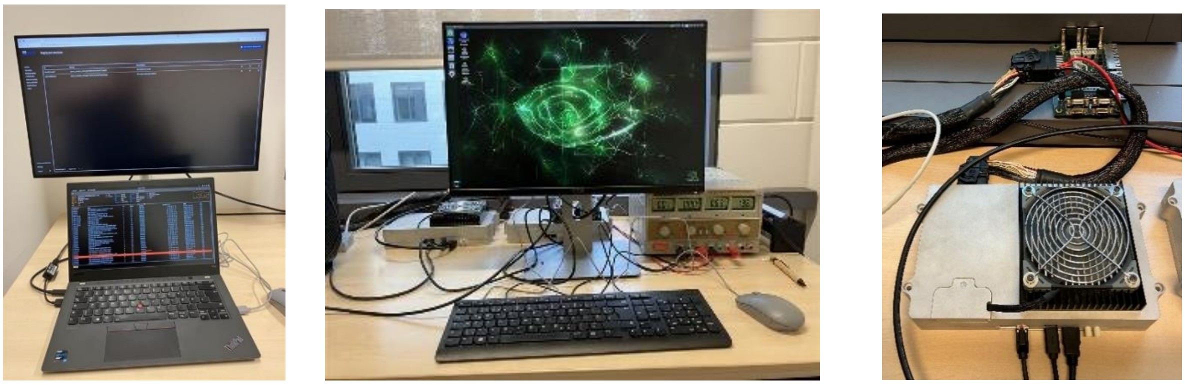
Measurement Results

- **Baseline (Pre-aerOS Deployment)**:
    - GPU: 2×128 GFLOPS FP16

  - CPU: 26000 DMIPS
  - No network connectivity available

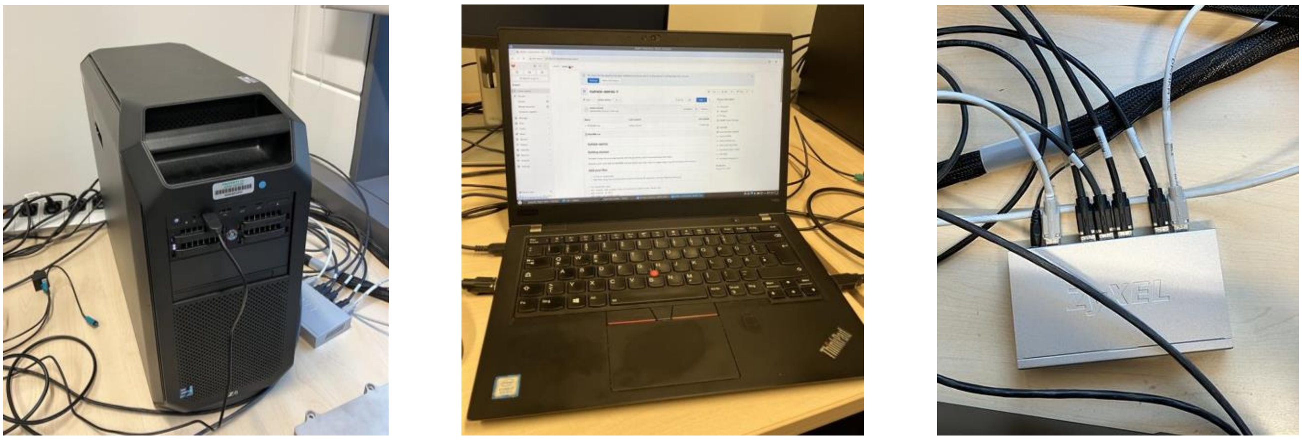
- **M24 (Deliverable D5.5)**:
  - GPU: 12.6 FP16 TFLOPS
  - CPU: SPEC int 2k6: 22; SPEC int rate: 140 Gflops
  - 4G/5G network successfully deployed and operational

The hardware prototypes are fully assembled and operational, actively used by TTControl and John Deere in laboratory settings.



IESE Laptop

TTControl ECU



JD Workstation

IESE Laptop (GitLab)

Ethernet Switch

Figure 63: Laboratory testbed environment including all infrastructure components installed and configured for KPI evaluation of the aerOS-based Pilot 3 use case in scenarios 1 and 2.

The performance targets were met, confirming the successful validation of KPI 2.3.1.a. Similarly, the deployment of temporary 4G/5G infrastructure validated KPI 2.3.1.b, demonstrating the feasibility of high-bandwidth connectivity in rural environments.

Quantitative Validation of KPI 2.3.2

This section focuses on the quantitative evaluation of the aerOS-based AI service distribution. The AI model, as described in WP.5.4 (P3-BP1-DA-13), is an AI-based weed detection method uses **object detection** to identify and localize weeds in images via bounding boxes and class labels, enabling precise control tasks like targeted spraying. Unlike simple image classification, this approach provides spatial localization, which is crucial for selective agricultural interventions.

The system employs **YOLO (You Only Look Once)** for real-time image processing, with cameras mounted on tractors or drones capturing data. The AI processes these images, detects weeds, and generates **georeferenced application maps** using GNSS data. These maps guide autonomous spraying systems for efficient field operations.

A series of benchmarks were conducted to assess the system's performance, particularly focusing on the trade-offs between throughput and latency under various configurations. The key variable adjusted across these tests was the batch size, which dictates how many video frames are processed together in a single cycle from the Edge to the Cloud and back.

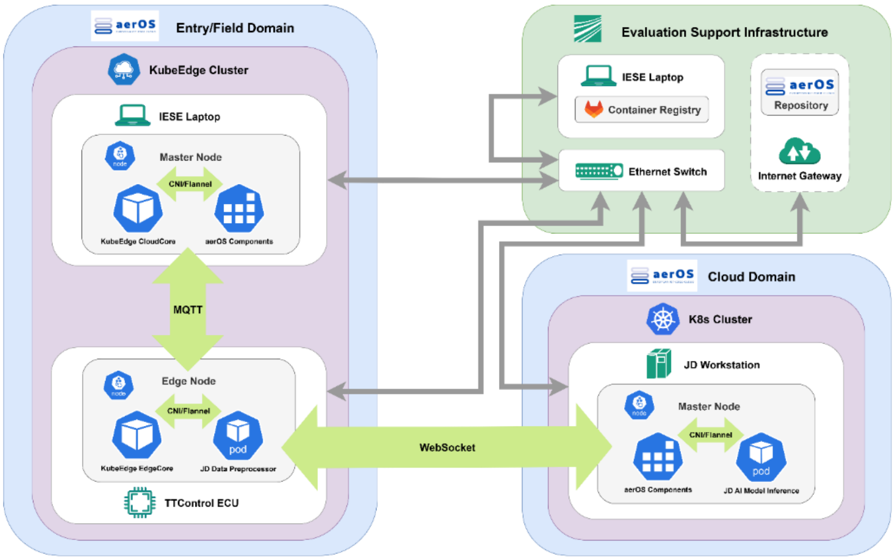


Figure 64: Pilot 3 Architecture

The primary metrics evaluated were:

- **Frames per second (FPS):** Measuring the overall processing throughput of the system.
- **Round-Trip Time (RTT):** The latency for a batch of frames to be processed on the edge device, sent to the cloud for AI inference, and have the results returned.
- **Edge and Cloud Processing Times:** The time spent on data preprocessing (Edge) and AI model inference (Cloud).

The following sections detail the results for each tested batch size.

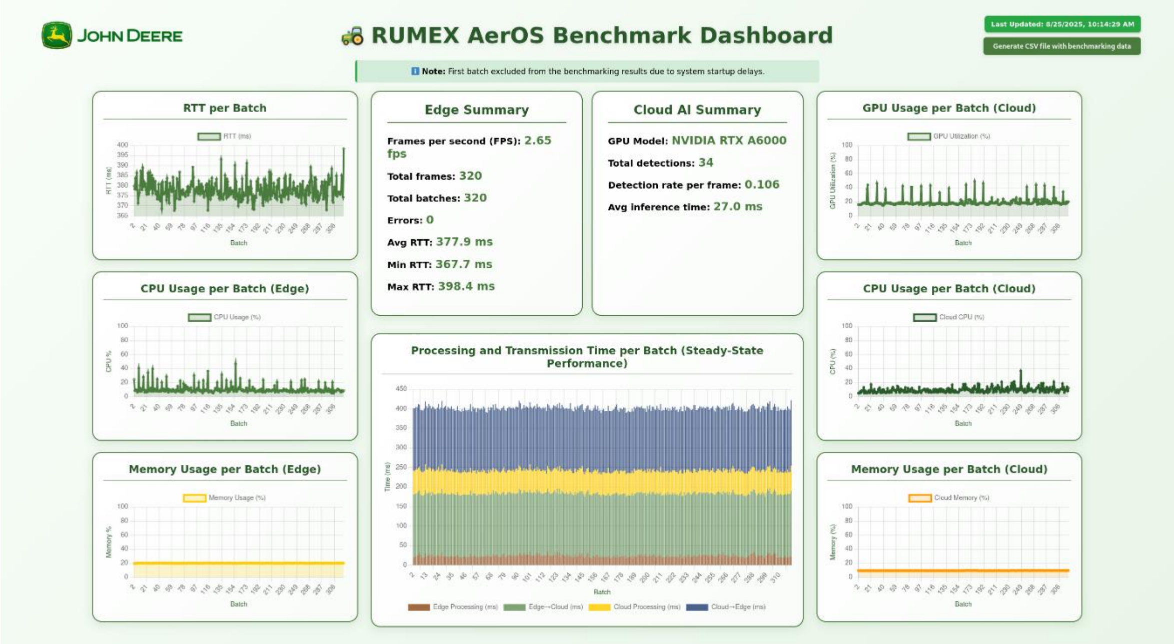
Benchmark 1: Batch Size 1

With a batch size of 1, each frame is processed individually, resulting in the lowest possible latency per frame but also the lowest throughput. Resource utilization reflects this intermittent workload; the Cloud GPU shows sharp, frequent spikes up to approximately 70% during inference, while CPU usage on both Edge and Cloud remains low, generally below 40%.

Table 18: Pilot 3 Benchmark 1

Metric	Value
Frames per second (FPS)	2.65 fps
Average RTT	377.9 ms

Min/Max RTT	367.7 ms / 398.4 ms
Average Edge Processing Time	24.3 ms
Average Cloud Processing Time	58.8 ms
Average Inference Time	27.0 ms


JOHN DEERE

RUMEX AerOS Benchmark Dashboard

Last Updated: 8/25/2025, 10:14:29 AM

Generate CSV file with benchmarking data

Note: First batch excluded from the benchmarking results due to system startup delays.

Edge Summary

Frames per second (FPS): 2.65 fps

Total frames: 320

Total batches: 320

Errors: 0

Avg RTT: 377.9 ms

Min RTT: 367.7 ms

Max RTT: 398.4 ms

Cloud AI Summary

GPU Model: NVIDIA RTX A6000

Total detections: 34

Detection rate per frame: 0.106

Avg inference time: 27.0 ms

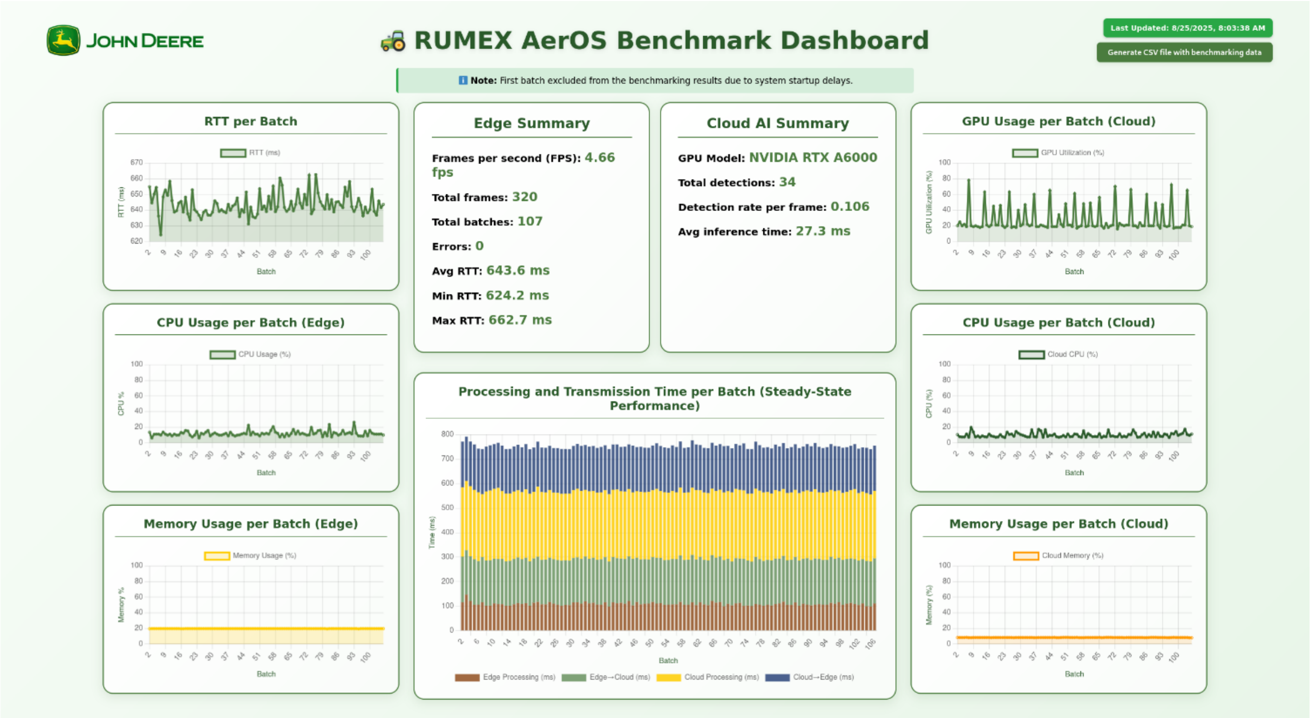
Figure 65: Pilot 3 Benchmark Dashboard for Batch Size 1

Benchmark 2: Batch Size 3

Increasing the batch size to 3 improves the frames processed per second, indicating higher efficiency. As the batch size increases, the Cloud GPU utilization pattern shows slightly wider and less frequent peaks, still reaching around 70%. CPU usage on both the Edge and Cloud systems remains low and stable.

Table 19: Pilot 3 Benchmark 2

Metric	Value
Frames per second (FPS)	4.66 fps
Average RTT	643.6 ms
Min/Max RTT	624.2 ms / 662.7 ms
Average Edge Processing Time	109.1 ms
Average Cloud Processing Time	276.4 ms
Average Inference Time	27.3 ms



JOHN DEERE

RUMEX AerOS Benchmark Dashboard

Last Updated: 8/25/2025, 8:03:38 AM

Generate CSV file with benchmarking data

Note: First batch excluded from the benchmarking results due to system startup delays.

Edge Summary

Frames per second (FPS): 4.66 fps

Total frames: 320

Total batches: 107

Errors: 0

Avg RTT: 643.6 ms

Min RTT: 624.2 ms

Max RTT: 662.7 ms

Cloud AI Summary

GPU Model: NVIDIA RTX A6000

Total detections: 34

Detection rate per frame: 0.106

Avg inference time: 27.3 ms

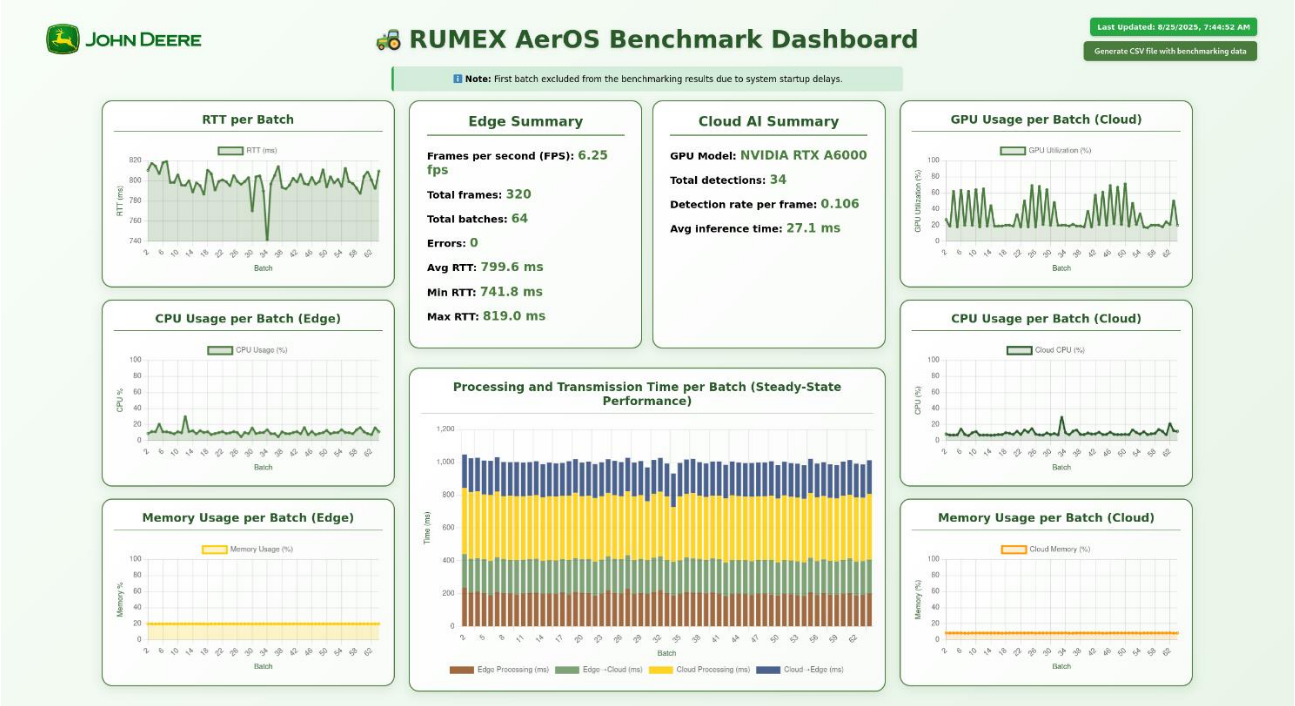
Figure 66: Pilot 3 Benchmark Dashboard for Batch Size 3

Benchmark 3: Batch Size 5

A batch size of 5 continues the trend of increasing throughput while also increasing the overall latency. The Cloud GPU continues to show a pattern of utilization spikes reaching ~70% that become wider with the larger batch size. CPU usage on both the Edge and Cloud devices remains moderate, typically under 40%.

Table 20: Pilot 3 Benchmark 3

Metric	Value
Frames per second (FPS)	4.66 fps
Average RTT	643.6 ms
Min/Max RTT	624.2 ms / 662.7 ms
Average Edge Processing Time	109.1 ms
Average Cloud Processing Time	276.4 ms
Average Inference Time	27.3 ms


JOHN DEERE

RUMEX AerOS Benchmark Dashboard

Last Updated: 8/25/2025, 7:44:52 AM

Generate CSV file with benchmarking data

Note: First batch excluded from the benchmarking results due to system startup delays.

Edge Summary

Frames per second (FPS): 6.25 fps

Total frames: 320

Total batches: 64

Errors: 0

Avg RTT: 799.6 ms

Min RTT: 741.8 ms

Max RTT: 819.0 ms

Cloud AI Summary

GPU Model: NVIDIA RTX A6000

Total detections: 34

Detection rate per frame: 0.106

Avg inference time: 27.1 ms

Figure 67: Pilot 3 Benchmark Dashboard for Batch Size 5

Summary of Findings

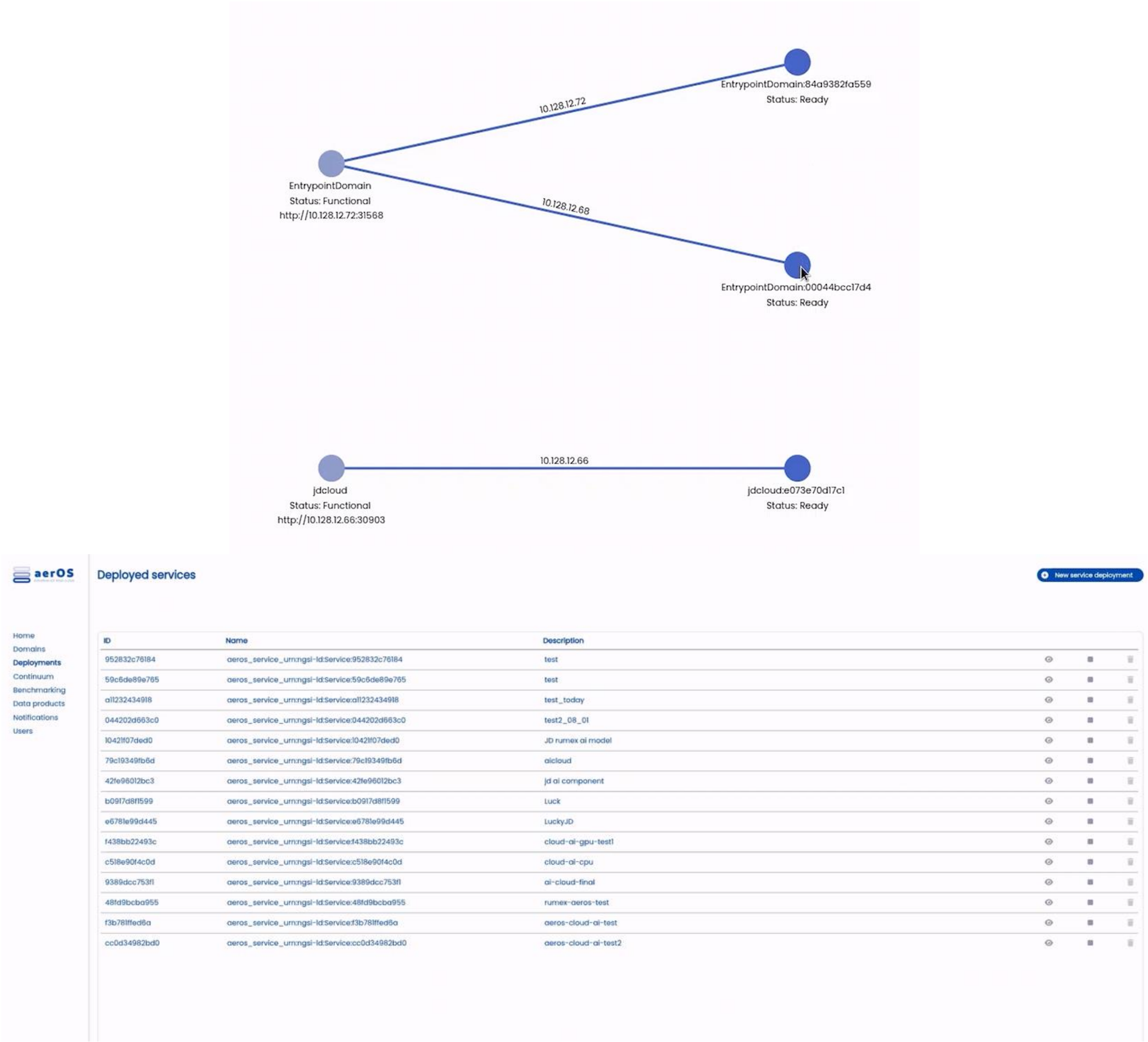
The benchmark results demonstrate a clear and predictable relationship between batch size, processing throughput (FPS), and round-trip time (RTT). Increasing the batch size allows the system to process a higher number of frames per second, improving overall throughput. However, this comes at the cost of increased RTT, as more time is spent collecting and processing frames in a single, larger batch, leading to higher latency for the results of that batch.

Notably, the average inference time for the AI model in the cloud remained consistent at approximately 27 ms across all tests, indicating that the AI processing is not the bottleneck. The primary factor influencing RTT is the time required for edge processing and data transmission, which scales with the batch size. The optimal batch size will therefore depend on the specific requirements of the use case, balancing the need for high throughput against the application's tolerance for latency.

Pilot 3 – Business Process 1 – Activity - 24 (P3-BP1-VA2): KPI Validation (Real)

KPI 2.3.2 Swarm of vehicle performance improvement

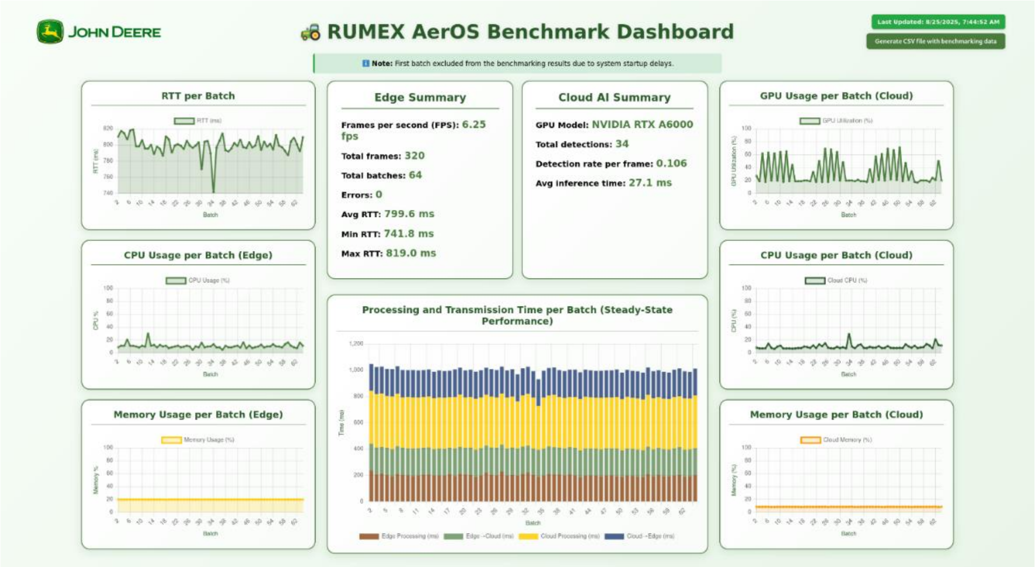
Based on the evaluation for the laboratory environment, a monitoring framework was established to evaluate key performance parameters, including frames per second (FPS), image processing time, and round-trip latency. This setup enabled controlled testing of various AI model configurations and image resolutions within a laboratory environment. By leveraging the aerOS Edge-Cloud Continuum, the system was successfully optimized to achieve a 20% increase in FPS, enhancing real-time processing capabilities.



Deployed services

ID	Name	Description
952832c76184	aeros_service_urn:ngsi-ld:Service:952832c76184	test
59c6de89e765	aeros_service_urn:ngsi-ld:Service:59c6de89e765	test
a11232434918	aeros_service_urn:ngsi-ld:Service:a11232434918	test_today
044202d663c0	aeros_service_urn:ngsi-ld:Service:044202d663c0	test2_08_01
10421f07ded0	aeros_service_urn:ngsi-ld:Service:10421f07ded0	JD rumex ai model
79c19349fb6d	aeros_service_urn:ngsi-ld:Service:79c19349fb6d	aicloud
42fe96012bc3	aeros_service_urn:ngsi-ld:Service:42fe96012bc3	jd ai component
b0917d8f1599	aeros_service_urn:ngsi-ld:Service:b0917d8f1599	Luck
e6781e99d445	aeros_service_urn:ngsi-ld:Service:e6781e99d445	LuckyJD
f438bb22493c	aeros_service_urn:ngsi-ld:Service:f438bb22493c	cloud-ai-gpu-test1
c518e90f4c0d	aeros_service_urn:ngsi-ld:Service:c518e90f4c0d	cloud-ai-cpu
9389dcc753f1	aeros_service_urn:ngsi-ld:Service:9389dcc753f1	ai-cloud-final
48fd9bcba955	aeros_service_urn:ngsi-ld:Service:48fd9bcba955	rumex-aeros-test
f3b781ffed6a	aeros_service_urn:ngsi-ld:Service:f3b781ffed6a	aeros-cloud-ai-test
cc0d34982bd0	aeros_service_urn:ngsi-ld:Service:cc0d34982bd0	aeros-cloud-ai-test2

Figure 68: Pilot 3 - aerOS Portal continuum view and deployed services



RUMEX AerOS Benchmark Dashboard

Note: First batch excluded from the benchmarking results due to system startup delays.

Edge Summary

Frames per second (FPS): 6.25 fps
Total frames: 320
Total batches: 64
Errors: 0
Avg RTT: 799.6 ms
Min RTT: 741.8 ms
Max RTT: 819.0 ms

Cloud AI Summary

GPU Model: NVIDIA RTX A6000
Total detections: 34
Detection rate per frame: 0.106
Avg inference time: 27.1 ms

Figure 69: Pilot 3 Benchmark Dashboard

Following the successful verification of the laboratory configuration, the system was integrated into our field operations. During this phase, the same performance parameters—such as frames per second (FPS), image processing time, and round-trip latency—were systematically measured under real-world conditions.

This allowed to validate the consistency and reliability of the setup outside the controlled lab environment and confirm that the optimizations achieved during testing translated effectively to operational use.

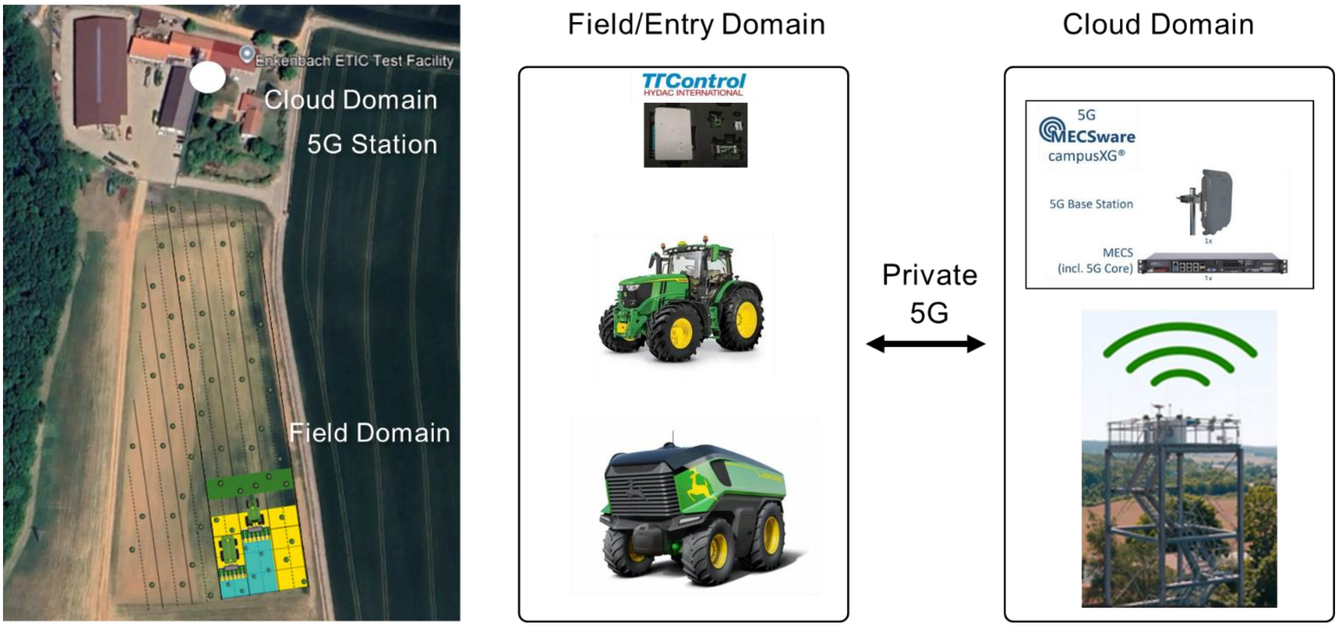


Figure 70: Pilot 3 Field Setup

.

KPI 2.3.3 CO2 emissions reduction thanks to platooning (TTC)

For the evaluation, a spatially accurate prescription map was generated using the aerOS edge-cloud continuum and AI-based image analysis (e.g., from satellite, drone, or tractor-mounted cameras) that identifies:

- Weed or pest hotspots
- Crop health variability
- Soil moisture or nutrient zones

This map is then used to control spraying intensity and location for multiple machines.

Figure 71: Example of prescription map on John Deere Display

Thanks to the implementation of the aerOS components we could measure the following results for 40% CO_2 reduction for diesel and electric tractors in a swarm environment.

Figure 72: Pilot 3 CO2 reduction overview

The following setup was conducted:

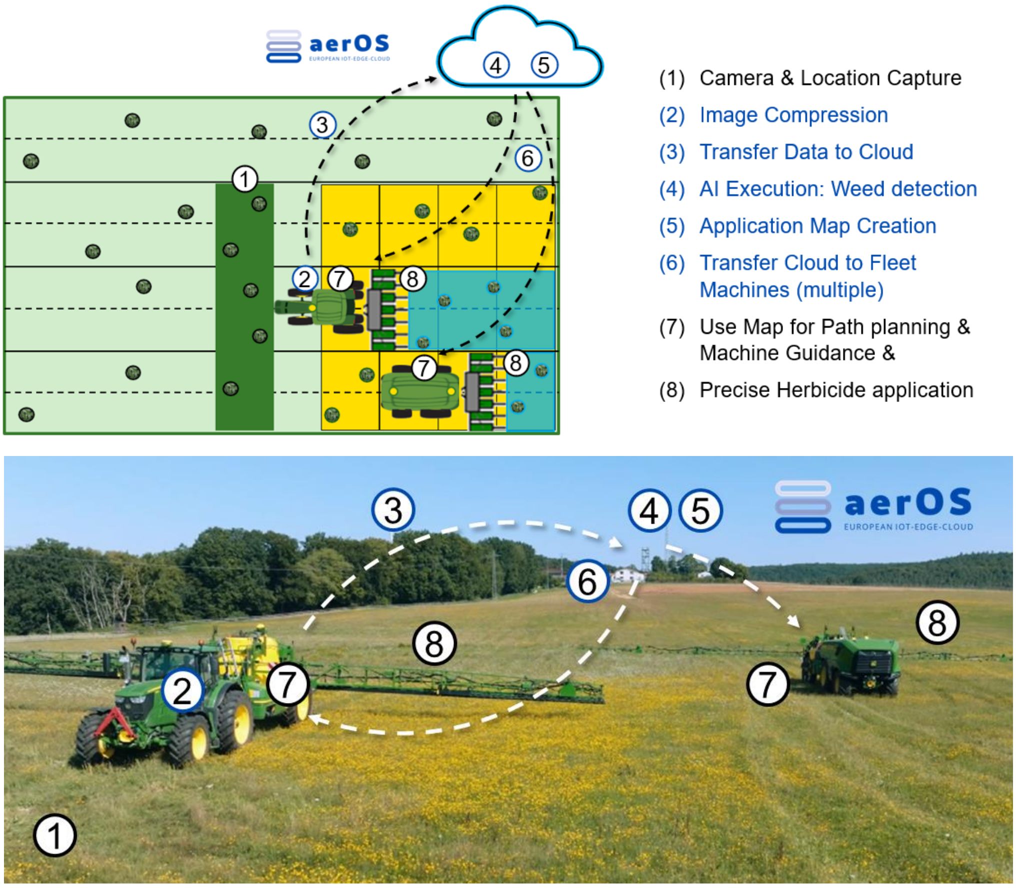


Figure 73: Pilot 3 aerOS Field Integration Overview

The following data was captured on the machines to evaluate the KPI:

Assuming a baseline CO_2 emission of 41.3 kg/ha for diesel tractor spraying:

- With GPS guidance there was an achievement of **20% reduction**: 41.3×0.80=33.0 kg CO_2/ha | 41.3-33.0 kg CO_2/ha → Savings: ~8.3 kg CO_2/ha

- With basic overlap reduction due to capturing the already deployed herbicides there was an achievement of **2–7%** reduction: 41.3×0.93=38.4 kg CO_2/ha | 41.3-38,4 kg CO_2/ha → Savings: ~2.9 kg CO_2/ha
- With swarm coordinated ground spraying and task distribution a reduction of **15%** was achieved: 41.3×0.85=35.1 kg CO_2/ha | 41.3-35,1 kg CO_2/ha → Savings: ~6.2 kg CO_2/ha

Conclusion:
The integration of aerOS components and AI-based image analysis enabled the creation of a spatially precise prescription map, identifying weed and pest hotspots, crop health variability, and soil moisture or nutrient zones. This map facilitated targeted, multi-machine spraying, resulting in significant environmental benefits. In swarm operation with diesel and electric tractors, CO_2 emissions were reduced by up to 40%. Individual measures such as GPS guidance, overlap reduction, and coordinated task distribution contributed reductions of 20%, 7%, and 15%, respectively, demonstrating the effectiveness of aerOS in sustainable precision agriculture.

Pilot 4 – Smart edge services for the Port Continuum

Pilot 4 migrated from a single server to a multi-domain aerOS setup spanning cloud, on-prem, and far-edge nodes; predictive maintenance runs on cranes and straddle carriers, while Jetson-based computer vision at the edge performs container ID/damage/seal checks, with distributed order management and ROS planners completing the operational loop.

Pilot 4		**2024**											**2025**								
Code	**Name**	**M19**	**M20**	**M21**	**M22**	**M23**	**M24**	**M25**	**M26**	**M27**	**M28**	**M29**	**M30**	**M31**	**M32**	**M33**	**M34**	**M35**	**M36**	**M37**	**M38**
Business Process (BP) 1 (Scenario 1) - Predictive maintenance of Container Handling Equipment		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Setup & Procurement Activities		■	■	■	■	■	■														
P4-BP1-SA1	A1 - STS Cranes PLCs	🟦	🟦	🟦	🟦	🟦	🟦														
P4-BP1-SA2	A2 - Straddle Carriers PLCs	🟦	🟦	🟦	🟦	🟦	🟦														
P4-BP1-SA3	A3 - Straddle Carriers sensors		🟦	🟦	🟦	🟦	🟦														
P4-BP1-SA4	A4 - Straddle Carriers GPSs		🟦	🟦	🟦	🟦	🟦														
P4-BP1-SA5	A5 - Straddle Carriers Human Machine Interfaces		🟦	🟦	🟦	🟦	🟦														
P4-BP1-SA6	A6 - Straddle Carriers 4G Routers		🟦	🟦	🟦	🟦	🟦														
P4-BP1-SA7	A7 - Straddle Carriers and STS IoT Gateways (IE1-IE4)	🟦	🟦	🟦	🟦	🟦	🟦														
P4-BP1-SA8	A8 - EUROGATE Domain Server (IE5)				🟦	🟦	🟦														
P4-BP1-SA9	A9 - Entrypoint domain Server (IE0)					🟦	🟦														
Development Activities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
P4-BP1-DA1	A10 - PLC data gathering						🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦							
P4-BP1-DA2	A11 - GPS and sensors data gathering						🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦							
P4-BP1-DA3	A12 - TOS data acquisition																🟥	🟥	🟥	🟥	🟥
P4-BP1-DA4	A13 - CMMS data acquisition														🟦	🟦	🟦	🟦	🟦	🟦	🟦
P4-BP1-DA5	A14 - STS cranes AI models training				🟦	🟦	🟦	🟦				🟦	🟦	🟦	🟦			🟦	🟦	🟦	🟦
P4-BP1-DA6	A15 - Straddle carriers AI models training								🟦	🟦	🟦	🟦			🟦	🟦	🟦	🟦			
P4-BP1-DA7	A16 - GIS cartography generation	🟦	🟦	🟦																	
Integration Activities							■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
P4-BP1-IA1	A17 - aerOS core services integration into etrypoint domain (IE0)						🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦					

P4-BP1-IA2	A18 - aerOS core services integration into EUROGATE domain (IE5)							🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦			
P4-BP1-IA3	A97 - aerOS core services integration into EUROGATE domain (IE1-IE4)													🟦	🟦	🟦	🟦	🟦	🟦		
P4-BP1-IA4	A20 - aerOS auxiliary services integration into entrypoint domain (IE0)																🟦	🟦	🟦	🟦	🟦
Validation Activities									■	■	■	■	■	■	■	■	■	■	■	■	■
P4-BP1-VA1	A21 - Data acquisition								🟦	🟦	🟦	🟦	🟦	🟦							
P4-BP1-VA2	A22 - Data storage									🟦	🟦	🟦	🟦	🟦	🟦						
P4-BP1-VA3	A23 - STS and Straddle carriers AI model inference verification																	🟦	🟦	🟦	🟦
P4-BP1-VA4	A24 - aerOS entrypoint domain - EUROGATE domain communication															🟦	🟦	🟦	🟦	🟦	🟦
BP2 (Scenario 2) - Risk prevention via Computer Vision in the edge		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Setup & Procurement Activities		■	■	■	■	■	■						■	■	■						
P4-BP2-SA1	A25 - EUROGATE Cameras	🟦	🟦	🟦	🟦	🟦	🟦														
P4-BP2-SA2	A26 - EUROGATE NVR	🟦	🟦	🟦	🟦	🟦	🟦														
P4-BP2-SA3	A27 - Jetson Orin (IE6-IE7)												🟦	🟦	🟦						
P4-BP2-SA4	A28 - CUT Domain Server (IE8)					🟦	🟦														
Development Activities		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
P4-BP2-DA1	A29 - Video collection					🟦	🟦	🟦			🟦	🟦	🟦			🟦	🟦	🟦			
P4-BP2-DA2	A30 - Container ID recognition model training	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦												
P4-BP2-DA3	A31 - Container damage recognition model training								🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦					
P4-BP2-DA4	A32 - Container seal recognition model training															🟦	🟦	🟦	🟦	🟦	🟦
P4-BP2-DA5	A33 - Yard inventory damaged container dashboard																🟦	🟦	🟦	🟦	🟦
Integration Activities												■	■	■	■	■	■	■	■	■	
P4-BP2-IA1	A34 - aerOS core services integration into CUT domain (IE6)											🟦	🟦	🟦	🟦	🟦					
P4-BP2-IA2	A35 - aerOS core services integration into Jetson Orin (IE7-IE8)															🟦	🟦	🟦	🟦	🟦	
Validation Activities										■	■	■	■	■	■	■	■	■	■	■	■
P4-BP2-VA1	A36 - Video storage									🟦	🟦	🟦	🟦	🟦	🟦						
P4-BP2-VA2	A37 - CV model inference verification																		🟦	🟦	🟦
P4-BP2-VA3	A38 - aerOS entrypoint domain - CUT domain communication															🟦	🟦	🟦	🟦	🟦	

For Pilot 4, the activities carried out are related to the data acquisition and storage, the STS and Straddle Carriers AI model inference verification, the communication between aerOS entrypoint domain and EUROGATE domain, the video storage and the communication between aerOS entrypoint domain and CUT domain. All of these activities will be described in detail below.

Pilot 4 – Business Process 1 – Activity - 21 (P4-BP1-VA1): Data acquisition

Different testbenches have been performed for the verification of data acquisition from the different data sources, i.e., PLCs from STS and straddle carriers, GPS, and straddle carrier sensors. All of them are captured through custom-made data acquisition flows with a Node-RED no-code tool on the edge (i.e., in the different IEs of the CHEs under consideration). Some of these flows are slightly presented in the following figures:

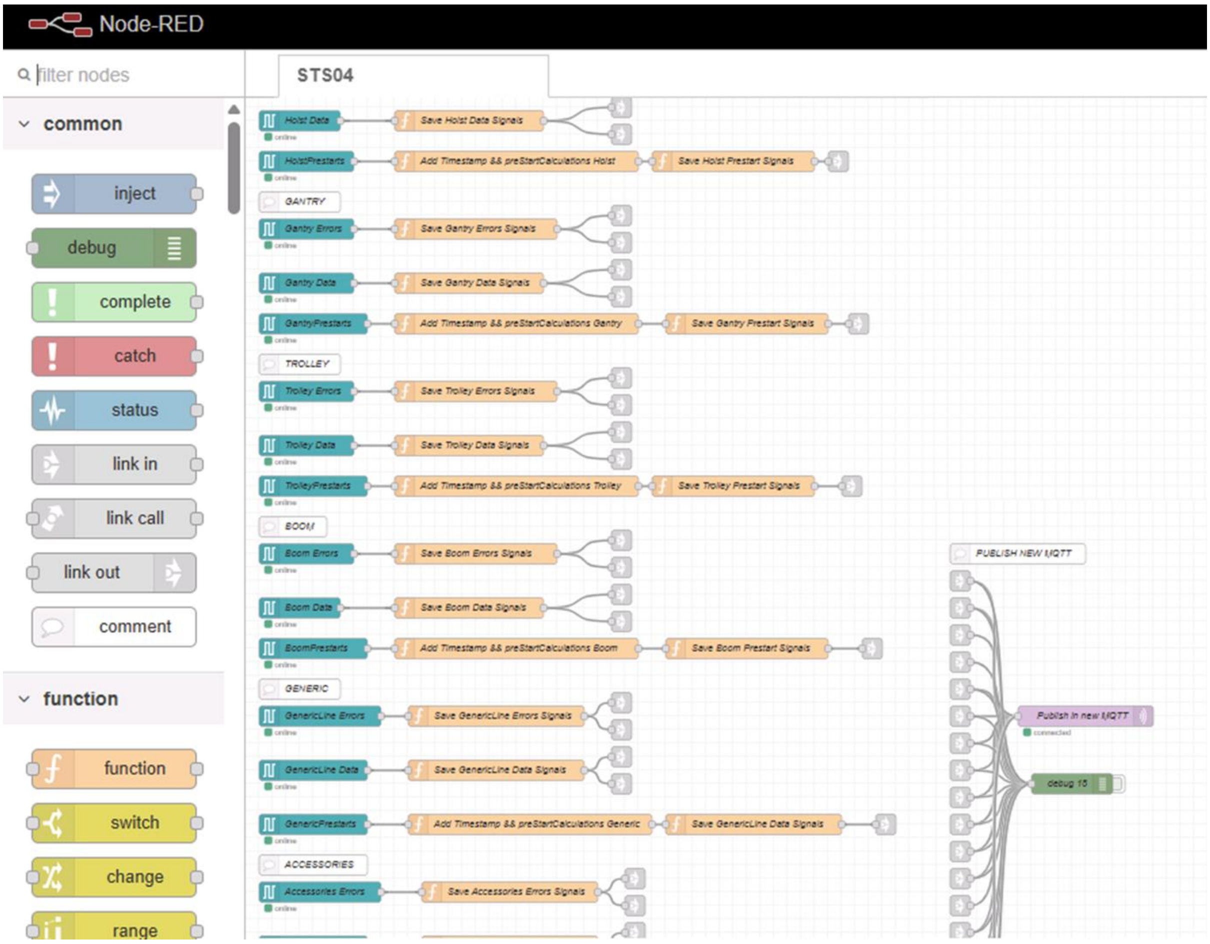


Figure 74: STS-04 data acquisition flows.

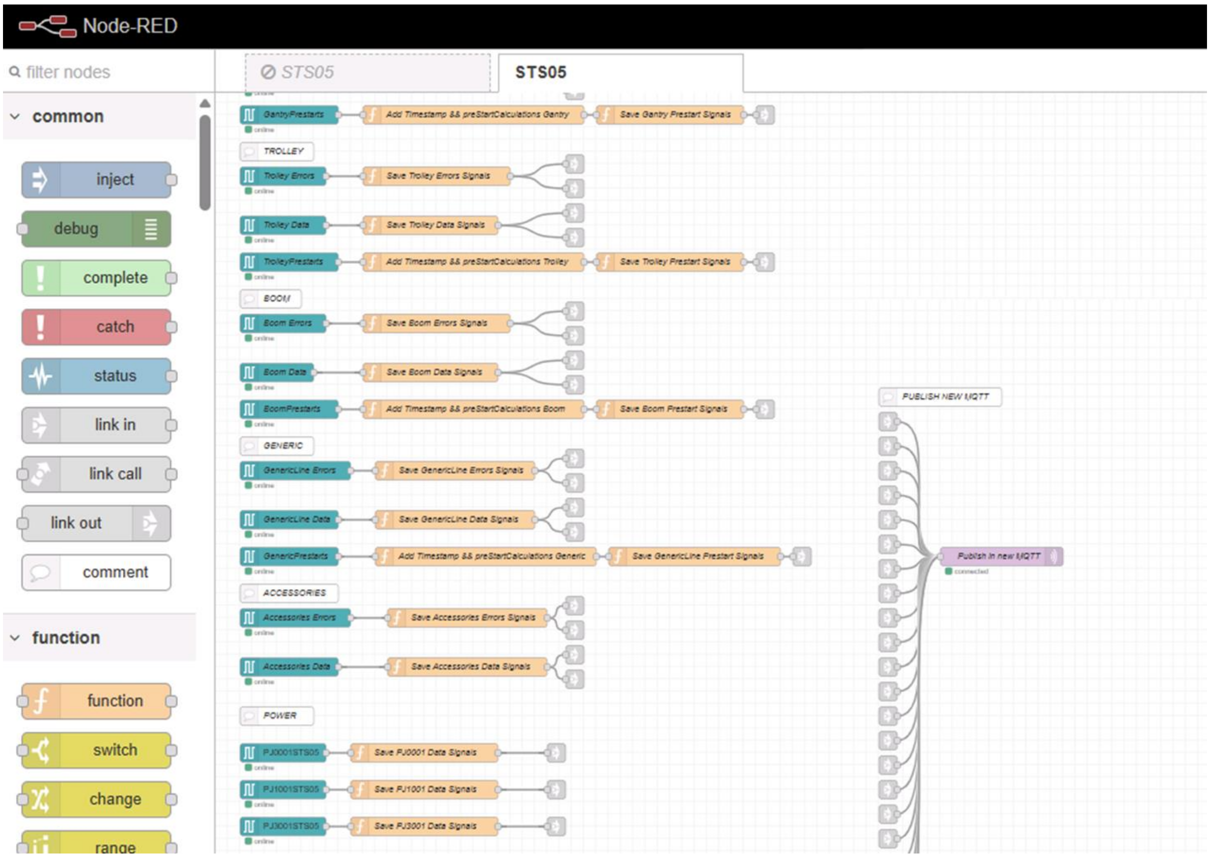


Figure 75: STS-05 data acquisition flows.

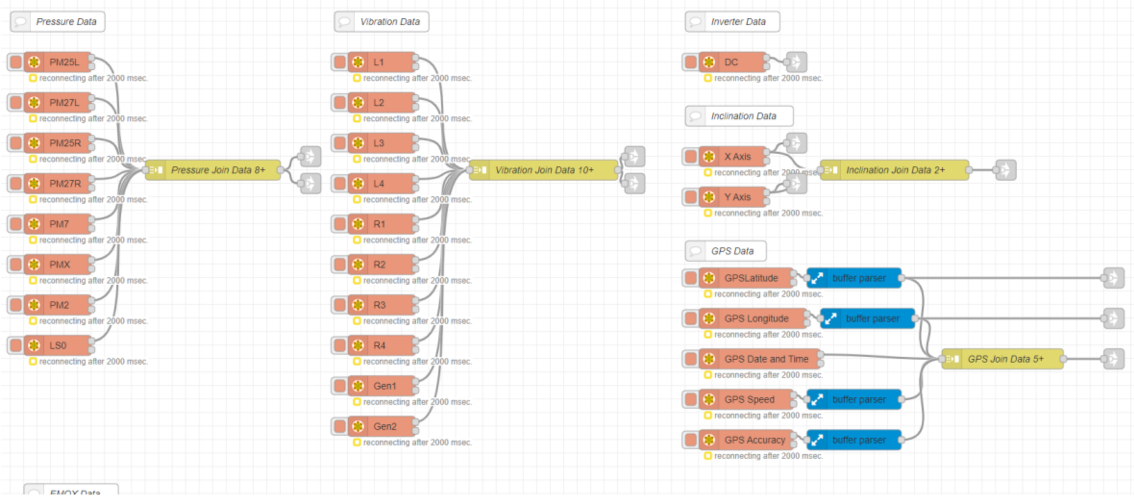
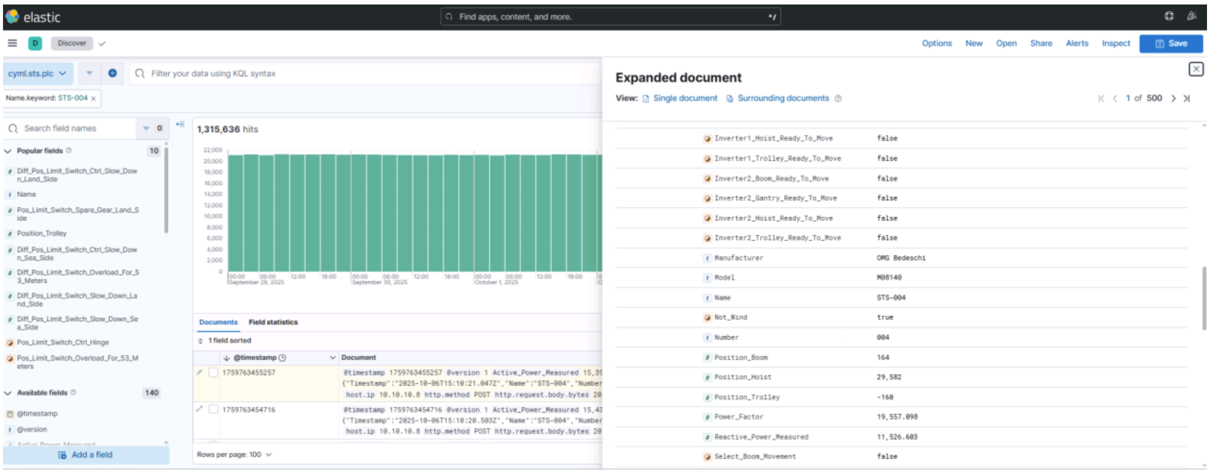


Figure 76: Straddle Carrier SCH-167 sensors data acquisition flows.

Pilot 4 – Business Process 1 – Activity - 22 (P4-BP1-VA2): Data Storage

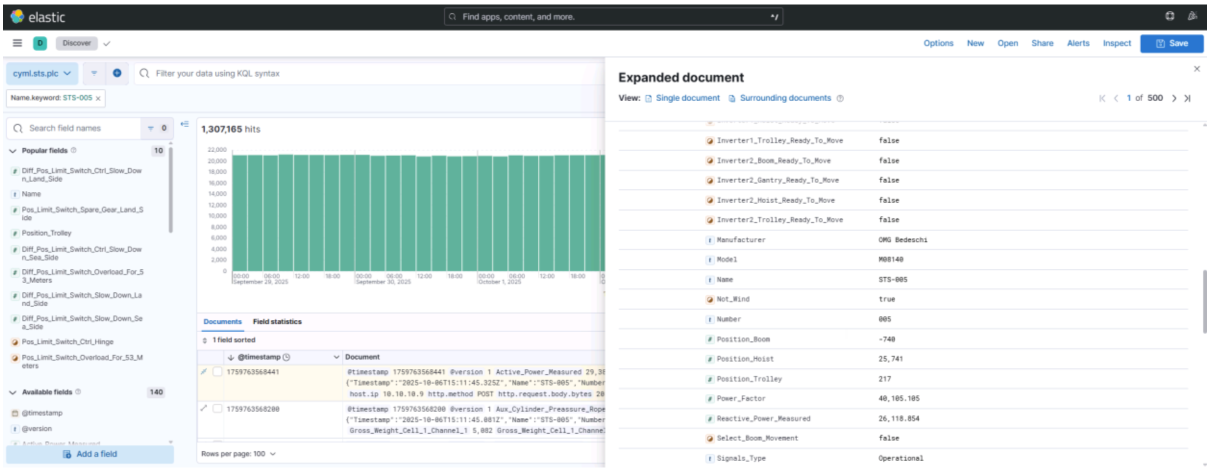
Two parallel NoSQL databases have been used for data storage: Elasticsearch in EUROGATE domain, and InfluxDB in CUT domain. The data collected has been used as training and validation datasets for the multiple AI-based PdM models developed and deployed in the pilot. Some screenshots showing some of the Elasticsearch indices associated with the data sources stored are presented below:



```
1759763455257   @timestamp 1759763455257 @version 1 Active_Power_Measured 15,39
                {"Timestamp":"2025-10-06T15:10:21.047Z","Name":"STS-004","Number
                host.ip 10.10.10.8 http.method POST http.request.body.bytes 20
1759763454716   @timestamp 1759763454716 @version 1 Active_Power_Measured 15,43
                {"Timestamp":"2025-10-06T15:10:20.503Z","Name":"STS-004","Number
                host.ip 10.10.10.8 http.method POST http.request.body.bytes 20
```

Inverter1_Hoist_Ready_To_Move	false
Inverter1_Trolley_Ready_To_Move	false
Inverter2_Boom_Ready_To_Move	false
Inverter2_Gantry_Ready_To_Move	false
Inverter2_Hoist_Ready_To_Move	false
Inverter2_Trolley_Ready_To_Move	false
Manufacturer	OMG Bedeschi
Model	M08140
Name	STS-004
Not_Wind	true
Number	004
Position_Boom	164
Position_Hoist	29,582
Position_Trolley	-160
Power_Factor	19,557.098
Reactive_Power_Measured	11,526.603
Select_Boom_Movement	false

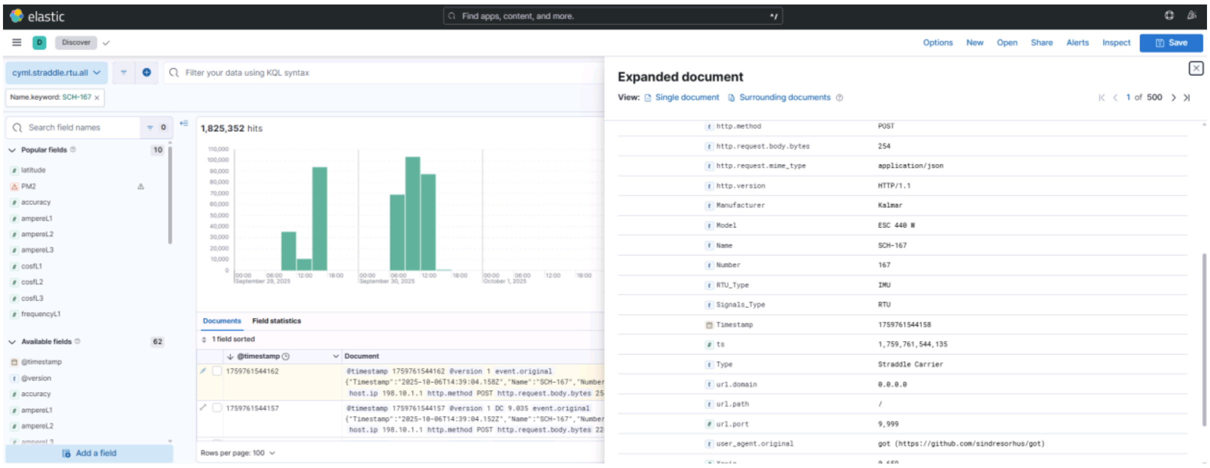
Figure 77: STS-04 PLC data stored in Elasticsearch cyml.sts.plc index



```
1759763568441   @timestamp 1759763568441 @version 1 Active_Power_Measured 29,38
                {"Timestamp":"2025-10-06T15:11:45.325Z","Name":"STS-005","Number
                host.ip 10.10.10.9 http.method POST http.request.body.bytes 20
1759763568200   @timestamp 1759763568200 @version 1 Aux_Cylinder_Preassure_Rope
                {"Timestamp":"2025-10-06T15:11:45.081Z","Name":"STS-005","Number
                Gross_Weight_Cell_1_Channel_1 5,082 Gross_Weight_Cell_1_Channel
```

Inverter1_Trolley_Ready_To_Move	false
Inverter2_Boom_Ready_To_Move	false
Inverter2_Gantry_Ready_To_Move	false
Inverter2_Hoist_Ready_To_Move	false
Inverter2_Trolley_Ready_To_Move	false
Manufacturer	OMG Bedeschi
Model	M08140
Name	STS-005
Not_Wind	true
Number	005
Position_Boom	-740
Position_Hoist	25,741
Position_Trolley	217
Power_Factor	40,105.105
Reactive_Power_Measured	26,118.854
Select_Boom_Movement	false
Signals_Type	Operational

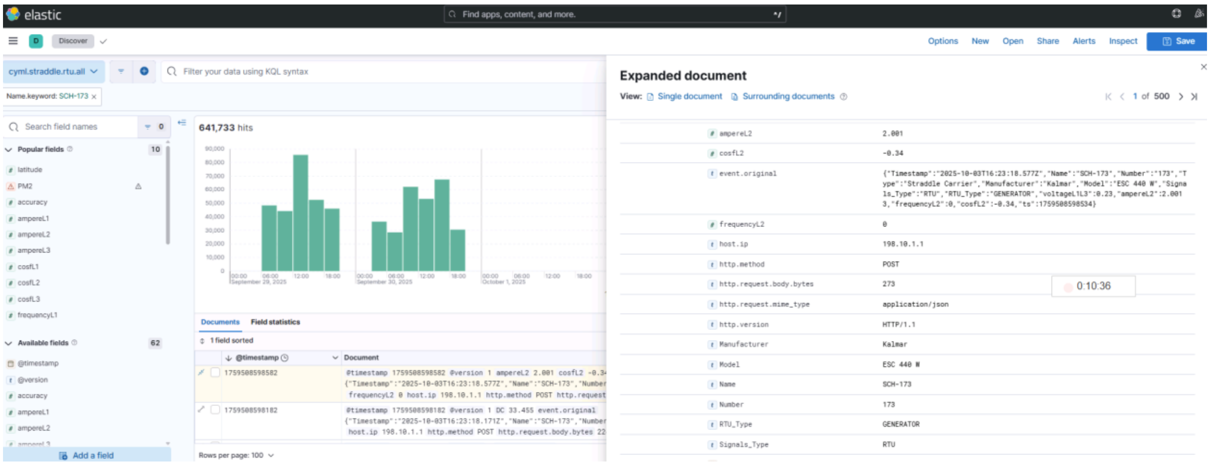
Figure 78: STS-05 PLC data stored in Elasticsearch cyml.sts.plc index



```
1759761544162   @timestamp 1759761544162 @version 1 event.original
                {"Timestamp":"2025-10-06T14:39:04.158Z","Name":"SCH-167","Number
                host.ip 198.10.1.1 http.method POST http.request.body.bytes 25
1759761544157   @timestamp 1759761544157 @version 1 DC 9.035 event.original
                {"Timestamp":"2025-10-06T14:39:04.152Z","Name":"SCH-167","Number
                host.ip 198.10.1.1 http.method POST http.request.body.bytes 22
```

http.method	POST
http.request.body.bytes	254
http.request.mime_type	application/json
http.version	HTTP/1.1
Manufacturer	Kalmar
Model	ESC 440 W
Name	SCH-167
Number	167
RTU_Type	IMU
Signals_Type	RTU
Timestamp	1759761544158
ts	1,759,761,544,135
Type	Straddle Carrier
url.domain	0.0.0.0
url.path	/
url.port	9,999
user_agent.original	got (https://github.com/sindresorhus/got)

Figure 79: Straddle Carrier SCH-167 sensors data stored in Elasticsearch cyml.straddle.rtu.all index



ampereL2	2.001
cosfL2	-0.34
event.original	{"Timestamp":"2025-10-03T16:23:18.577Z","Name":"SCH-173","Number":"173","Type":"Straddle Carrier","Manufacturer":"Kalmar","Model":"ESC 440 W","Signals_Type":"RTU","RTU_Type":"GENERATOR","voltageL1L3":0.23,"ampereL2":2.0013,"frequencyL2":0,"cosfL2":-0.34,"ts":1759508598534}
frequencyL2	0
host.ip	198.10.1.1
http.method	POST
http.request.body.bytes	273
http.request.mime_type	application/json
http.version	HTTP/1.1
Manufacturer	Kalmar
Model	ESC 440 W
Name	SCH-173
Number	173
RTU_Type	GENERATOR
Signals_Type	RTU

Figure 80: Straddle Carrier SCH-173 sensors data stored in Elasticsearch cyml.straddle.rtu.all index.

Pilot 4 – Business Process 1 – Activity - 23 (P4-BP1-VA3): STS and Straddle Carriers AI model inference verification

The different AI-based models have been verified on STS and Straddle Carriers real time maintenance:

➢ *STS trolley wire rope:*

The problem posed significant data-related and methodological challenges. The only continuous measurement available was a signal derived from the *pos limit switch*, which was designed as a safety monitoring mechanism rather than as a calibrated instrument to measure physical elongation. This signal suffered from several shortcomings:

- Due to the data acquisition strategy, with a millisecond-level sampling frequency to determine the switch position, the signal behaved like a step function. The persistence of redundant consecutive values suggested that retaining the entire signal would overweight inactive periods, potentially biasing any downstream analysis, below is an example of this trend on a data sample of approximately 4 months:

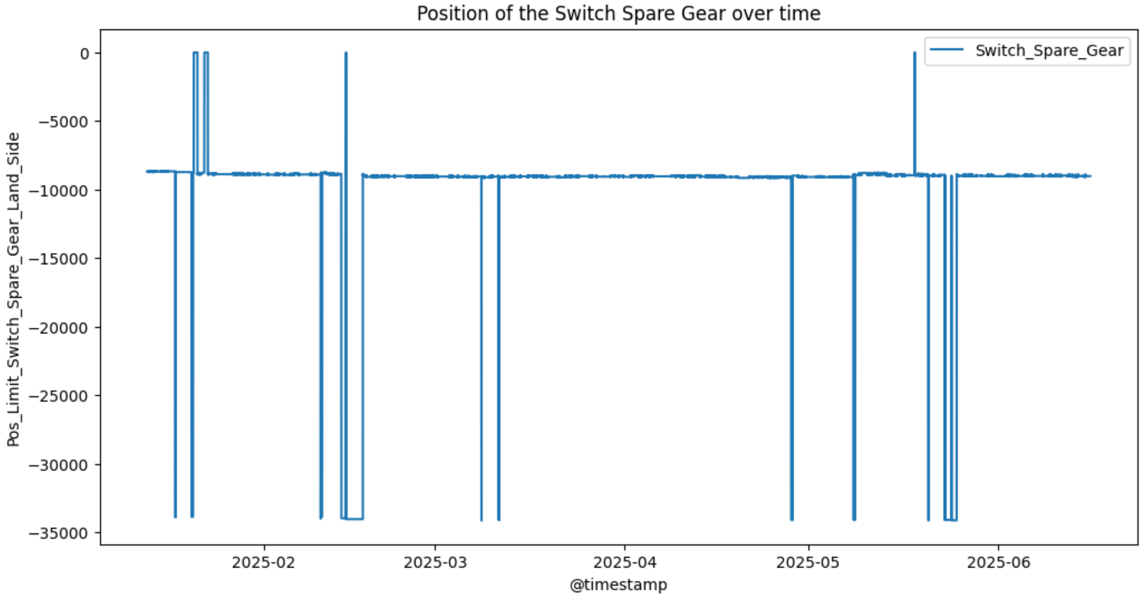


Figure 81: Position of the 'Pos_Limit_Switch_Spare_Gear_Land_Side' along 4 months. There were high peaks that were discarded for the model training.

- In the analyzed data, with approximately 8 months of historical records, inconsistencies were found regarding the crane's operational continuity. Once the target variable was preliminarily processed to mitigate the step-like behavior of the raw signal (caused by the millisecond-level sampling frequency and the persistence of redundant consecutive values, see previous point) interruptions in activity were identified. These interruptions pose an analytical challenge when approaching the problem as a time series, especially without prior knowledge of the periods between cable cuts to validate any approach. The following figure shows an evaluation of the raw variable ('Pos_Limit_Switch_Spare_Gear_Land_Side'), highlighting the moments where service interruptions longer than 50 hours begin

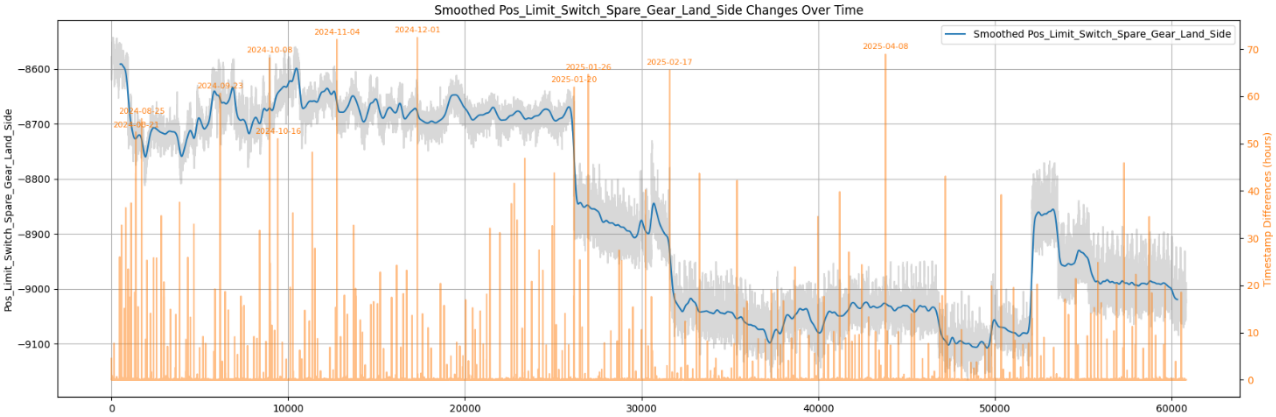


Figure 82: 'Pos_Limit_Switch_Spare_Gear_Land_Side' variable evolution.

It was contaminated by external mechanisms with unknown effects over to rope degradation, such as adjustments in the rope winding and re-tensioning system. The figure below shows the joint distribution of *Trolley_Piston_Cylinder_Pressure* and the target variable *Pos_Limit_Switch_Spare_Gear_Land_Side*. The wide spread of pressure values across the two dominant switch positions illustrates the inconsistency of the pressure signal and its inability to explain the large oscillations observed in the raw target variableComplementary analyses of linear (Pearson) and non-linear (Spearman, Mutual Information) correlations using different elongation proxies — including *Pos_Limit_Switch_Spare_Gear_Land_Side* — confirmed that no meaningful association could be established between piston pressure fluctuations and actual changes in elongation, making any segmentation based on this variable unfeasible.

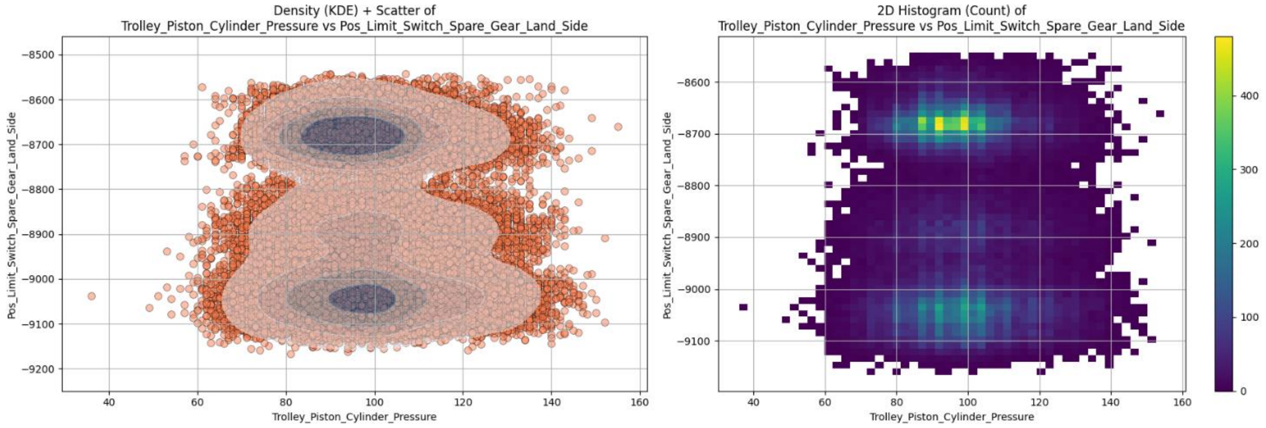


Figure 83: Joint distribution of Trolley_Piston_Cylinder_Pressure and the target variable Pos_Limit_Switch_Spare_Gear_Land_Side.

It lacked recalibration after each rope cut, leading to shifting baselines and loss of absolute reference. Moreover, in the previous figure, the nominal pressure range (between 90 and 110) operates across several ranges of the target variable, with differences exceeding the 500 mm safety threshold. This suggests that no system recalibration was performed when tension adjustments were applied

- It mixed states of load and no-load, blurring the distinction between elastic and plastic components of deformation.

These limitations meant that traditional engineering models—such as Hooke's law for elasticity or Basquin's law for fatigue—could not be applied directly, since the necessary physical parameters and clean calibration points were absent. Instead, the project required a pragmatic data-driven approach, focused on transforming and refining the available signal into a usable proxy of elongation.

The goal was therefore:

1. **To enhance the quality of the raw signal** through systematic preprocessing and filtering, isolating a consistent representation of rope deformation despite noise and interruptions.
2. **To design a predictive modeling framework** that was robust, interpretable, and deployable in production environments, while remaining computationally lightweight enough for corporate and potentially on-edge deployments.

Once the raw data issues were identified, the goal was to capture the progressive and irreversible stretching of the rope. Several proxy variables were tested, based on the following physical principles of rope elongation, plastic-elastic deformation relations and noisy data:

- **Cumulative displacement**: Direct integration of position changes over time, aiming to reflect total elongation.
- **Plastic deformation proxy**: Long-term monotonic trends in the signal, consistent with irreversible stretching.
- **Hybrid indices**: Combinations of differences and rolling statistics designed to balance sensitivity to load cycles with robustness against baseline shifts.

Each of these proxies was intended to bring the noisy signal closer to a mechanical interpretation: cumulative elongation, reversible elastic response, or progressive plastic deformation. Although no one perfectly described the physical phenomena, this exploration clarified which formulations were most stable and interpretable under the constraints of the available data. To evaluate the behavior of these candidate proxies, small-scale analyses were performed using both the raw variables and their smoothed versions. The ***absolute differences between consecutive records*** were finally selected as the principal proxy for elongation due to its robustness to baseline shifts, independence from directionality, preservation of local dynamics, predictability and consistency.

Finally, two **linear models** were adopted as the core predictive framework:

1. **Simple linear regression**: to capture the overall rate of change in the selected proxy of elongation.
2. **Quantile regression ($\tau = 0.75$)**: to provide a more robust estimate of the upper trend, less sensitive to local fluctuations and outliers. This model was intentionally designed as part of a **contingency system**, deliberately overestimating elongation compared to the simple regression, thereby serving as a preventive safeguard.

- ➢ *Straddle Carrier hydraulic system failures*:

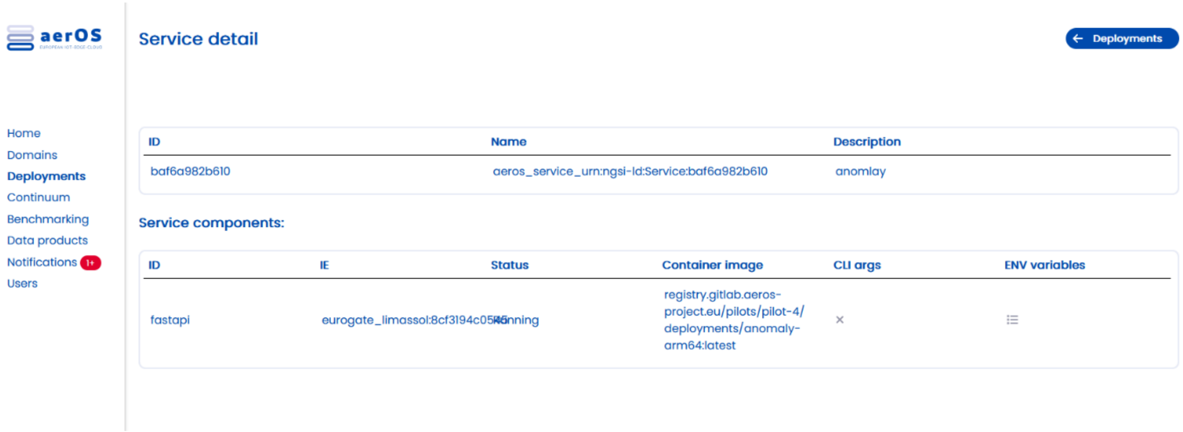
Faced with the lack of high-quality labels, we employed an unsupervised anomaly detection method based on more traditional statistical tests to identify potential anomalies in the dataset, which consists of time series for the 4 hydraulic signals measured every 100ms from June to October 2024, totalling 4 GB. This approach generated a list of potentially anomalous timestamps; each assigned a likelihood score indicating their rarity as anomalies. Statistical analysis validated the significance of the detected outliers. To further ensure the quality of the developed method, three main metrics were considered (precision, recall, and accuracy). A domain expert manually reviewed a subset of flagged timestamps over a continuous 5-month period (June-October 2024).

As it can be seen in the Confusion Matrix, on the left part of the Table 21, during that time frame, timestamps were considered, of which, in turn, 45 of them were flagged as potentially anomalous by the model. In parallel, the engineering team informed about 6 actual anomalous events. By comparison of these two reported labels, it was determined that 83% of the actual anomalies were correctly flagged as potential anomalies by the model. In the right part of Table 21, the binary metrics lead to a model with low recall due to a large quantity of false positives but a high precision. In that sense, for this particular case study, the most relevant metric is precision since the cost of a false negative is significantly higher than that of a false positive. Other relevant metrics are accuracy with 91.8% and specificity (True Negative Rate) with 99.78%.

These results demonstrate that our unsupervised technique effectively identifies meaningful anomalies even in the absence of labelled training data.

Table 21: Confusion Matrix and classification metrics for the hydraulic anomaly detection model.

	Confusion Matrix			Binary Classification Metrics	
	Predicted			**Metric**	**Value**
Actual	**Positive**	**Negative**	**Total**	Accuracy	0.918
Positive	5	1	6	Precision	0.833
Negative	40	453	493	Recall (sensitivity)	0.111
Total	45	454	499	Specificity	0.998



Service detail

ID	Name	Description
baf6a982b610	aeros_service_urn:ngsi-ld:Service:baf6a982b610	anomlay

Service components:

ID	IE	Status	Container image	CLI args	ENV variables
fastapi	eurogate_limassol:8cf3194c05	Running	registry.gitlab.aeros-project.eu/pilots/pilot-4/deployments/anomaly-arm64:latest	×	☰

Figure 84: Hydraulic anomaly detection AI model deployed as an aerOS service through the management portal of the pilot.

- *Straddle Carrier engine, brake and inverter overheating:*

Table 22 shows the performance results of the five ML models on the test dataset using four common classification performance metrics, namely, accuracy, precision, recall, and F1-score, while Figure 85 shows the corresponding confusion matrices. The ANN model outperforms others across all metrics with a 98.7% accuracy and 98.0% F1-score, likely due to its superior capability in capturing nonlinear relationships between input sensor variables and the target fault condition, as well as due to its multilayered architecture and ability to learn complex feature representations. The ensemble-based models, such as Random Forest and XGBoost, demonstrated identical performance with an accuracy of 95.4% and an F1-score of 97.0%. Interestingly, the ANN model did not classify any of the normal data points as faulty, with only a small number of false positives (2%), while RF and XGBoost achieved zero false positives (see Figure 85). These results suggest that these models are highly effective at capturing fault patterns and minimizing misclassification. Further to ensemble-based models, GNB showed a lower performance with an accuracy of 94.8% and F1-score of 92.5%, likely due to its strong assumption of independence between features. Finally, even though DT also did not produce any false positives, DT achieved the lowest accuracy at 94.4% and an even lower F1 score of 91.5% because of its increased false negative rate. Overall, these results indicate that the ANN model effectively captures temporal dependencies within the data, making it highly suitable for predictive maintenance tasks.

Table 22: Performance analysis in terms of accuracy, precision, recall, and F1-score.

Model	Accuracy	Precision	Recall	Score
ANN	0.9873	0.975	0.990	0.980
DT	0.9441	0.965	0.885	0.915
RF	0.9532	0.970	0.900	0.930
XGBoost	0.9532	0.970	0.900	0.930
GNB	0.9478	0.970	0.890	0.925

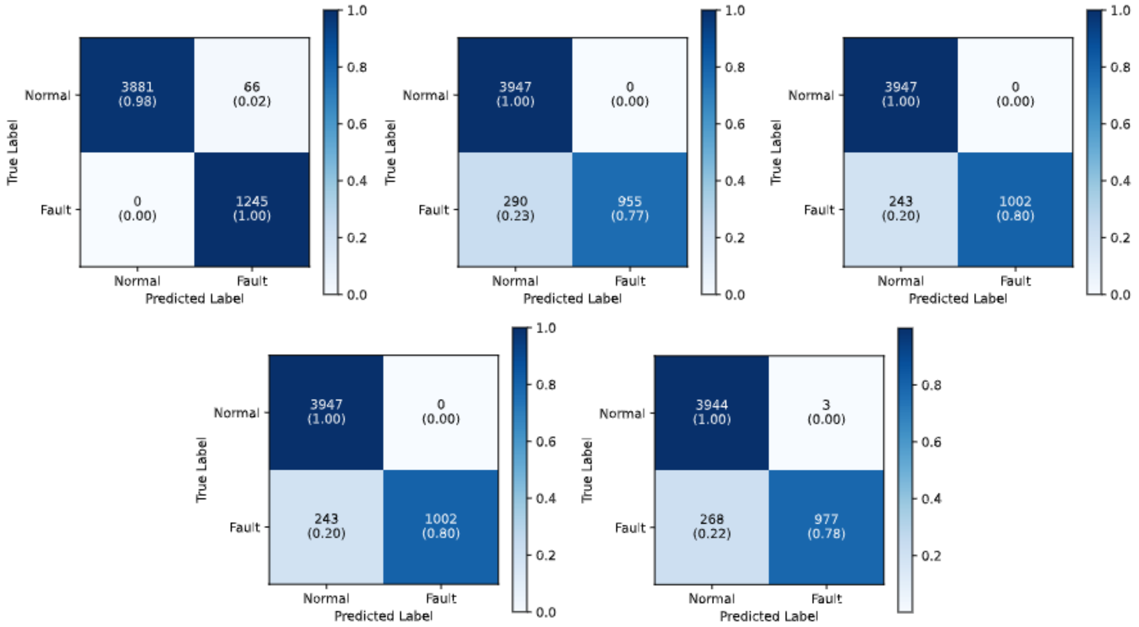


Figure 85: Confusion matrices for the five ML models: ANN, DT, RF, XGBoost, and GNB

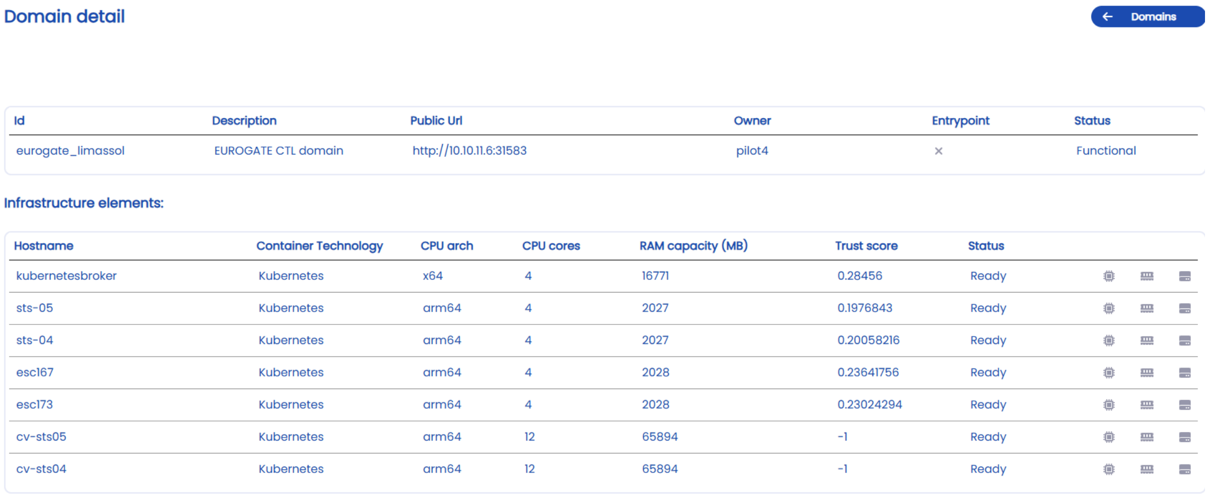
Pilot 4 – Business Process 1 – Activity - 24 (P4-BP1-VA4): aerOS entrypoint domain – EUROGATE domain communication

This validation activity aims at verifying the proper communication between two of the domains of the pilot (from the entrypoint to the EUROGATE domain). To do so, after confirming its integration in deliverable D5.4, for the final verification it was proposed to carry out with the deployment of an aerOS service from the management portal in the entrypoint domain to any of the registered IEs of the EUROGATE domain. In particular, it was decided to test communication by deploying one of the PdM models developed in the scenario. The following images present some screenshots of the management portal pages, including the 3 domains of the pilot, the specific information of the EUROGATE domain, and the status of the anomaly detection PdM model aerOS service on the STS-04 IE.

Domains list

Id	Description	Public Url	Owner	Entrypoint	Status
pilot4_domain1	Pilot4 domain1	https://aeros-pilot4.prodevelopataws.com	pilot4	✓	Preliminary
eurogate_limassol	EUROGATE CTL domain	http://10.10.11.6:31583	pilot4	×	Functional
cutdomain	This is a new domain, which is not the entrypoint	https://cutdom.aeros.cut.ac.cy	CUT	×	Preliminary

Figure 86: Pilot 4 domains available in the management portal.



Domain detail

Id	Description	Public Url	Owner	Entrypoint	Status
eurogate_limassol	EUROGATE CTL domain	http://10.10.11.6:31583	pilot4	×	Functional

Infrastructure elements:

Hostname	Container Technology	CPU arch	CPU cores	RAM capacity (MB)	Trust score	Status
kubernetesbroker	Kubernetes	x64	4	16771	0.28456	Ready
sts-05	Kubernetes	arm64	4	2027	0.1976843	Ready
sts-04	Kubernetes	arm64	4	2027	0.20058216	Ready
esc167	Kubernetes	arm64	4	2028	0.23641756	Ready
esc173	Kubernetes	arm64	4	2028	0.23024294	Ready
cv-sts05	Kubernetes	arm64	12	65894	-1	Ready
cv-sts04	Kubernetes	arm64	12	65894	-1	Ready

Figure 87: Specific information of the EUROGATE domain in the management portal.

Service detail

ID	Name	Description
963df23cf18c	aeros_service_urn:ngsi-ld:Service:963df23cf18c	anomaly-sts04

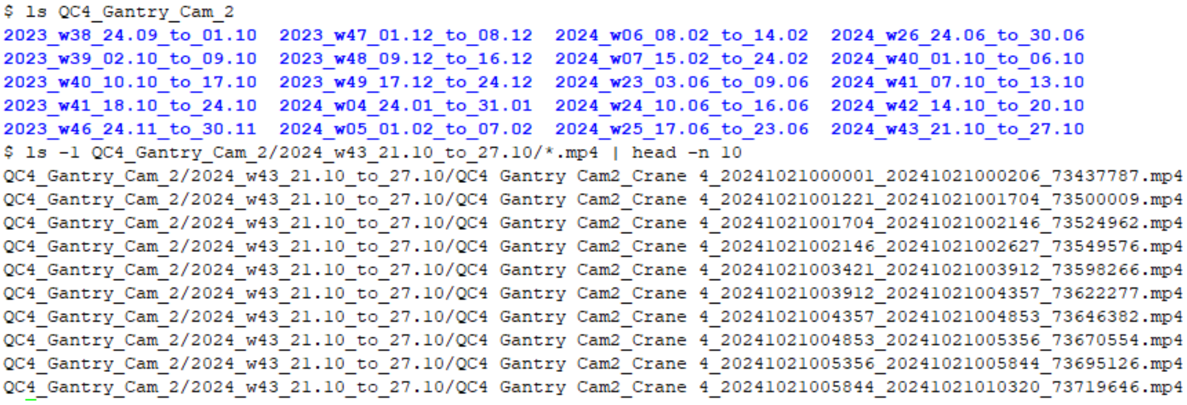
Service components:

ID	IE	Status	Container image	CLI args	ENV variables
anomaly	eurogate_limassol:8cf319c5301d	Running	registry.gitlab.aeros-project.eu/pilots/pilot-4/deployments/anomaly-arm64:latest	×	☰

Figure 88: Running status of the anomaly detection aerOS service on the STS-04 IE.

Pilot 4 – Business Process 2 – Activity - 36 (P4-BP2-VA1): Video storage

This validation activity intends to guarantee that the video streams captured by the IPTV cameras are properly recorded and stored for further used as datasets on CV models training, and validation. In that sense, videos taken throughout the year 2024 were collected, at different times of day and night, and under different weather conditions (e.g., sunny, cloudy, rainy). Then these videos were converted into frames, after which each frame was manually reviewed, and defined with a bounding box around the containers using an open-source tool called LabelImg. A total of 1927 images with containers were labeled, containing containers of different sizes (e.g., 20 ft, 40 ft), types (e.g., regular, reefer, open top), and colors (e.g., yellow, blue, red). A partial list of the collected videos is shown in the next image.

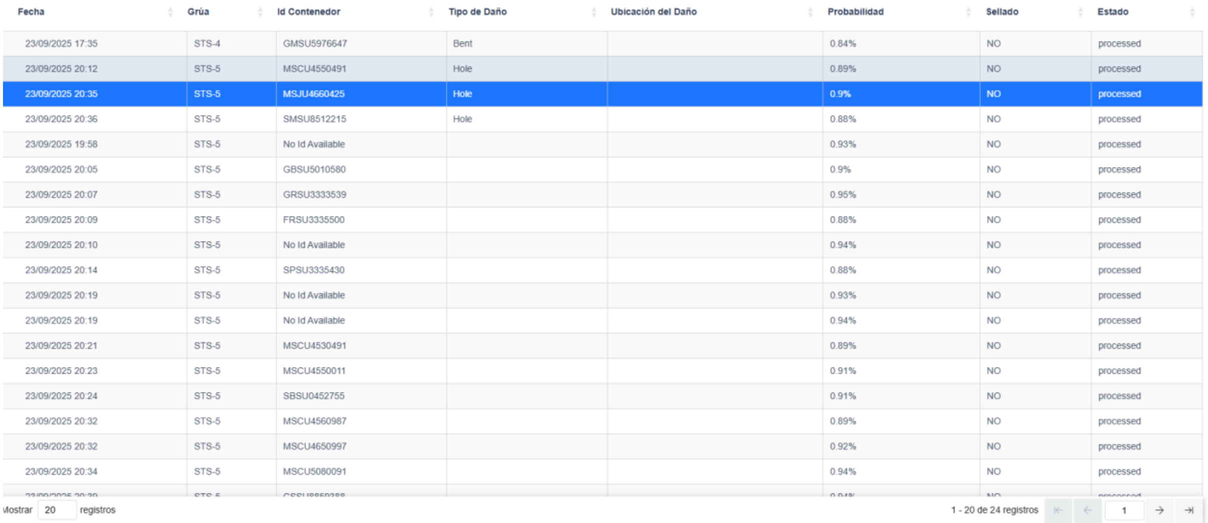


```
$ ls QC4_Gantry_Cam_2
2023_w38_24.09_to_01.10  2023_w47_01.12_to_08.12  2024_w06_08.02_to_14.02  2024_w26_24.06_to_30.06
2023_w39_02.10_to_09.10  2023_w48_09.12_to_16.12  2024_w07_15.02_to_24.02  2024_w40_01.10_to_06.10
2023_w40_10.10_to_17.10  2023_w49_17.12_to_24.12  2024_w23_03.06_to_09.06  2024_w41_07.10_to_13.10
2023_w41_18.10_to_24.10  2024_w04_24.01_to_31.01  2024_w24_10.06_to_16.06  2024_w42_14.10_to_20.10
2023_w46_24.11_to_30.11  2024_w05_01.02_to_07.02  2024_w25_17.06_to_23.06  2024_w43_21.10_to_27.10
$ ls -1 QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/*.mp4 | head -n 10
QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/QC4 Gantry Cam2_Crane 4_20241021000001_20241021000206_73437787.mp4
QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/QC4 Gantry Cam2_Crane 4_20241021001221_20241021001704_73500009.mp4
QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/QC4 Gantry Cam2_Crane 4_20241021001704_20241021002146_73524962.mp4
QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/QC4 Gantry Cam2_Crane 4_20241021002146_20241021002627_73549576.mp4
QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/QC4 Gantry Cam2_Crane 4_20241021003421_20241021003912_73598266.mp4
QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/QC4 Gantry Cam2_Crane 4_20241021003912_20241021004357_73622277.mp4
QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/QC4 Gantry Cam2_Crane 4_20241021004357_20241021004853_73646382.mp4
QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/QC4 Gantry Cam2_Crane 4_20241021004853_20241021005356_73670554.mp4
QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/QC4 Gantry Cam2_Crane 4_20241021005356_20241021005844_73695126.mp4
QC4_Gantry_Cam_2/2024_w43_21.10_to_27.10/QC4 Gantry Cam2_Crane 4_20241021005844_20241021010320_73719646.mp4
```

Figure 89: STS-04 collected videos during October 2024.

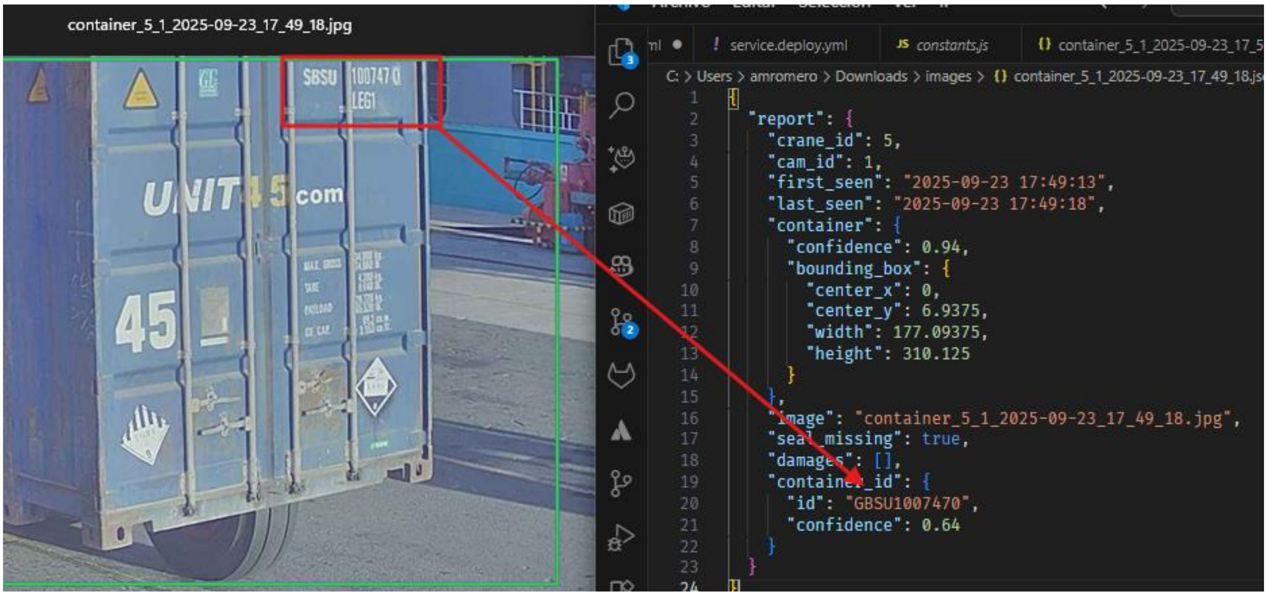
Pilot 4 – Business Process 2 – Activity - 37 (P4-BP2-VA2): Model inference verification

After the three different CV models were trained they were put into operation of the pilot. To do so, the outputs or inferences of the models were transmitted in two different topics to the MQTT broker of the pilot. On the one hand, the captured frame with the overlapped bounding box identifying the damaged container was serialized and sent as an MQTT message to the /aerOS/cv/damage MQTT topic. On the other hand, the associated metadata of the detection was sent in JSON format to the /aerOS/cv/reports MQTT topic. Next, a web application subscribed to those topics and recorded the reports and images in its own database, providing an overall view of the containers loaded and unloaded into EUROGATE premises. A screenshot of the MQTT messages as well as from the web application form is provided below.



Fecha	Grúa	Id Contenedor	Tipo de Daño	Ubicación del Daño	Probabilidad	Sellado	Estado
23/09/2025 17:35	STS-4	GMSU5976647	Bent		0.84%	NO	processed
23/09/2025 20:12	STS-5	MSCU4550491	Hole		0.89%	NO	processed
23/09/2025 20:35	STS-5	MSJU4660425	Hole		0.9%	NO	processed
23/09/2025 20:36	STS-5	SMSU8512215	Hole		0.88%	NO	processed
23/09/2025 19:58	STS-5	No Id Available			0.93%	NO	processed
23/09/2025 20:05	STS-5	GBSU5010580			0.9%	NO	processed
23/09/2025 20:07	STS-5	GRSU3333539			0.95%	NO	processed
23/09/2025 20:09	STS-5	FRSU3335500			0.88%	NO	processed
23/09/2025 20:10	STS-5	No Id Available			0.94%	NO	processed
23/09/2025 20:14	STS-5	SPSU3335430			0.88%	NO	processed
23/09/2025 20:19	STS-5	No Id Available			0.93%	NO	processed
23/09/2025 20:19	STS-5	No Id Available			0.94%	NO	processed
23/09/2025 20:21	STS-5	MSCU4530491			0.89%	NO	processed
23/09/2025 20:23	STS-5	MSCU4550011			0.91%	NO	processed
23/09/2025 20:24	STS-5	SBSU0452755			0.91%	NO	processed
23/09/2025 20:32	STS-5	MSCU4560987			0.89%	NO	processed
23/09/2025 20:32	STS-5	MSCU4650997			0.92%	NO	processed
23/09/2025 20:34	STS-5	MSCU5080091			0.94%	NO	processed

Figure 90: Web application with the list of containers inferred with the CV models of aerOS.

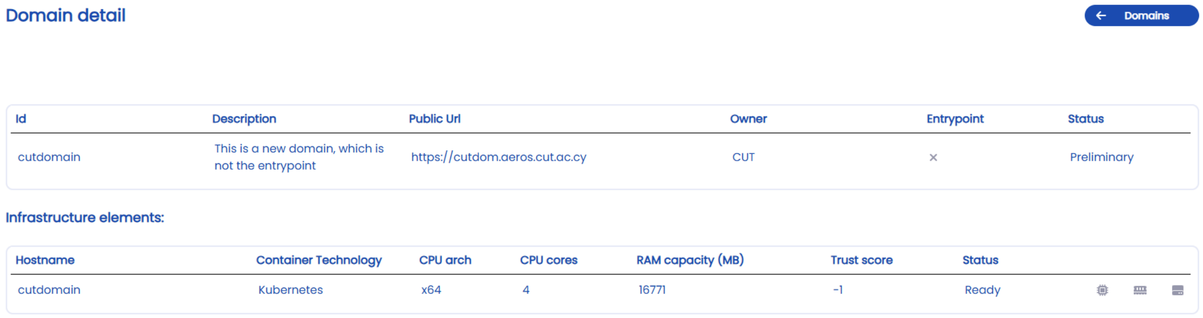


```
[
  "report": {
    "crane_id": 5,
    "cam_id": 1,
    "first_seen": "2025-09-23 17:49:13",
    "last_seen": "2025-09-23 17:49:18",
    "container": {
      "confidence": 0.94,
      "bounding_box": {
        "center_x": 0,
        "center_y": 6.9375,
        "width": 177.09375,
        "height": 310.125
      }
    },
    "image": "container_5_1_2025-09-23_17_49_18.jpg",
    "seal_missing": true,
    "damages": [],
    "container_id": {
      "id": "GBSU1007470",
      "confidence": 0.64
    }
  }
]
```

Figure 91: Image and report outputs from the CV models of aerOS.

Pilot 4 – Business Process 2 – Activity - 38 (P4-BP1-VA3): aerOS entrypoint domain – CUT domain communication

Like P4-BP1-VA4, this validation activity aims at verifying the proper communication between two of the domains of the pilot. In this case, from the entrypoint to the CUT domain. To do so, an aerOS service in the form of a nginx image test was deployed in the CUT IE. The following images present some screenshots of the management portal pages, including the specific information of the CUT domain, the deployment description of the nginx aerOS service forced to be deployed in CUT IE, and the final running status of that aerOS service.



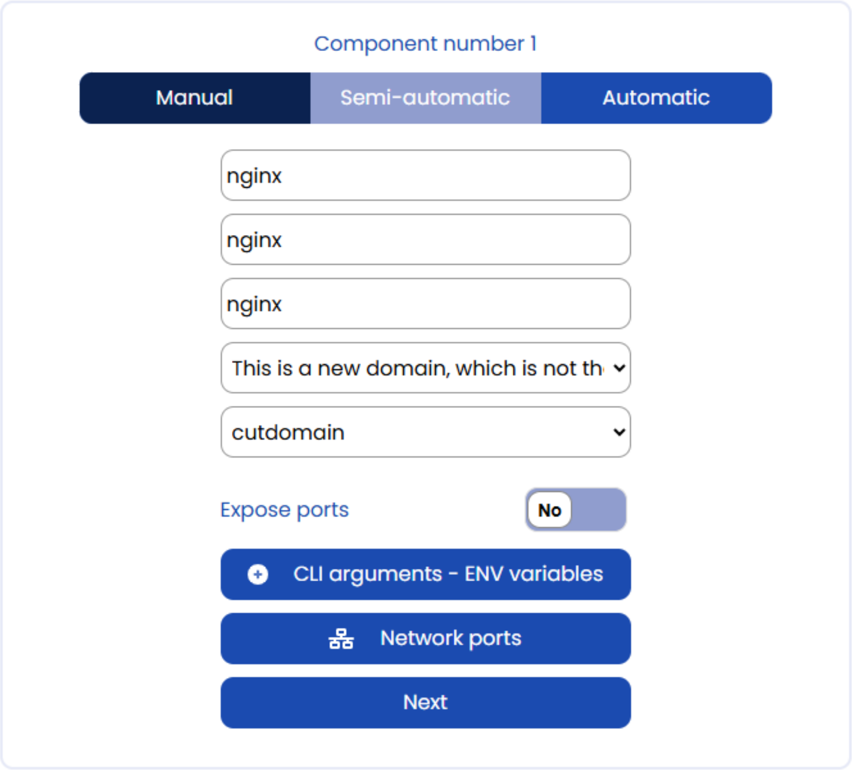
Domain detail

Id	Description	Public Url	Owner	Entrypoint	Status
cutdomain	This is a new domain, which is not the entrypoint	https://cutdom.aeros.cut.ac.cy	CUT	×	Preliminary

Infrastructure elements:

Hostname	Container Technology	CPU arch	CPU cores	RAM capacity (MB)	Trust score	Status
cutdomain	Kubernetes	x64	4	16771	-1	Ready

Figure 92: Specific information of the CUT domain in the management portal



Component number 1

nginx

nginx

nginx

This is a new domain, which is not th

cutdomain

Expose ports No

Figure 93: Deployment configuration of the nginx aerOS service into the CUTdomain IE.

ID	Name	Description
a150f26efe9f	aeros_service_urn:ngsi-ld:Service:a150f26efe9f	nginx

Service components:

ID	IE	Status	Container image	CLI args	ENV variables
nginx	cutdomain:52540072804e	Running	nginx:latest	×	×

Figure 94: Running status of the nginx aerOS service on the CUTdomain IE.

Pilot 5 - Energy Efficient, Health Safe & Sustainable Smart Buildings

Pilot 5 has closed an end-to-end user journey across two aerOS domains and a far-edge tier: IoT data feeds forecasting, an optimizer, and a desk recommender via Orion-LD; users sign in, receive seat suggestions aligned with comfort/health/energy goals, and building indicators update in real time under secure, orchestrated control.

Pilot 5		2024											2025								
Code	Name	M19	M20	M21	M22	M23	M24	M25	M26	M27	M28	M29	M30	M31	M32	M33	M34	M35	M36	M37	M38
Business Process 1 (Scenario 1) - Smart Buildings - Intelligent Occupational Safety and Health																					
Setup & Procurement Activities																					
P5-BP1-SA1	Site survey for the Selection of Pilot5 Building/Rooms	completed by M18																			
P5-BP1-SA2	Procurement of Servers & Equipment	completed by M18																			
P5-BP1-SA3	Identification of Appropriate Smart Building Sensors	completed by M18																			
Development Activities																					
P5-BP1-DA4	Deployment & Maintenance of Sensors	completed by M18	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨
P5-BP1-DA5	Deployment of the IoT backend/Home Assistant	completed by M18																			
P5-BP1-DA6	Installation of the aerOS Domains	completed by M18																			
P5-BP1-DA7	Transformation of the IoT backend as AerOS IE	completed by M18																			
P5-BP1-DA8	HVAC/Plugs Actuator Component							▨	▨	▨	▨	▨	▨	▨	▨						
P5-BP1-DA9	Forecast Engine - Health Index AI Component	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨				
P5-BP1-DA10	Forecast Engine – Environmental AI Component	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨				
P5-BP1-DA11	Forecast Engine - Energy Efficiency AI Component	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨				
P5-BP1-DA12	Health and Energy Optimization	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨				
P5-BP1-DA13	Recommender	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨							
P5-BP1-DA14	End-user GUI Application	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨					
Integration Activities																					
P5-BP1-IA15	Integration of AerOS Basic (MVP) IE Components	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	
P5-BP1-IA16	Integration of aerOS non Basic IE Components								▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	

P5-BP1-IA17	Integration of Data fabric with the IoT backend										▨	▨	▨	▨	▨	▨	▨	▨			
P5-BP1-IA18	Integration of Data Fabric with Forecast Health Index System										▨	▨	▨	▨	▨	▨	▨	▨			
P5-BP1-IA19	Integration of Data Fabric with Forecast Environmental AI System										▨	▨	▨	▨	▨	▨	▨	▨			
P5-BP1-IA20	Integration of Data Fabric with Forecast Energy Efficiency System										▨	▨	▨	▨	▨	▨	▨	▨			
P5-BP1-IA21	Integration of Data Fabric with Optimization System										▨	▨	▨	▨	▨	▨	▨	▨			
P5-BP1-IA22	Integration of Data Fabric with Recommender System										▨	▨	▨	▨	▨	▨	▨	▨			
P5-BP1-IA23	Integration of Data Fabric with End-user GUI System										▨	▨	▨	▨	▨	▨	▨	▨			
P5-BP1-IA24	E2E Integration of all Application Components (IoT, Forecast Engine, Recommender, GUI)																▨	▨	▨		
Validation Activities																					
P5-BP1-VA25	End-to-End Demonstrator (Seating Recommendation)																▨	▨	▨	▨	▨
P5-BP1-VA26	Pilot Services Created, Managed and Operated by AerOS Orchestrator	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨			
P5-BP1-VA27	Energy Use Reduction							▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨		
P5-BP1-VA28	Edge Processing Performance Gains	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨										
P5-BP1-VA29	Service Availability within the AerOS IE	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨										
P5-BP1-VA30	Service Creation / Scalability							▨	▨	▨	▨										
P5-BP1-VA31	Improvement of Air Quality							▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨	▨		
Business Process 2 (Scenario 2) - Cybersecurity and data privacy in building automation																					
Setup & Procurement Activities																					
P5-BP2-SA1	Procurement of Servers & Equipment	Completed by M18 (See use case 1)																			
P5-BP2-SA2	Identification of Targeted 5G Core	Completed by M18																			
Development Activities																					
P5-BP2-DA1	Installation of aerOS Domains	Completed by M18																			
P5-BP2-DA2	Definition of roles and access permissions		▨	▨	▨	▨															
P5-BP2-DA3	Testbed deployment for 5g capabilities extension over aerOS								▨	▨	▨										

P5-BP2-DA4	OpenCAPIF deployment for secure NEF exposure from aerOS										■	■	■							
P5-BP2-DA5	Adaptation of open5gs UPF for aerOS onboarding								■	■										
P5-BP2-DA6	Adaptation of open5gs NEF for aerOS onboarding								■	■	■	■	■							
Integration Activities																					
P5-BP2-IA1	Integration of aerOS cyber security services	Completed by M18																		
P5-BP2-IA2	Integration of roles and permissions for aerOS APIs access					■	■	■												
P5-BP2-IA3	Integration of UPF VNF in aerOS continuum													■	■					
P5-BP2-IA4	Integration of NEF VNF in aerOS continuum													■	■					
P5-BP2-IA5	Register aerOS services to Open-CAPIF															■	■			
Validation Activities																					
P5-BP2-VA1	5G E2E deployment validation with VNFs over aerOS (UERANSIM)															■	■	■	■	■
P5-BP2-VA2	Access control based on established RBAC rules										■	■	■	■	■	■	■	■	■	■

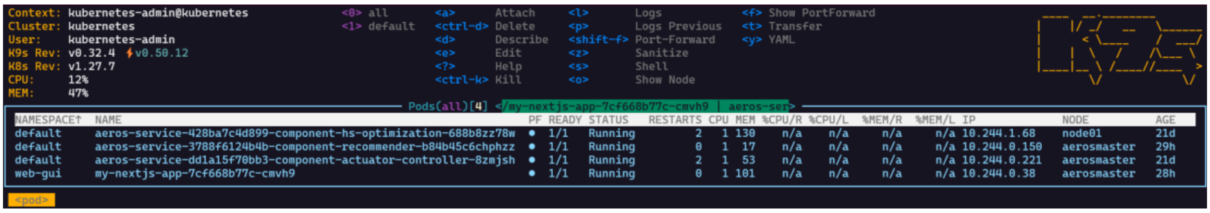
For Pilot 5, the activities carried out are related to the End-to-End Demonstrator (Seating Recommendation), the creation, management and operation of Pilot Services by aerOS Orchestrator, the reduction of the energy use, the edge processing performance gains, the service availability within an IE of aerOS, the creation and scalability of services, the improvement of Air Quality, the 5G E2E deployment validation with VNFs over aerOS (UERANSIM) and the validation of the access control based on established RBAC rules. All of these activities will be described in detail below.

Pilot 5 – Business Process 1 – Activity - 25 (P5-BP1-VA25): End-to-End Demonstrator (Seating Recommendation)

The validation of the end-to-end demonstrator is incorporating the end-user experience, from the moment that an employee approaches the enterprise building and receives the seating recommendation up until s/he leaves the premises.

The main prerequisites for the demonstrator to run are that:

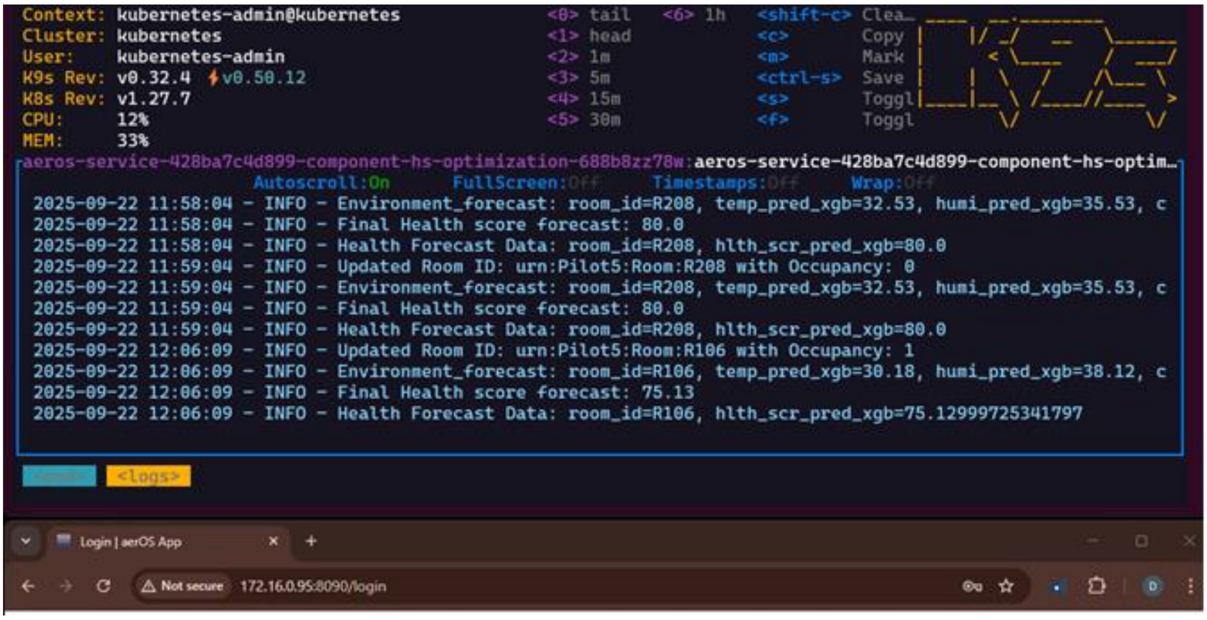
1. The pilot application components are running (as shown in Figre 95).



```
Context:  kubernetes-admin@kubernetes          <0> all       <a>        Attach    <l>         Logs           <f> Show PortForward
Cluster:  kubernetes                           <1> default   <ctrl-d>   Delete    <p>         Logs Previous  <t> Transfer
User:     kubernetes-admin                                   <d>        Describe  <shift-f>   Port-Forward   <y> YAML
K9s Rev:  v0.32.4 ⚡v0.50.12                                  <e>        Edit      <z>         Sanitize
K8s Rev:  v1.27.7                                            <?>        Help      <s>         Shell
CPU:      12%                                                <ctrl-k>   Kill      <o>         Show Node
MEM:      47%
                                          Pods(all)[4] </my-nextjs-app-7cf668b77c-cmvh9 | aeros-ser>
 NAMESPACE↑  NAME                                                          PF READY STATUS   RESTARTS CPU MEM %CPU/R %CPU/L %MEM/R %MEM/L IP            NODE         AGE
 default     aeros-service-428ba7c4d899-component-hs-optimization-688b8zz78w ●  1/1   Running         2   1 130    n/a    n/a    n/a    n/a 10.244.1.68   node01       21d
 default     aeros-service-3788f6124b4b-component-recommender-b84b45c6chphzz ●  1/1   Running         0   1  17    n/a    n/a    n/a    n/a 10.244.0.150  aerosmaster  29h
 default     aeros-service-dd1a15f70bb3-component-actuator-controller-8zmjsh ●  1/1   Running         2   1  53    n/a    n/a    n/a    n/a 10.244.0.221  aerosmaster  21d
 web-gui     my-nextjs-app-7cf668b77c-cmvh9                                  ●  1/1   Running         0   1 101    n/a    n/a    n/a    n/a 10.244.0.38   aerosmaster  28h
<pod>
```

Figure 95: Pilot5 aerOS Services running

2. The Forecasting component is working on the background and generates the health score per room:

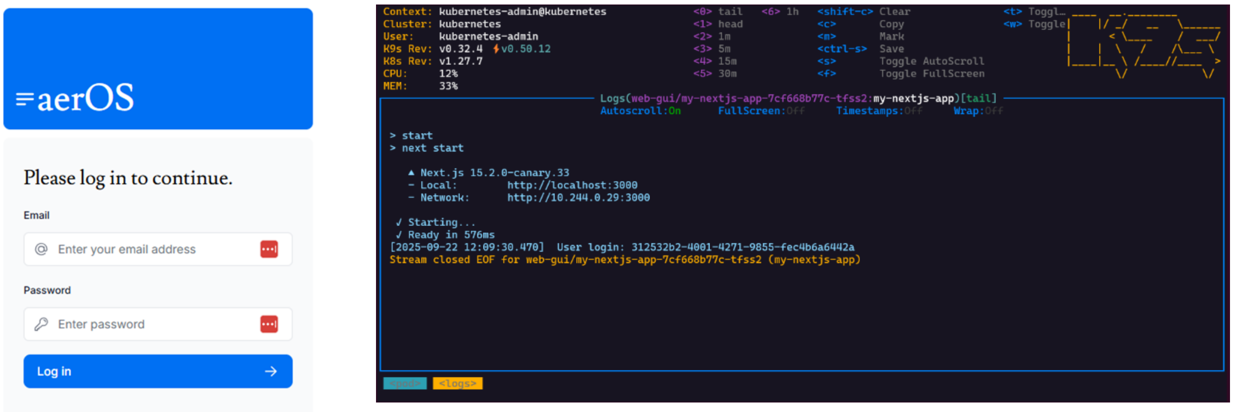


```
Context:  kubernetes-admin@kubernetes          <0> tail  <6> 1h   <shift-c> Clea…
Cluster:  kubernetes                           <1> head           <c>       Copy
User:     kubernetes-admin                     <2> 1m             <m>       Mark
K9s Rev:  v0.32.4 ⚡v0.50.12                    <3> 5m             <ctrl-s>  Save
K8s Rev:  v1.27.7                              <4> 15m            <s>       Toggl
CPU:      12%                                  <5> 30m            <f>       Toggl
MEM:      33%
aeros-service-428ba7c4d899-component-hs-optimization-688b8zz78w:aeros-service-428ba7c4d899-component-hs-optim…
          Autoscroll:On      FullScreen:Off      Timestamps:Off      Wrap:Off
 2025-09-22 11:58:04 - INFO - Environment_forecast: room_id=R208, temp_pred_xgb=32.53, humi_pred_xgb=35.53, c
 2025-09-22 11:58:04 - INFO - Final Health score forecast: 80.0
 2025-09-22 11:58:04 - INFO - Health Forecast Data: room_id=R208, hlth_scr_pred_xgb=80.0
 2025-09-22 11:59:04 - INFO - Updated Room ID: urn:Pilot5:Room:R208 with Occupancy: 0
 2025-09-22 11:59:04 - INFO - Environment_forecast: room_id=R208, temp_pred_xgb=32.53, humi_pred_xgb=35.53, c
 2025-09-22 11:59:04 - INFO - Final Health score forecast: 80.0
 2025-09-22 11:59:04 - INFO - Health Forecast Data: room_id=R208, hlth_scr_pred_xgb=80.0
 2025-09-22 12:06:09 - INFO - Updated Room ID: urn:Pilot5:Room:R106 with Occupancy: 1
 2025-09-22 12:06:09 - INFO - Environment_forecast: room_id=R106, temp_pred_xgb=30.18, humi_pred_xgb=38.12, c
 2025-09-22 12:06:09 - INFO - Final Health score forecast: 75.13
 2025-09-22 12:06:09 - INFO - Health Forecast Data: room_id=R106, hlth_scr_pred_xgb=75.12999725341797
<logs>
```

Figure 96: Health Score Forecast Calculation per Room

Thereafter, the validation scenario is demonstrated through the following sequence of activities:

1. An employee approaching signs in the WEB GUI of Pilot5 with his/her credentials:



```
Context:  kubernetes-admin@kubernetes          <0> tail   <6> 1h   <shift-c> Clear            <t> Toggl…
Cluster:  kubernetes                           <1> head             <c>       Copy             <w> Toggle
User:     kubernetes-admin                     <2> 1m               <m>       Mark
K9s Rev:  v0.32.4 ⚡v0.50.12                    <3> 5m               <ctrl-s>  Save
K8s Rev:  v1.27.7                              <4> 15m              <s>       Toggle AutoScroll
CPU:      12%                                  <5> 30m              <f>       Toggle FullScreen
MEM:      33%
                       Logs(web-gui/my-nextjs-app-7cf668b77c-tfss2:my-nextjs-app)[tail]
                       Autoscroll:On      FullScreen:Off      Timestamps:Off      Wrap:Off

> start
> next start

   ▲ Next.js 15.2.0-canary.33
   - Local:        http://localhost:3000
   - Network:      http://10.244.0.29:3000

 ✓ Starting...
 ✓ Ready in 576ms
[2025-09-22 12:09:30.470]  User login: 312532b2-4001-4271-9855-fec4b6a6442a
Stream closed EOF for web-gui/my-nextjs-app-7cf668b77c-tfss2 (my-nextjs-app)
```

Figure 97: Pilot5 Login Page and Web-GUI Log for User Login

2. Following the successful login, the user can update its seating preferences:

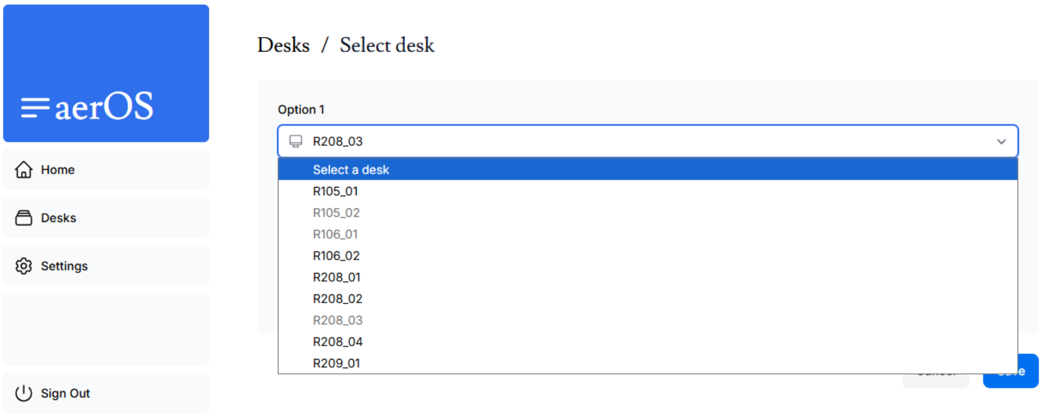
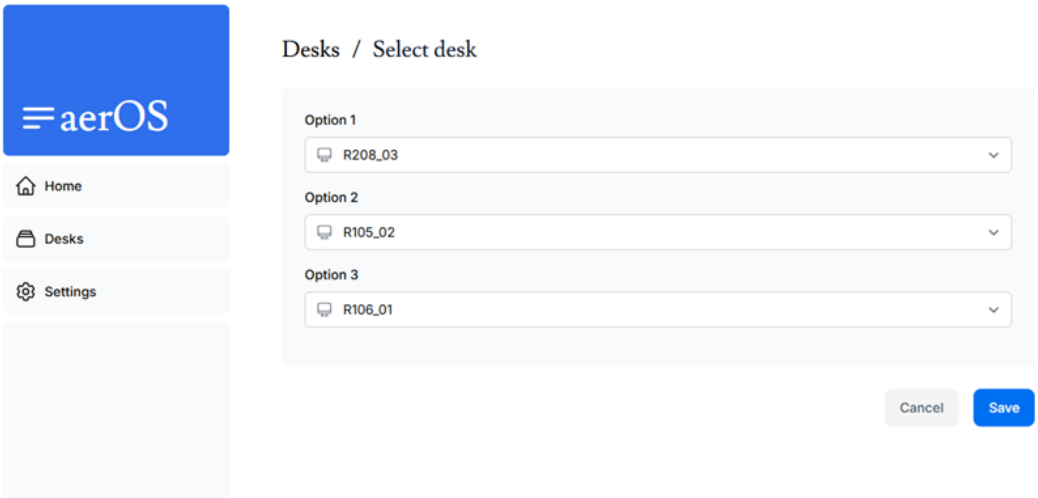
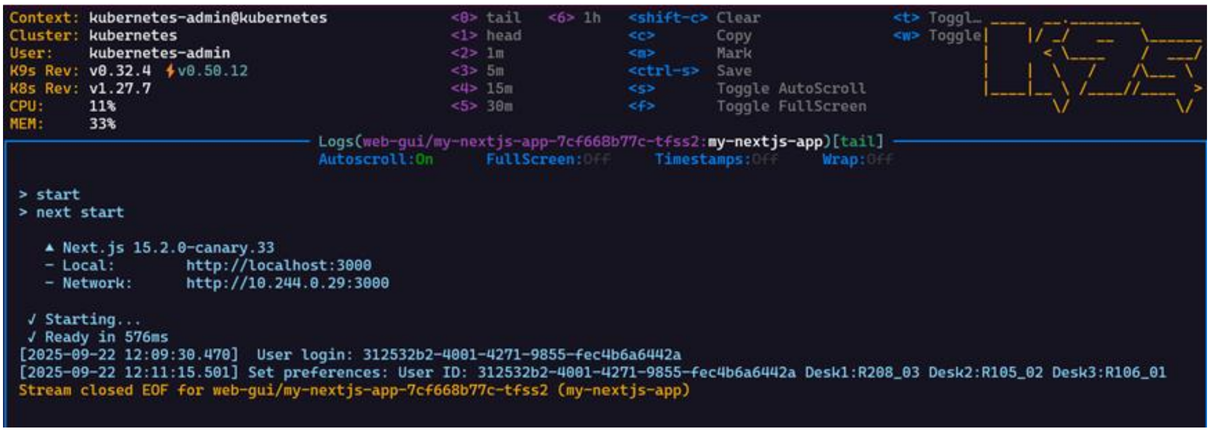


Figure 98: User Preferences Page: Select Preferred Seating (Room_Desk) for Option 1 (Most Preferred)



Option 1
R208_03
Option 2
R105_02
Option 3
R106_01

Figure 99: User Preferences Page – List of Preferred Desks Selected



```
Context:  kubernetes-admin@kubernetes              <0> tail   <6> 1h     <shift-c> Clear                <t> Toggl…
Cluster:  kubernetes                               <1> head              <c>       Copy                 <w> Toggle
User:     kubernetes-admin                         <2> 1m                <m>       Mark
K9s Rev:  v0.32.4  v0.50.12                        <3> 5m                <ctrl-s>  Save
K8s Rev:  v1.27.7                                  <4> 15m               <s>       Toggle AutoScroll
CPU:      11%                                      <5> 30m               <f>       Toggle FullScreen
MEM:      33%
                         Logs(web-gui/my-nextjs-app-7cf668b77c-tfss2:my-nextjs-app)[tail]
                         Autoscroll:On      FullScreen:Off      Timestamps:Off      Wrap:Off

 > start
 > next start

   ▲ Next.js 15.2.0-canary.33
   - Local:        http://localhost:3000
   - Network:      http://10.244.0.29:3000

 ✓ Starting...
 ✓ Ready in 576ms
[2025-09-22 12:09:30.470]  User login: 312532b2-4001-4271-9855-fec4b6a6442a
[2025-09-22 12:11:15.501] Set preferences: User ID: 312532b2-4001-4271-9855-fec4b6a6442a Desk1:R208_03 Desk2:R105_02 Desk3:R106_01
Stream closed EOF for web-gui/my-nextjs-app-7cf668b77c-tfss2 (my-nextjs-app)
```

Figure 100: Web-GUI Log for User Preferences – Desks Selected

3. User Requests for the allocated desk:

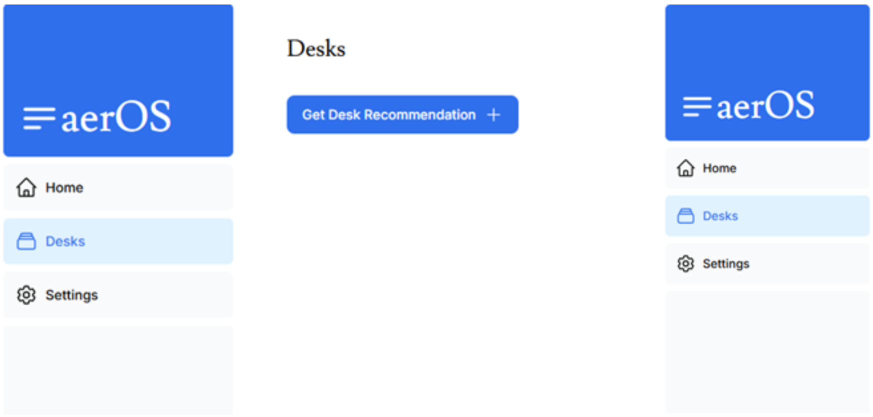
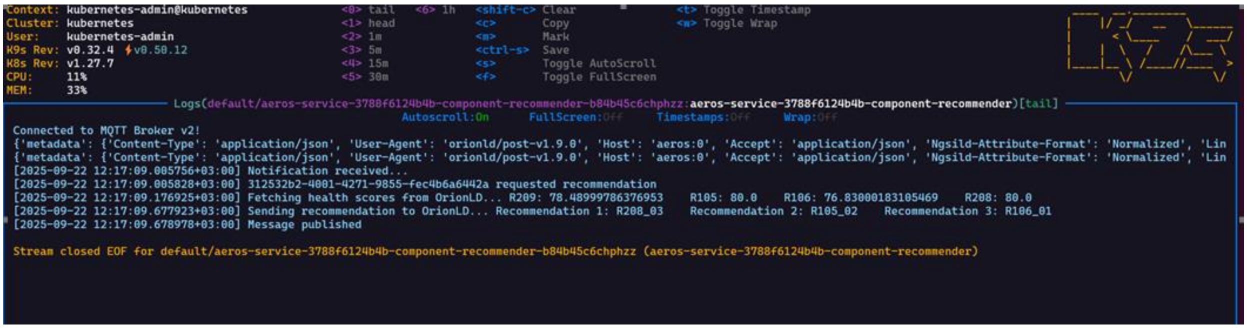


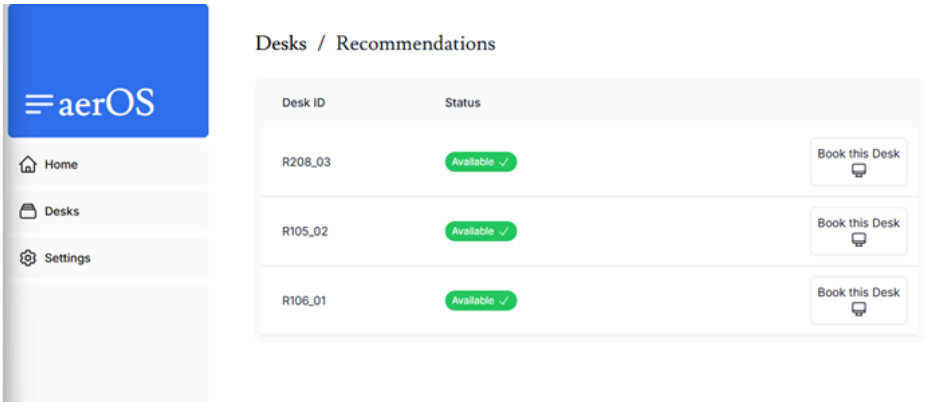
Figure 101: Seating Recommendation Page



```
Context:  kubernetes-admin@kubernetes              <0> tail   <6> 1h     <shift-c> Clear               <t> Toggle Timestamp
Cluster:  kubernetes                               <1> head              <c>       Copy                <w> Toggle Wrap
User:     kubernetes-admin                         <2> 1m                <m>       Mark
K9s Rev:  v0.32.4  v0.50.12                        <3> 5m                <ctrl-s>  Save
K8s Rev:  v1.27.7                                  <4> 15m               <s>       Toggle AutoScroll
CPU:      11%                                      <5> 30m               <f>       Toggle FullScreen
MEM:      33%
          Logs(default/aeros-service-3788f6124b4b-component-recommender-b84b45c6chphzz:aeros-service-3788f6124b4b-component-recommender)[tail]
                                  Autoscroll:On      FullScreen:Off      Timestamps:Off      Wrap:Off
Connected to MQTT Broker v2!
{'metadata': {'Content-Type': 'application/json', 'User-Agent': 'orionld/post-v1.9.0', 'Host': 'aeros:0', 'Accept': 'application/json', 'Ngsild-Attribute-Format': 'Normalized', 'Lin
{'metadata': {'Content-Type': 'application/json', 'User-Agent': 'orionld/post-v1.9.0', 'Host': 'aeros:0', 'Accept': 'application/json', 'Ngsild-Attribute-Format': 'Normalized', 'Lin
[2025-09-22 12:17:09.005756+03:00] Notification received...
[2025-09-22 12:17:09.005828+03:00] 312532b2-4001-4271-9855-fec4b6a6442a requested recommendation
[2025-09-22 12:17:09.176925+03:00] Fetching health scores from OrionLD... R209: 78.48999786376953    R105: 80.0    R106: 76.83000183105469    R208: 80.0
[2025-09-22 12:17:09.677923+03:00] Sending recommendation to OrionLD... Recommendation 1: R208_03    Recommendation 2: R105_02    Recommendation 3: R106_01
[2025-09-22 12:17:09.678978+03:00] Message published

Stream closed EOF for default/aeros-service-3788f6124b4b-component-recommender-b84b45c6chphzz (aeros-service-3788f6124b4b-component-recommender)
```

Figure 102: Recommender log for processing request and providing recommendations based on HI and preferences

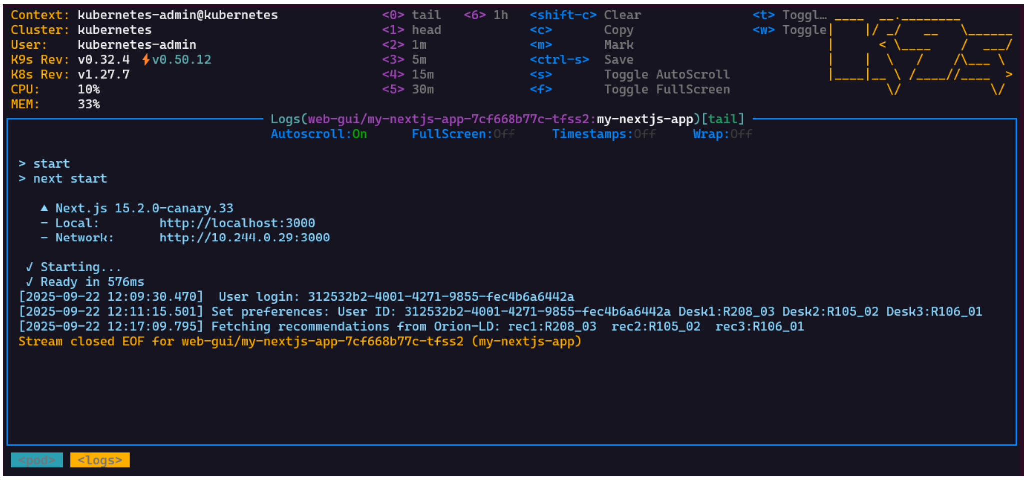
4. The recommended seating is presented in the Web GUI:



Desks / Recommendations

Desk ID	Status
R208_03	Available ✓
R105_02	Available ✓
R106_01	Available ✓

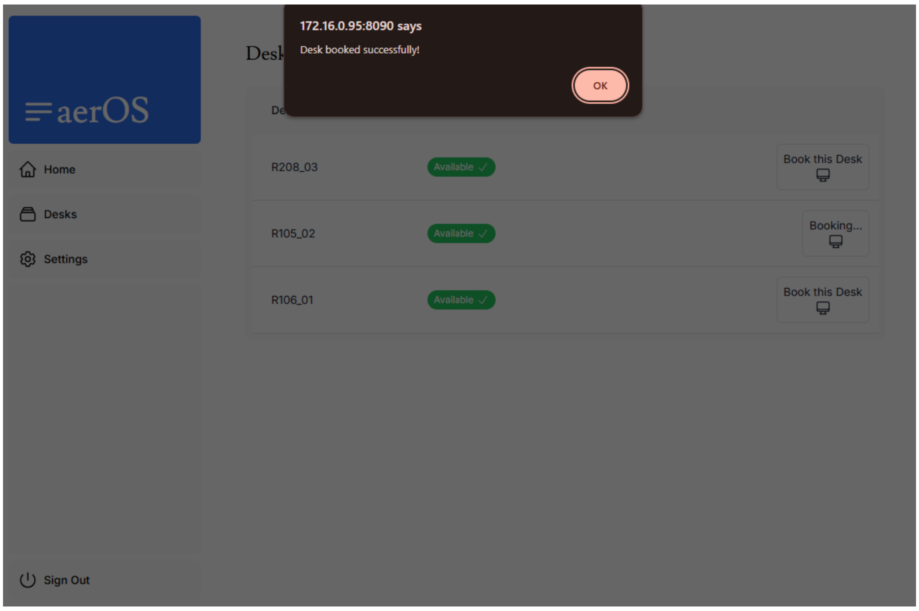
Figure 103: Seating Recommendation Results



```
Context: kubernetes-admin@kubernetes          <0> tail   <6> 1h   <shift-c> Clear            <t> Toggl…
Cluster: kubernetes                           <1> head            <c>       Copy             <w> Toggle
User:    kubernetes-admin                     <2> 1m              <m>       Mark
K9s Rev: v0.32.4 ⚡v0.50.12                    <3> 5m              <ctrl-s>  Save
K8s Rev: v1.27.7                              <4> 15m             <s>       Toggle AutoScroll
CPU:     10%                                  <5> 30m             <f>       Toggle FullScreen
MEM:     33%
                          Logs(web-gui/my-nextjs-app-7cf668b77c-tfss2:my-nextjs-app)[tail]
                          Autoscroll:On     FullScreen:Off     Timestamps:Off     Wrap:Off

> start
> next start

   ▲ Next.js 15.2.0-canary.33
   - Local:        http://localhost:3000
   - Network:      http://10.244.0.29:3000

 ✓ Starting...
 ✓ Ready in 576ms
[2025-09-22 12:09:30.470]  User login: 312532b2-4001-4271-9855-fec4b6a6442a
[2025-09-22 12:11:15.501] Set preferences: User ID: 312532b2-4001-4271-9855-fec4b6a6442a Desk1:R208_03 Desk2:R105_02 Desk3:R106_01
[2025-09-22 12:17:09.795] Fetching recommendations from Orion-LD: rec1:R208_03  rec2:R105_02  rec3:R106_01
Stream closed EOF for web-gui/my-nextjs-app-7cf668b77c-tfss2 (my-nextjs-app)
```

Figure 104: WebGUI Log for Fetching Recommendation Results from the Recommender Component

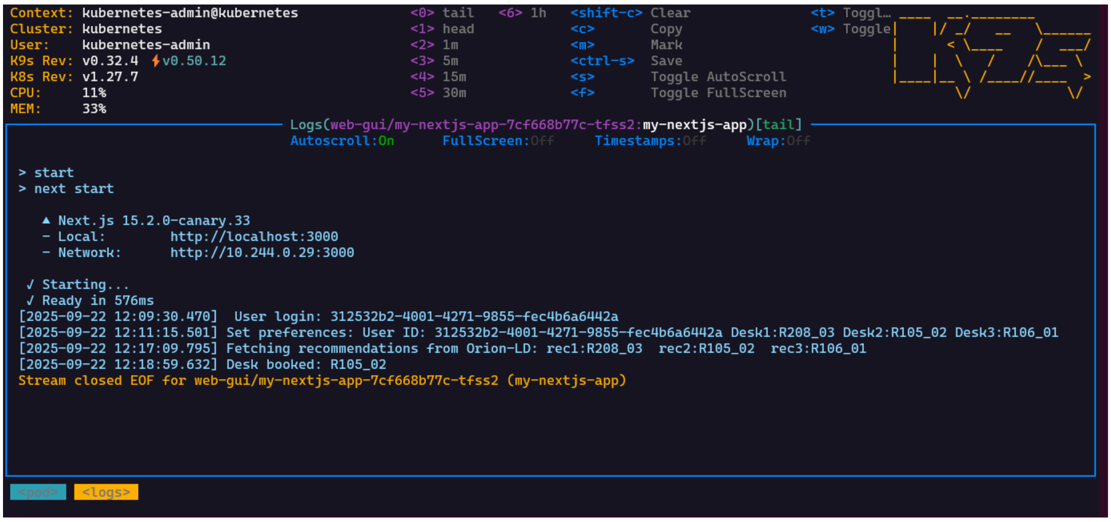
5. The user books the desk from the provided list:



172.16.0.95:8090 says

Desk booked successfully!

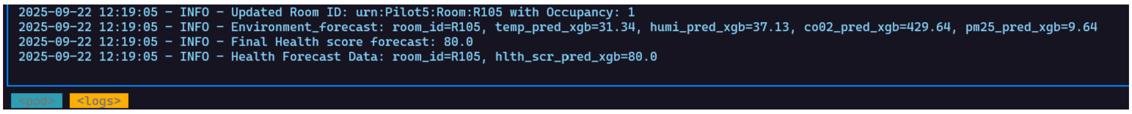
Figure 105: User Books a Desk



```
Context:  kubernetes-admin@kubernetes          <0> tail   <6> 1h   <shift-c> Clear              <t> Toggl…
Cluster:  kubernetes                           <1> head            <c>       Copy               <w> Toggle
User:     kubernetes-admin                     <2> 1m              <m>       Mark
K9s Rev:  v0.32.4 ⚡v0.50.12                   <3> 5m              <ctrl-s>  Save
K8s Rev:  v1.27.7                              <4> 15m             <s>       Toggle AutoScroll
CPU:      11%                                  <5> 30m             <f>       Toggle FullScreen
MEM:      33%
                    Logs(web-gui/my-nextjs-app-7cf668b77c-tfss2:my-nextjs-app)[tail]
                    Autoscroll:On      FullScreen:Off      Timestamps:Off      Wrap:Off

 > start
 > next start

   ▲ Next.js 15.2.0-canary.33
   - Local:        http://localhost:3000
   - Network:      http://10.244.0.29:3000

 ✓ Starting...
 ✓ Ready in 576ms
[2025-09-22 12:09:30.470]  User login: 312532b2-4001-4271-9855-fec4b6a6442a
[2025-09-22 12:11:15.501] Set preferences: User ID: 312532b2-4001-4271-9855-fec4b6a6442a Desk1:R208_03 Desk2:R105_02 Desk3:R106_01
[2025-09-22 12:17:09.795] Fetching recommendations from Orion-LD: rec1:R208_03  rec2:R105_02  rec3:R106_01
[2025-09-22 12:18:59.632] Desk booked: R105_02
Stream closed EOF for web-gui/my-nextjs-app-7cf668b77c-tfss2 (my-nextjs-app)

<pod> <logs>
```

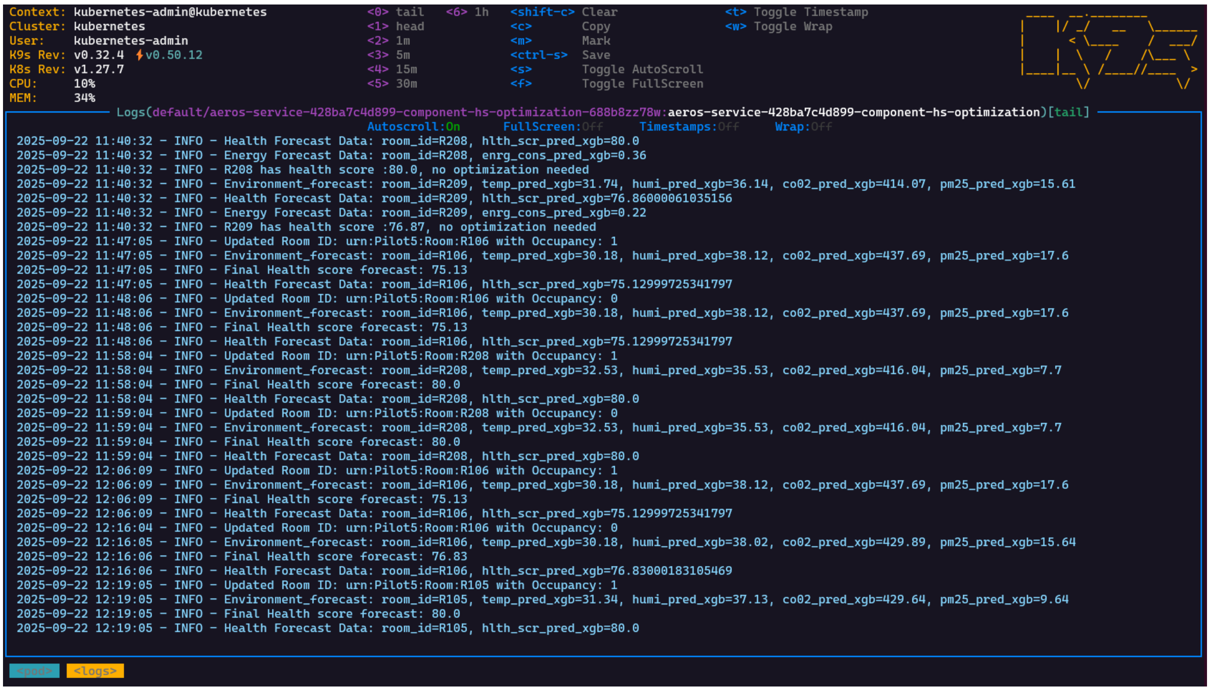
Figure 106: WebGUI Log for User's Desk Booking

6. The forecast component updates the room's occupancy (R105). Note that health score considers the updated occupancy and runs upon occupancy changes and every thirty (30) minutes.



```
2025-09-22 12:19:05 - INFO - Updated Room ID: urn:Pilot5:Room:R105 with Occupancy: 1
2025-09-22 12:19:05 - INFO - Environment_forecast: room_id=R105, temp_pred_xgb=31.34, humi_pred_xgb=37.13, co02_pred_xgb=429.64, pm25_pred_xgb=9.64
2025-09-22 12:19:05 - INFO - Final Health score forecast: 80.0
2025-09-22 12:19:05 - INFO - Health Forecast Data: room_id=R105, hlth_scr_pred_xgb=80.0

<pod> <logs>
```

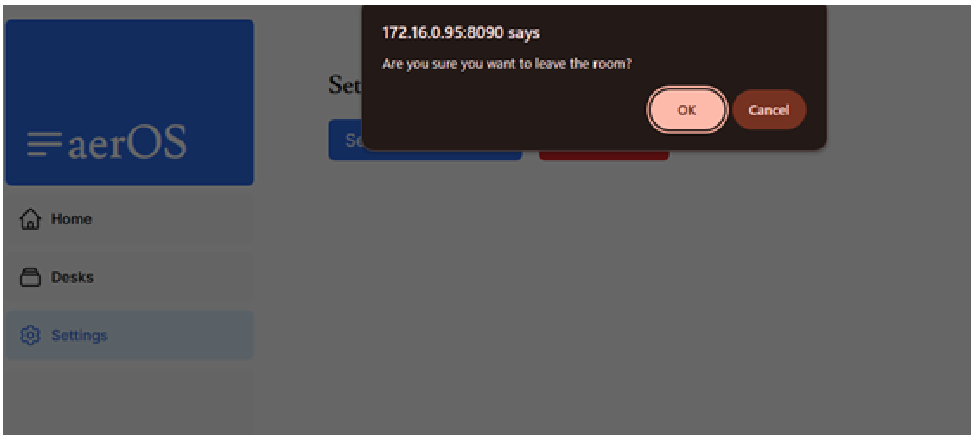
Figure 107: Forecasting Logs for Desk Occupancy Updates (Occupancy = 1)



```
Context:  kubernetes-admin@kubernetes          <0> tail   <6> 1h   <shift-c> Clear              <t> Toggle Timestamp
Cluster:  kubernetes                           <1> head            <c>       Copy               <w> Toggle Wrap
User:     kubernetes-admin                     <2> 1m              <m>       Mark
K9s Rev:  v0.32.4 ⚡v0.50.12                   <3> 5m              <ctrl-s>  Save
K8s Rev:  v1.27.7                              <4> 15m             <s>       Toggle AutoScroll
CPU:      10%                                  <5> 30m             <f>       Toggle FullScreen
MEM:      34%
 Logs(default/aeros-service-428ba7c4d899-component-hs-optimization-688b8zz78w:aeros-service-428ba7c4d899-component-hs-optimization)[tail]
                    Autoscroll:On      FullScreen:Off      Timestamps:Off      Wrap:Off
2025-09-22 11:40:32 - INFO - Health Forecast Data: room_id=R208, hlth_scr_pred_xgb=80.0
2025-09-22 11:40:32 - INFO - Energy Forecast Data: room_id=R208, enrg_cons_pred_xgb=0.36
2025-09-22 11:40:32 - INFO - R208 has health score :80.0, no optimization needed
2025-09-22 11:40:32 - INFO - Environment_forecast: room_id=R209, temp_pred_xgb=31.74, humi_pred_xgb=36.14, co02_pred_xgb=414.07, pm25_pred_xgb=15.61
2025-09-22 11:40:32 - INFO - Health Forecast Data: room_id=R209, hlth_scr_pred_xgb=76.86000061035156
2025-09-22 11:40:32 - INFO - Energy Forecast Data: room_id=R209, enrg_cons_pred_xgb=0.22
2025-09-22 11:40:32 - INFO - R209 has health score :76.87, no optimization needed
2025-09-22 11:47:05 - INFO - Updated Room ID: urn:Pilot5:Room:R106 with Occupancy: 1
2025-09-22 11:47:05 - INFO - Environment_forecast: room_id=R106, temp_pred_xgb=30.18, humi_pred_xgb=38.12, co02_pred_xgb=437.69, pm25_pred_xgb=17.6
2025-09-22 11:47:05 - INFO - Final Health score forecast: 75.13
2025-09-22 11:47:05 - INFO - Health Forecast Data: room_id=R106, hlth_scr_pred_xgb=75.12999725341797
2025-09-22 11:48:06 - INFO - Updated Room ID: urn:Pilot5:Room:R106 with Occupancy: 0
2025-09-22 11:48:06 - INFO - Environment_forecast: room_id=R106, temp_pred_xgb=30.18, humi_pred_xgb=38.12, co02_pred_xgb=437.69, pm25_pred_xgb=17.6
2025-09-22 11:48:06 - INFO - Final Health score forecast: 75.13
2025-09-22 11:48:06 - INFO - Health Forecast Data: room_id=R106, hlth_scr_pred_xgb=75.12999725341797
2025-09-22 11:58:04 - INFO - Updated Room ID: urn:Pilot5:Room:R208 with Occupancy: 1
2025-09-22 11:58:04 - INFO - Environment_forecast: room_id=R208, temp_pred_xgb=32.53, humi_pred_xgb=35.53, co02_pred_xgb=416.04, pm25_pred_xgb=7.7
2025-09-22 11:58:04 - INFO - Final Health score forecast: 80.0
2025-09-22 11:58:04 - INFO - Health Forecast Data: room_id=R208, hlth_scr_pred_xgb=80.0
2025-09-22 11:59:04 - INFO - Updated Room ID: urn:Pilot5:Room:R208 with Occupancy: 0
2025-09-22 11:59:04 - INFO - Environment_forecast: room_id=R208, temp_pred_xgb=32.53, humi_pred_xgb=35.53, co02_pred_xgb=416.04, pm25_pred_xgb=7.7
2025-09-22 11:59:04 - INFO - Final Health score forecast: 80.0
2025-09-22 11:59:04 - INFO - Health Forecast Data: room_id=R208, hlth_scr_pred_xgb=80.0
2025-09-22 12:06:09 - INFO - Updated Room ID: urn:Pilot5:Room:R106 with Occupancy: 1
2025-09-22 12:06:09 - INFO - Environment_forecast: room_id=R106, temp_pred_xgb=30.18, humi_pred_xgb=38.12, co02_pred_xgb=437.69, pm25_pred_xgb=17.6
2025-09-22 12:06:09 - INFO - Final Health score forecast: 75.13
2025-09-22 12:06:09 - INFO - Health Forecast Data: room_id=R106, hlth_scr_pred_xgb=75.12999725341797
2025-09-22 12:16:04 - INFO - Updated Room ID: urn:Pilot5:Room:R106 with Occupancy: 0
2025-09-22 12:16:05 - INFO - Environment_forecast: room_id=R106, temp_pred_xgb=30.18, humi_pred_xgb=38.02, co02_pred_xgb=429.89, pm25_pred_xgb=15.64
2025-09-22 12:16:06 - INFO - Final Health score forecast: 76.83
2025-09-22 12:16:06 - INFO - Health Forecast Data: room_id=R106, hlth_scr_pred_xgb=76.83000183105469
2025-09-22 12:19:05 - INFO - Updated Room ID: urn:Pilot5:Room:R105 with Occupancy: 1
2025-09-22 12:19:05 - INFO - Environment_forecast: room_id=R105, temp_pred_xgb=31.34, humi_pred_xgb=37.13, co02_pred_xgb=429.64, pm25_pred_xgb=9.64
2025-09-22 12:19:05 - INFO - Final Health score forecast: 80.0
2025-09-22 12:19:05 - INFO - Health Forecast Data: room_id=R105, hlth_scr_pred_xgb=80.0

<pod> <logs>
```

Figure 108: Forecasting Logs for Updating Rooms' Health Score

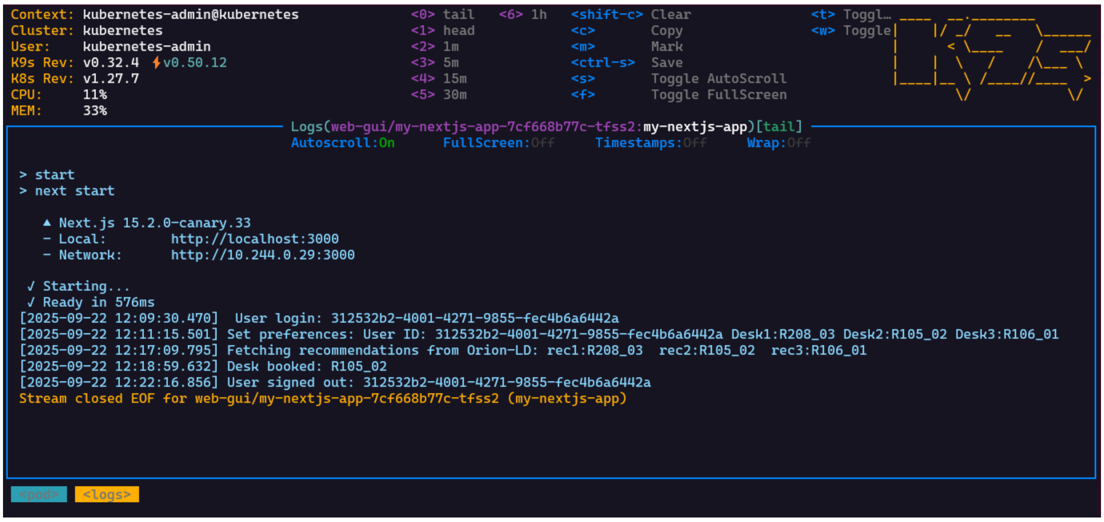
7. On user's exit, the desk reservation is released:



172.16.0.95:8090 says

Are you sure you want to leave the room?

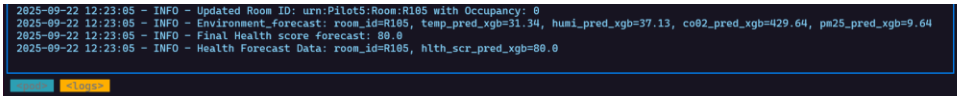
Figure 109: User Releases Desk



```
Context:  kubernetes-admin@kubernetes          <0> tail   <6> 1h   <shift-c> Clear             <t> Toggl…
Cluster:  kubernetes                           <1> head            <c>       Copy              <w> Toggle
User:     kubernetes-admin                     <2> 1m              <m>       Mark
K9s Rev:  v0.32.4 ⚡v0.50.12                    <3> 5m              <ctrl-s>  Save
K8s Rev:  v1.27.7                              <4> 15m             <s>       Toggle AutoScroll
CPU:      11%                                  <5> 30m             <f>       Toggle FullScreen
MEM:      33%
                         Logs(web-gui/my-nextjs-app-7cf668b77c-tfss2:my-nextjs-app)[tail]
                         Autoscroll:On      FullScreen:Off     Timestamps:Off     Wrap:Off

> start
> next start

   ▲ Next.js 15.2.0-canary.33
   - Local:        http://localhost:3000
   - Network:      http://10.244.0.29:3000

 ✓ Starting...
 ✓ Ready in 576ms
[2025-09-22 12:09:30.470]  User login: 312532b2-4001-4271-9855-fec4b6a6442a
[2025-09-22 12:11:15.501] Set preferences: User ID: 312532b2-4001-4271-9855-fec4b6a6442a Desk1:R208_03 Desk2:R105_02 Desk3:R106_01
[2025-09-22 12:17:09.795] Fetching recommendations from Orion-LD: rec1:R208_03  rec2:R105_02  rec3:R106_01
[2025-09-22 12:18:59.632] Desk booked: R105_02
[2025-09-22 12:22:16.856] User signed out: 312532b2-4001-4271-9855-fec4b6a6442a
Stream closed EOF for web-gui/my-nextjs-app-7cf668b77c-tfss2 (my-nextjs-app)

<pod>  <logs>
```

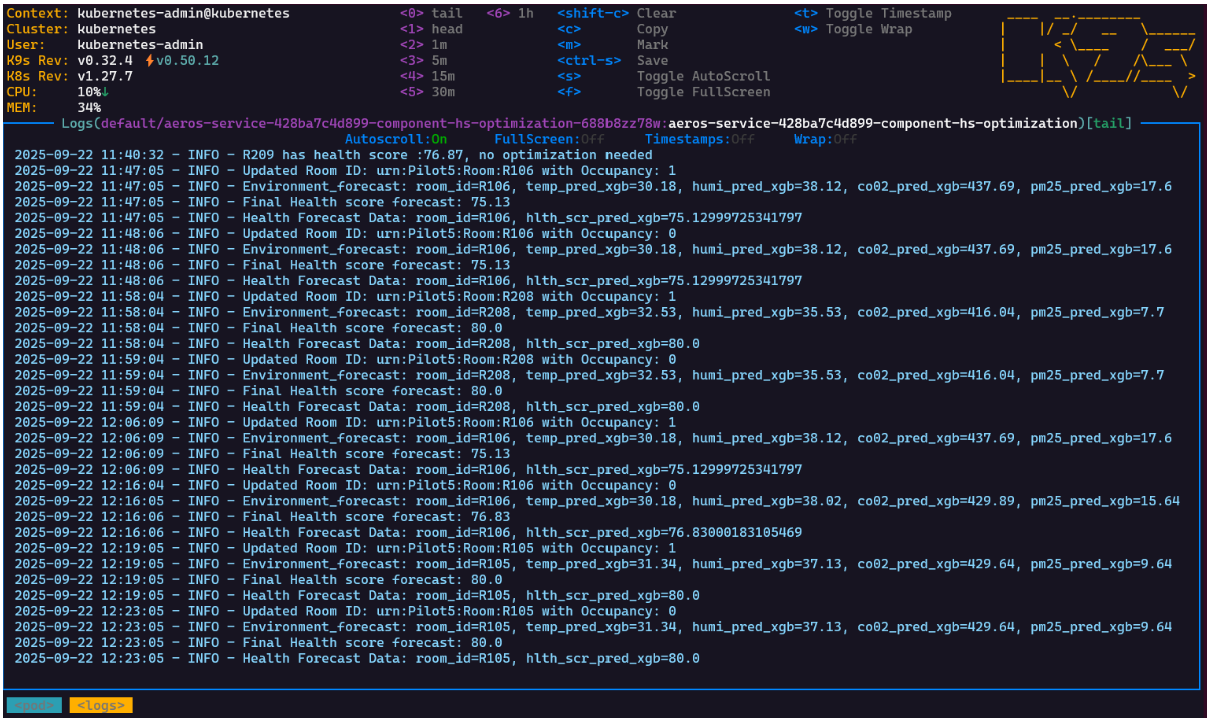
Figure 110: Web-gui Logs for Desk Release

8. The room's occupancy is updated:



```
2025-09-22 12:23:05 - INFO - Updated Room ID: urn:Pilot5:Room:R105 with Occupancy: 0
2025-09-22 12:23:05 - INFO - Environment_forecast: room_id=R105, temp_pred_xgb=31.34, humi_pred_xgb=37.13, co02_pred_xgb=429.64, pm25_pred_xgb=9.64
2025-09-22 12:23:05 - INFO - Final Health score forecast: 80.0
2025-09-22 12:23:05 - INFO - Health Forecast Data: room_id=R105, hlth_scr_pred_xgb=80.0

<pod>  <logs>
```

Figure 111: Room Occupancy Updated (Occupancy = 0)

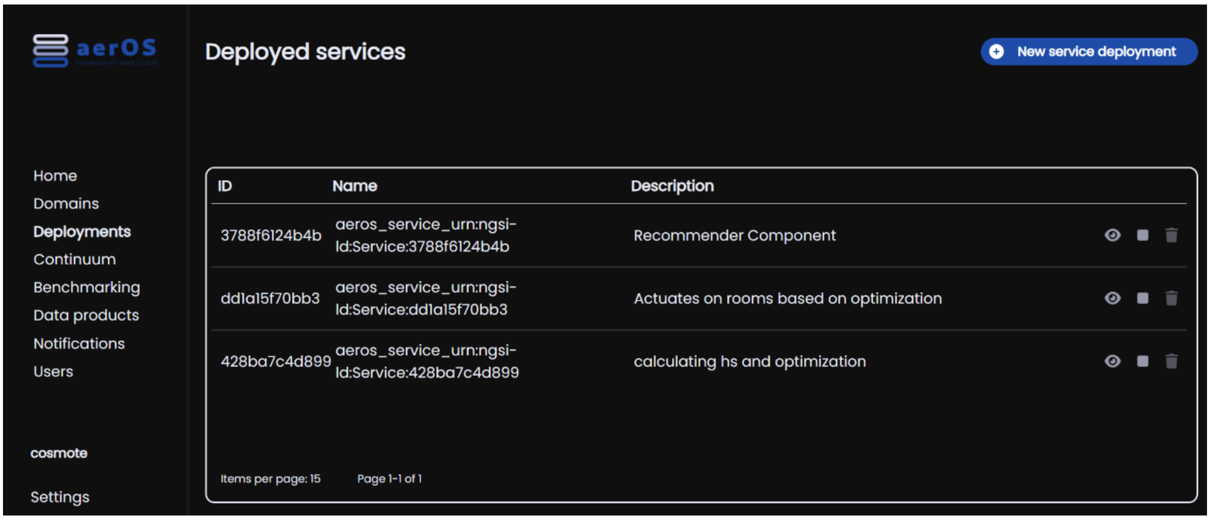


```
Context:  kubernetes-admin@kubernetes       <0> tail   <6> 1h    <shift-c> Clear             <t> Toggle Timestamp
Cluster:  kubernetes                        <1> head             <c>       Copy              <w> Toggle Wrap
User:     kubernetes-admin                  <2> 1m               <m>       Mark
K9s Rev:  v0.32.4 ⚡v0.50.12                <3> 5m               <ctrl-s>  Save
K8s Rev:  v1.27.7                           <4> 15m              <s>       Toggle AutoScroll
CPU:      10%↓                              <5> 30m              <f>       Toggle FullScreen
MEM:      34%
 Logs(default/aeros-service-428ba7c4d899-component-hs-optimization-688b8zz78w:aeros-service-428ba7c4d899-component-hs-optimization)[tail]
                 Autoscroll:On      FullScreen:Off      Timestamps:Off      Wrap:Off
 2025-09-22 11:40:32 - INFO - R209 has health score :76.87, no optimization needed
 2025-09-22 11:47:05 - INFO - Updated Room ID: urn:Pilot5:Room:R106 with Occupancy: 1
 2025-09-22 11:47:05 - INFO - Environment_forecast: room_id=R106, temp_pred_xgb=30.18, humi_pred_xgb=38.12, co02_pred_xgb=437.69, pm25_pred_xgb=17.6
 2025-09-22 11:47:05 - INFO - Final Health score forecast: 75.13
 2025-09-22 11:47:05 - INFO - Health Forecast Data: room_id=R106, hlth_scr_pred_xgb=75.12999725341797
 2025-09-22 11:48:06 - INFO - Updated Room ID: urn:Pilot5:Room:R106 with Occupancy: 0
 2025-09-22 11:48:06 - INFO - Environment_forecast: room_id=R106, temp_pred_xgb=30.18, humi_pred_xgb=38.12, co02_pred_xgb=437.69, pm25_pred_xgb=17.6
 2025-09-22 11:48:06 - INFO - Final Health score forecast: 75.13
 2025-09-22 11:48:06 - INFO - Health Forecast Data: room_id=R106, hlth_scr_pred_xgb=75.12999725341797
 2025-09-22 11:58:04 - INFO - Updated Room ID: urn:Pilot5:Room:R208 with Occupancy: 1
 2025-09-22 11:58:04 - INFO - Environment_forecast: room_id=R208, temp_pred_xgb=32.53, humi_pred_xgb=35.53, co02_pred_xgb=416.04, pm25_pred_xgb=7.7
 2025-09-22 11:58:04 - INFO - Final Health score forecast: 80.0
 2025-09-22 11:58:04 - INFO - Health Forecast Data: room_id=R208, hlth_scr_pred_xgb=80.0
 2025-09-22 11:59:04 - INFO - Updated Room ID: urn:Pilot5:Room:R208 with Occupancy: 0
 2025-09-22 11:59:04 - INFO - Environment_forecast: room_id=R208, temp_pred_xgb=32.53, humi_pred_xgb=35.53, co02_pred_xgb=416.04, pm25_pred_xgb=7.7
 2025-09-22 11:59:04 - INFO - Final Health score forecast: 80.0
 2025-09-22 11:59:04 - INFO - Health Forecast Data: room_id=R208, hlth_scr_pred_xgb=80.0
 2025-09-22 12:06:09 - INFO - Updated Room ID: urn:Pilot5:Room:R106 with Occupancy: 1
 2025-09-22 12:06:09 - INFO - Environment_forecast: room_id=R106, temp_pred_xgb=30.18, humi_pred_xgb=38.12, co02_pred_xgb=437.69, pm25_pred_xgb=17.6
 2025-09-22 12:06:09 - INFO - Final Health score forecast: 75.13
 2025-09-22 12:06:09 - INFO - Health Forecast Data: room_id=R106, hlth_scr_pred_xgb=75.12999725341797
 2025-09-22 12:16:04 - INFO - Updated Room ID: urn:Pilot5:Room:R106 with Occupancy: 0
 2025-09-22 12:16:05 - INFO - Environment_forecast: room_id=R106, temp_pred_xgb=30.18, humi_pred_xgb=38.02, co02_pred_xgb=429.89, pm25_pred_xgb=15.64
 2025-09-22 12:16:06 - INFO - Final Health score forecast: 76.83
 2025-09-22 12:16:06 - INFO - Health Forecast Data: room_id=R106, hlth_scr_pred_xgb=76.83000183105469
 2025-09-22 12:19:05 - INFO - Updated Room ID: urn:Pilot5:Room:R105 with Occupancy: 1
 2025-09-22 12:19:05 - INFO - Environment_forecast: room_id=R105, temp_pred_xgb=31.34, humi_pred_xgb=37.13, co02_pred_xgb=429.64, pm25_pred_xgb=9.64
 2025-09-22 12:19:05 - INFO - Final Health score forecast: 80.0
 2025-09-22 12:19:05 - INFO - Health Forecast Data: room_id=R105, hlth_scr_pred_xgb=80.0
 2025-09-22 12:23:05 - INFO - Updated Room ID: urn:Pilot5:Room:R105 with Occupancy: 0
 2025-09-22 12:23:05 - INFO - Environment_forecast: room_id=R105, temp_pred_xgb=31.34, humi_pred_xgb=37.13, co02_pred_xgb=429.64, pm25_pred_xgb=9.64
 2025-09-22 12:23:05 - INFO - Final Health score forecast: 80.0
 2025-09-22 12:23:05 - INFO - Health Forecast Data: room_id=R105, hlth_scr_pred_xgb=80.0

 <pod>  <logs>
```

Figure 112: Forecasting Logs for health Score Updates

Pilot 5 – Business Process 1 – Activity - 26 (P5-BP1-VA26): Pilot Services Created, Managed and Operated by aerOS Orchestrator

This validation activity evaluates the KPI 2.5.6 ‘Services directly managed by the aerOS orchestrator’ as reported in D5.5. The Pilot5 components deployment through the aerOS portal is depicted in the image below.

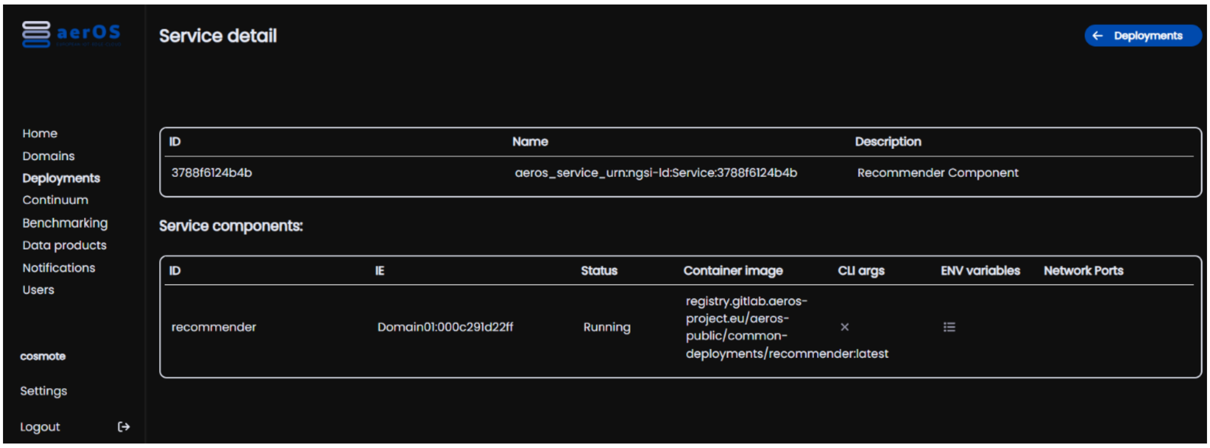


ID	Name	Description
3788f6124b4b	aeros_service_urn:ngsi-ld:Service:3788f6124b4b	Recommender Component
dd1a15f70bb3	aeros_service_urn:ngsi-ld:Service:dd1a15f70bb3	Actuates on rooms based on optimization
428ba7c4d899	aeros_service_urn:ngsi-ld:Service:428ba7c4d899	calculating hs and optimization

Items per page: 15   Page 1-1 of 1

Figure 113: Portal View of Deployed Pilot5 Services

Details about the deployment characteristics for these components namely Recommender, Forecasting and Actuator can be seen in the iamges below respectively.



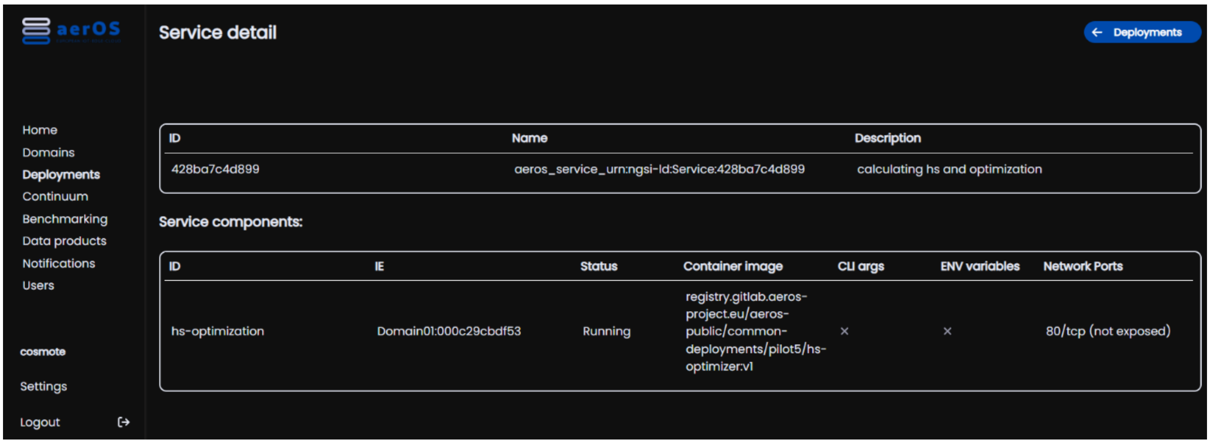
Service detail

ID	Name	Description
3788f6124b4b	aeros_service_urn:ngsi-ld:Service:3788f6124b4b	Recommender Component

Service components:

ID	IE	Status	Container image	CLI args	ENV variables	Network Ports
recommender	Domain01:000c291d22ff	Running	registry.gitlab.aeros-project.eu/aeros-public/common-deployments/recommender:latest	×	≡	

Figure 114: Recommender Component Deployment Status



Service detail

ID	Name	Description
428ba7c4d899	aeros_service_urn:ngsi-ld:Service:428ba7c4d899	calculating hs and optimization

Service components:

ID	IE	Status	Container image	CLI args	ENV variables	Network Ports
hs-optimization	Domain01:000c29cbdf53	Running	registry.gitlab.aeros-project.eu/aeros-public/common-deployments/pilot5/hs-optimizer:v1	×	×	80/tcp (not exposed)

Figure 115: Forecasting Component Deployment Status

Service detail

ID	Name	Description
dd1a15f70bb3	aeros_service_urn:ngsi-ld:Service:dd1a15f70bb3	Actuates on rooms based on optimization

Service components:

ID	IE	Status	Container image	CLI args	ENV variables	Network Ports
actuator-controller	Domain01:000c291d22ff	Running	registry.gitlab.aeros-project.eu/aeros-public/common-deployments/actuator-controller:latest	×	×	

Figure 116: Actuator Component Deployment Status

Pilot 5 – Business Process 1 – Activity - 27 (P5-BP1-VA27): Energy use Reduction

This validation activity evaluates the KPI 2.5.1 ‘Energy Use Reduction’ as reported in D5.5 and D5.6. The energy consumption is collected through the Shelly plugs and power meters installed per room, as described in D5.4. With the aerOS intelligence, the power consuming devices, such as an air conditioner are used less time, since the Forecaster upon evaluating the health score of a room, requests through the actuator the necessary adaptations (e.g. on/off). The validation is demonstrated by measuring energy consumption for a working week without the aerOS pilot running (1-5/9/25) and with the aerOS pilot running (8-12/9/25). Comparing the daily consumptions, the energy use reduction through the aerOS intelligence is evident, as shown in the below graphs for Rooms 105, 106, 208 and 209 respectively. The datasets are available in a repository of the aerOS Gitlab (https://gitlab.aeros-project.eu/pilots/pilot-5/data-pilot-5).

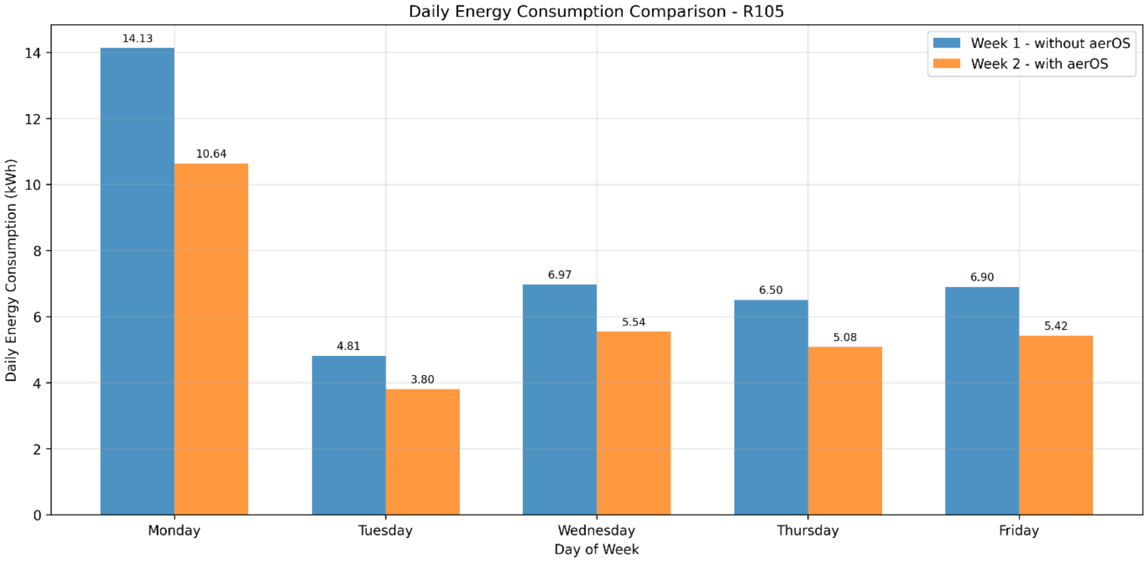


Figure 117: Daily (Day-of-Working-Week) Energy Consumption Before/After for Room 105

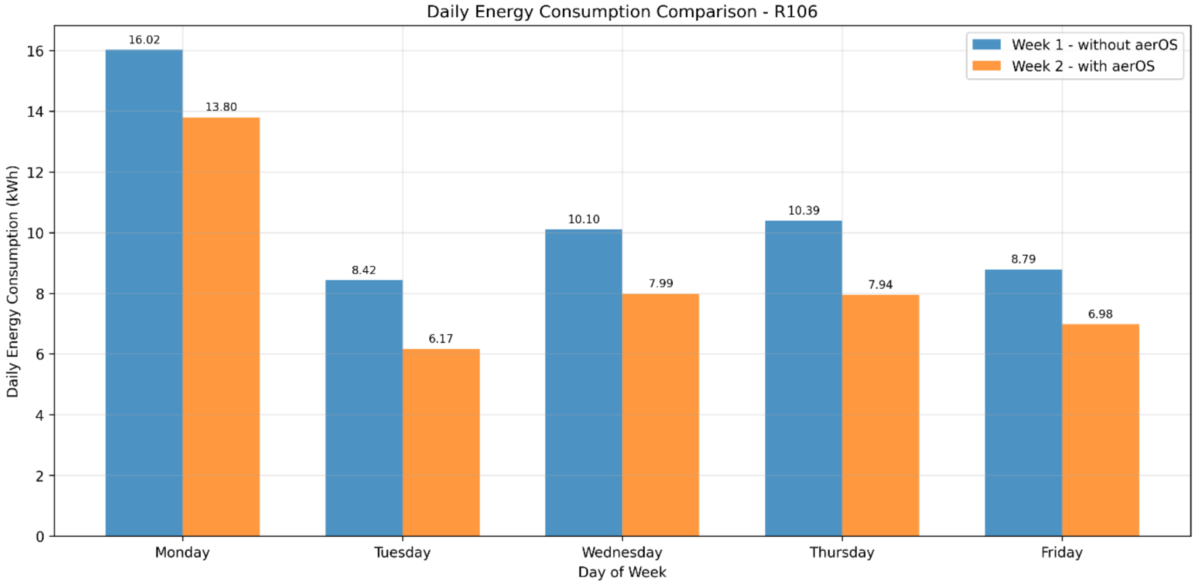


Figure 118:Daily (Day-of-Working-Week) Energy Consumption Before/After for Room 106

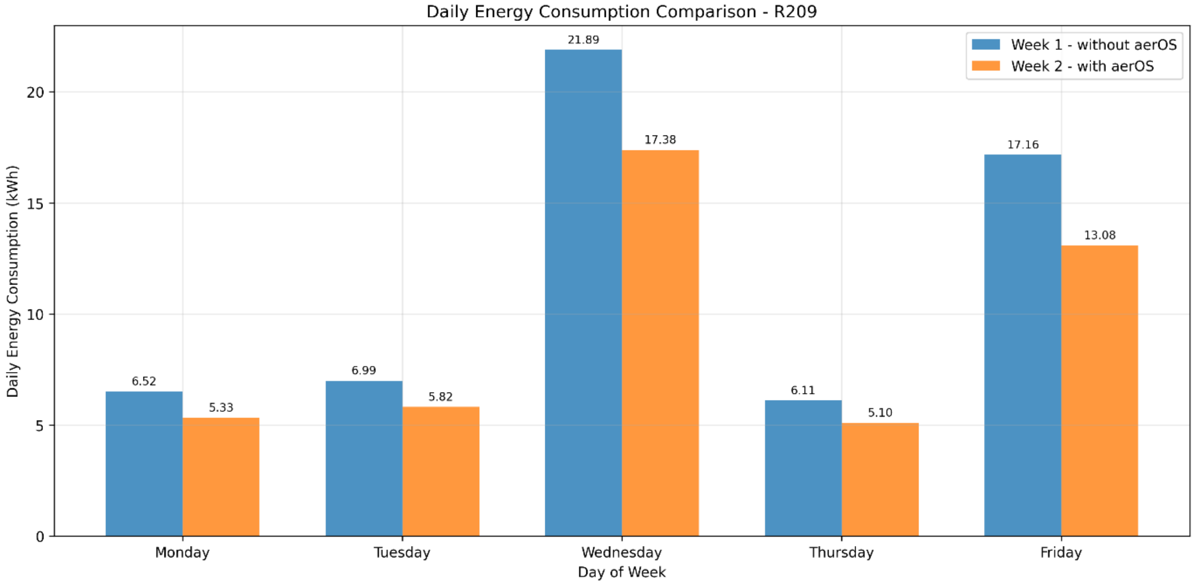


Figure 119: Daily (Day-of-Working-Week) Energy Consumption Before/After for Room 208

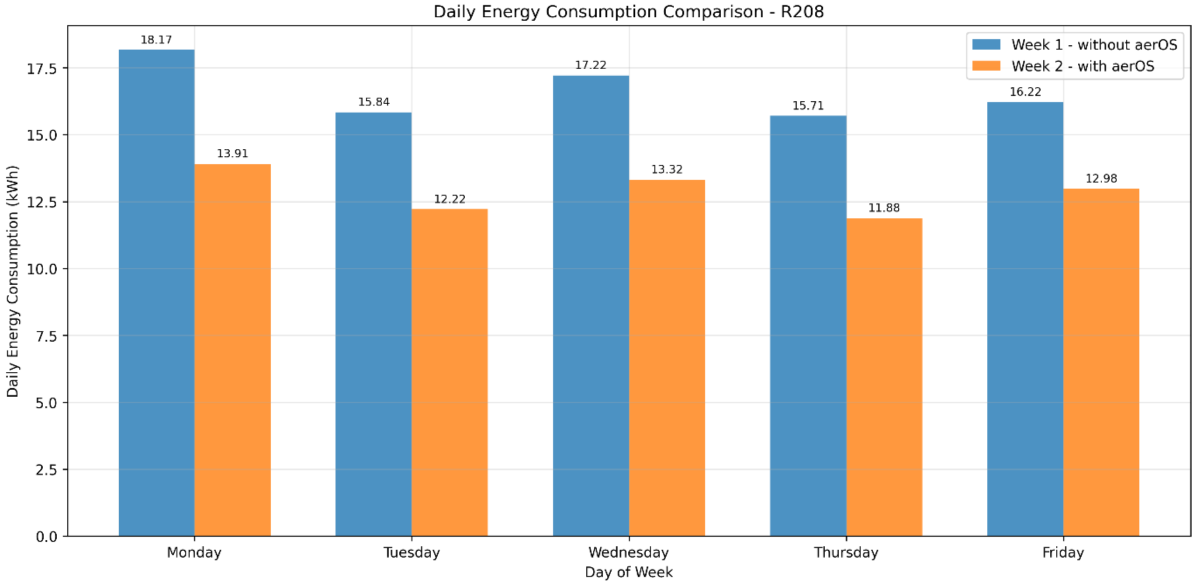


Figure 120: Daily (Day-of-Working-Week) Energy Consumption Before/After for Room 209

The AI optimization component was developed to maintain a healthy indoor environment while keeping energy consumption as low as possible. It operates by combining forecasts of environmental conditions (temperature, humidity, CO_2, and particulate matter) with the predicted energy consumption values. The forecasting and energy consumption predictions are both powered by XGBoost regression models, trained on room-specific environmental and contextual features. Using these inputs, the system evaluates whether the predicted environmental conditions deviate from their healthy thresholds and computes the minimal adjustments required to restore optimal conditions at the lowest possible energy cost. The optimization logic itself is implemented through a SciPy-based optimization engine that minimizes energy use subject to health constraints. This balance ensures that rooms remain comfortable and safe without unnecessary energy use. The optimization results define the target environmental state—rather than direct control actions—and are shared through the Orion-LD Context Broker for execution by downstream components such as Actuator. In this way, the system enables proactive, AI data-driven management of indoor spaces that jointly optimizes for both human well-being and energy efficiency.

Frugal AI has been applied to make the energy consumption prediction AI model more efficient by trying to find a model that is smaller yet as good quality as the original models, measured through the same or smaller RMSE[1] and the same or bigger R-squared metrics[2]. An example of a visualization of the energy consumption model behaviour during the architecture search can be seen in the figure below.

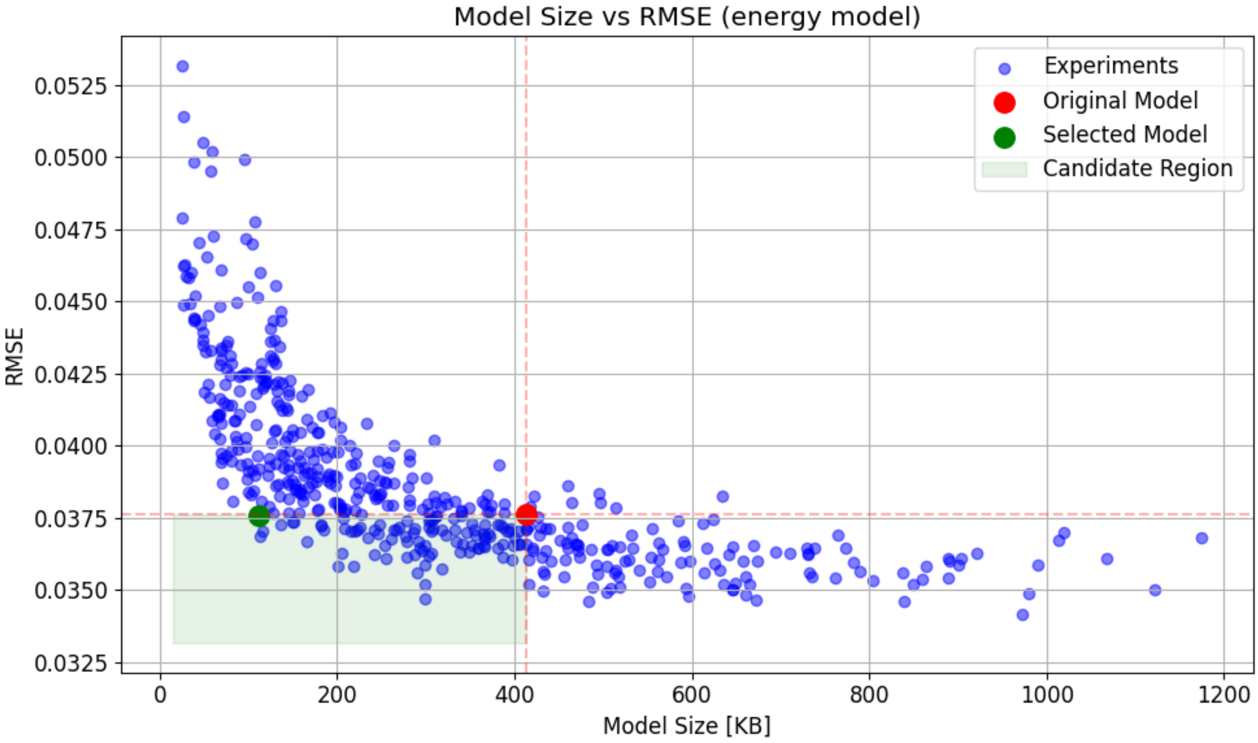


Figure 121: Visualization of the energy consumption model behaviour

Each blue dot is a test metric result (RMSE in this case) for a model trained with a set of hyperparameters. The green region contains models that are smaller and better than the original one. It can be sees that there are multiple candidates, however as per the pilot objective it has been chosen the smallest one that was, at least, as good as the original one.

The code and the results from the experiments are available in a repository of the aerOS Gitlab (https://gitlab.aeros-project.eu/pilots/pilot-5/forecasting-health-energy/-/tree/dev/src/app/fai).

Pilot 5 – Business Process 1 – Activity - 28 (P5-BP1-VA28): Edge Processing Performance Gains

This validation activity evaluates the KPI 2.5.2 ‘Edge processing performance gains’ measurement as reported in D5.5. The measurement of the Edge processing performance gains is a composite KPI that refers to gains such as:

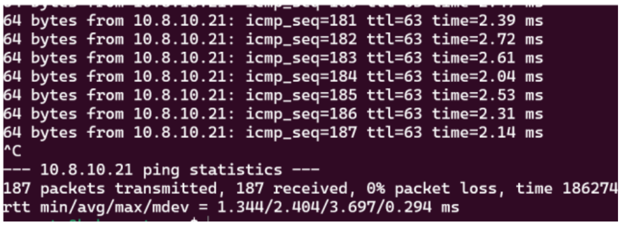
1. Exhibit average E2E Communication Latency < 100 ms for the aerOS nodes deployed locally (in the edge), measured through ping tools.

---

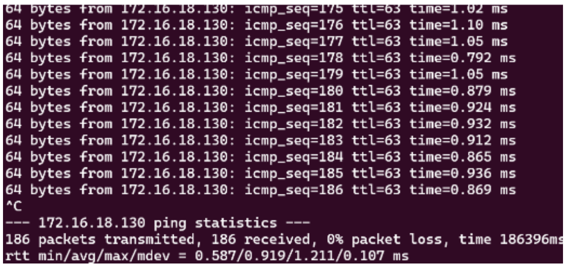
[1] **Root Mean Squared Error**, is a standard metric for measuring the difference between predicted and actual values, commonly used in regression analysis

[2] **R-Squared** (R^2 or the coefficient of determination) is a statistical measure in a regression model that determines the proportion of variance in the dependent variable that can be explained by the independent variable. In other words, r-squared shows how well the data fit the regression model (the goodness of fit).

Latency Average with aerOS has been reduced from 2.4 ms (left side of Figure 122) to 0.919 (right side of Figure 122) with aerOS transformation.



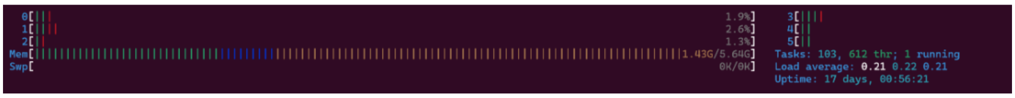
```
64 bytes from 10.8.10.21: icmp_seq=181 ttl=63 time=2.39 ms
64 bytes from 10.8.10.21: icmp_seq=182 ttl=63 time=2.72 ms
64 bytes from 10.8.10.21: icmp_seq=183 ttl=63 time=2.61 ms
64 bytes from 10.8.10.21: icmp_seq=184 ttl=63 time=2.04 ms
64 bytes from 10.8.10.21: icmp_seq=185 ttl=63 time=2.53 ms
64 bytes from 10.8.10.21: icmp_seq=186 ttl=63 time=2.31 ms
64 bytes from 10.8.10.21: icmp_seq=187 ttl=63 time=2.14 ms
^C
--- 10.8.10.21 ping statistics ---
187 packets transmitted, 187 received, 0% packet loss, time 186274
rtt min/avg/max/mdev = 1.344/2.404/3.697/0.294 ms
```



```
64 bytes from 172.16.18.130: icmp_seq=175 ttl=63 time=1.02 ms
64 bytes from 172.16.18.130: icmp_seq=176 ttl=63 time=1.10 ms
64 bytes from 172.16.18.130: icmp_seq=177 ttl=63 time=1.05 ms
64 bytes from 172.16.18.130: icmp_seq=178 ttl=63 time=0.792 ms
64 bytes from 172.16.18.130: icmp_seq=179 ttl=63 time=1.05 ms
64 bytes from 172.16.18.130: icmp_seq=180 ttl=63 time=0.879 ms
64 bytes from 172.16.18.130: icmp_seq=181 ttl=63 time=0.924 ms
64 bytes from 172.16.18.130: icmp_seq=182 ttl=63 time=0.932 ms
64 bytes from 172.16.18.130: icmp_seq=183 ttl=63 time=0.912 ms
64 bytes from 172.16.18.130: icmp_seq=184 ttl=63 time=0.865 ms
64 bytes from 172.16.18.130: icmp_seq=185 ttl=63 time=0.936 ms
64 bytes from 172.16.18.130: icmp_seq=186 ttl=63 time=0.869 ms
^C
--- 172.16.18.130 ping statistics ---
186 packets transmitted, 186 received, 0% packet loss, time 186396ms
rtt min/avg/max/mdev = 0.587/0.919/1.211/0.107 ms
```

Figure 122: Baseline Latency Average between two application nodes (Before aerOS) and Latency Average between Pilot nodes (with aerOS)

2. Demonstrate the gains of KubeEdge (with aerOS transformation) vs. K8 deployments (baseline) utilizing light devices at the far edge gaining 20 % less memory resources consumption comparing the cluster reported average measurement values. Memory usage with aerOS has been reduced from 1.6 Gbps (Figure 123) to 730 Mbps (Figure 124) with aerOS transformation.



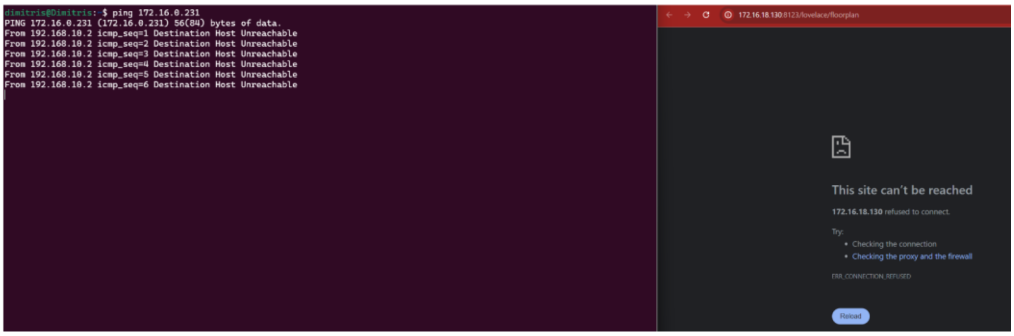
```
  0[|||                                              1.9%]    3[||||
  1[||||                                             2.6%]    4[||
  2[||                                               1.3%]    5[||
Mem[||||||||||||||||||||||||||||||||||||||||||||1.43G/5.64G]  Tasks: 103, 612 thr; 1 running
Swp[                                               0K/0K]     Load average: 0.21 0.22 0.21
                                                              Uptime: 17 days, 00:56:21
```

Figure 123: Baseline Memory Usage of IoT K8 (before aerOS)

```
  0[|||||                                              3.4%]   Tasks: 62, 245 thr; 1 running
  1[||||||                                             4.6%]   Load average: 0.35 0.20 0.12
Mem[|||||||||||||||||||||||||||||||||||||||||||||||730M/3.82G]  Uptime: 17 days, 00:44:24
Swp[|                                            1.02M/3.83G]
```

Figure 124: Memory Usage (with aerOS)

3. Demonstrate the gains of Kube Edge for service resilience, measuring the service recovery time showcasing that the pilot services at the edge still operate even when the master node is down or there is a network connectivity issue. In the baseline setup (before aerOS) when the K8 master node or network is down (left window of Figure 125) the IoT Application is not running (right window of Figure 125).



```
dimitris@Dimitris:~$ ping 172.16.0.231
PING 172.16.0.231 (172.16.0.231) 56(84) bytes of data.
From 192.168.10.2 icmp_seq=1 Destination Host Unreachable
From 192.168.10.2 icmp_seq=2 Destination Host Unreachable
From 192.168.10.2 icmp_seq=3 Destination Host Unreachable
From 192.168.10.2 icmp_seq=4 Destination Host Unreachable
From 192.168.10.2 icmp_seq=5 Destination Host Unreachable
From 192.168.10.2 icmp_seq=6 Destination Host Unreachable
```

172.16.18.130:8123/lovelace/floorplan

This site can't be reached

172.16.18.130 refused to connect.

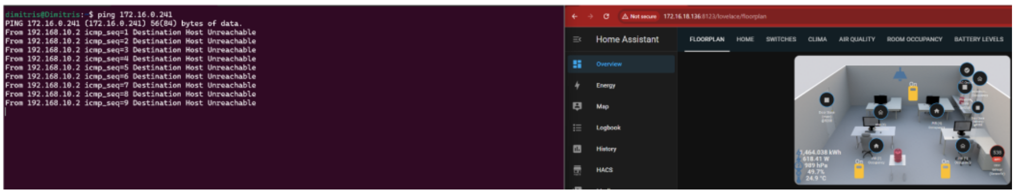
Try:

- Checking the connection
- Checking the proxy and the firewall

ERR_CONNECTION_REFUSED

Figure 125: Service Recovery Baseline (before aerOS) – IoT App is Down when Master Node is Down

However, employing aerOS kubeEdge, the IoT Application (right window of Figure 126**¡Error! No se encuentra el origen de la referencia.**) is running when K8 master node or network is down (left window of Figure 125).

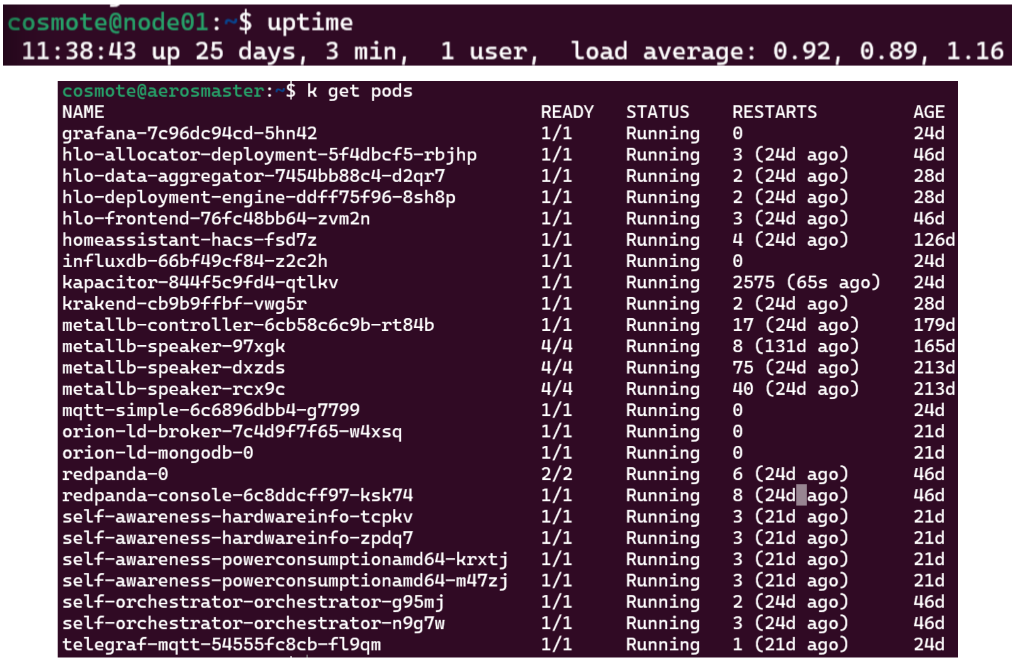


```
dimitris@Dimitris:~$ ping 172.16.0.241
PING 172.16.0.241 (172.16.0.241) 56(84) bytes of data.
From 192.168.10.2 icmp_seq=1 Destination Host Unreachable
From 192.168.10.2 icmp_seq=2 Destination Host Unreachable
From 192.168.10.2 icmp_seq=3 Destination Host Unreachable
From 192.168.10.2 icmp_seq=4 Destination Host Unreachable
From 192.168.10.2 icmp_seq=5 Destination Host Unreachable
From 192.168.10.2 icmp_seq=6 Destination Host Unreachable
From 192.168.10.2 icmp_seq=7 Destination Host Unreachable
From 192.168.10.2 icmp_seq=8 Destination Host Unreachable
From 192.168.10.2 icmp_seq=9 Destination Host Unreachable
```

Figure 126: Service Recovery Baseline (before aerOS) – IoT App is Running when Master Node is Down

Pilot 5 – Business Process 1 – Activity - 29 (P5-BP1-VA29): Service Availability within the aerOS IE

This validation activity evaluates the KPI 2.5.4 ‘Service availability’ as reported in D5.5. It measures the uptime of the pilot nodes, for a period of one month using raw data from the host (see Figure 127). A known maintenance activity is included in this period explaining the uptime to be 25 days.



```
cosmote@node01:~$ uptime
 11:38:43 up 25 days, 3 min,  1 user,  load average: 0.92, 0.89, 1.16
```

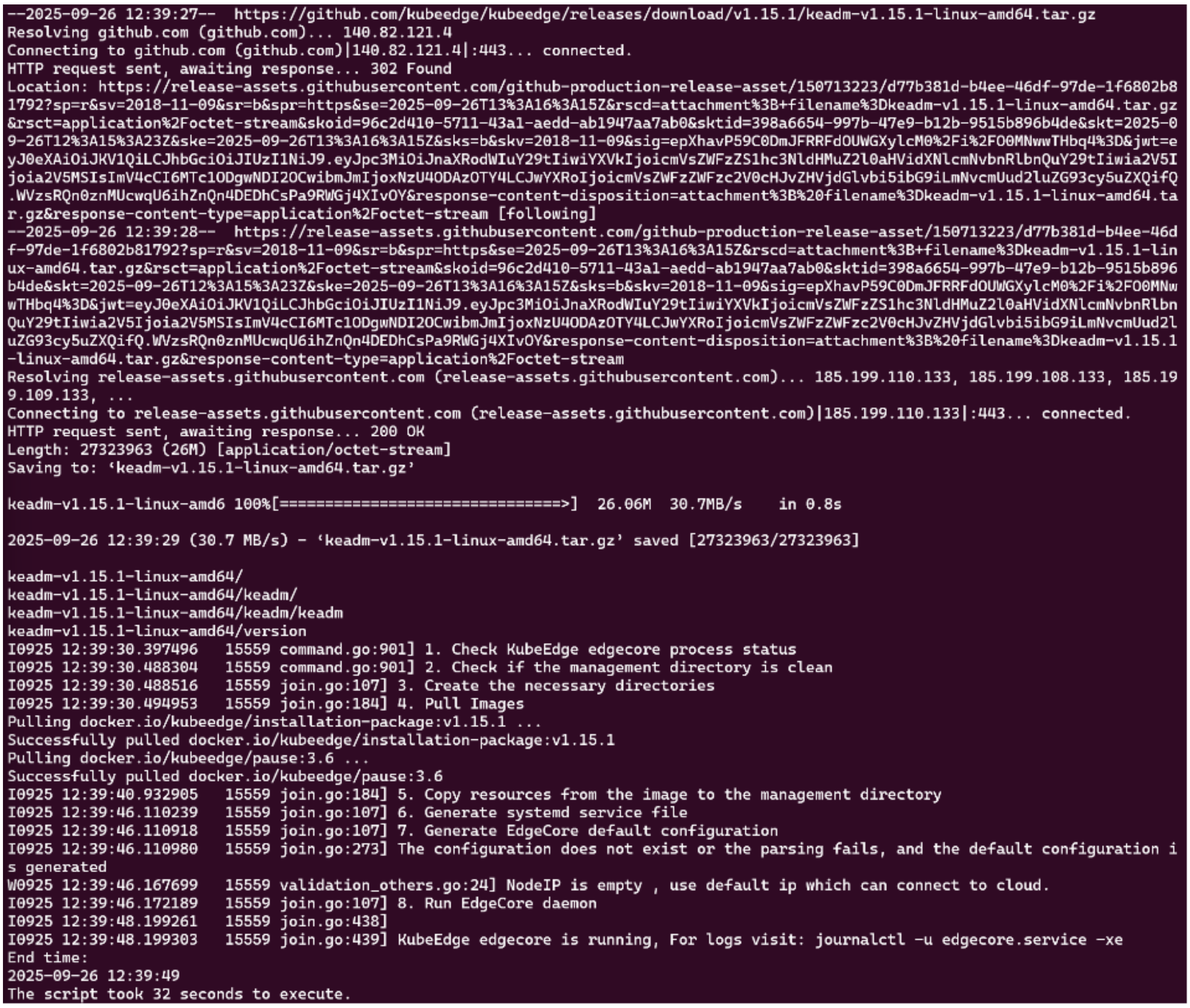
```
cosmote@aerosmaster:~$ k get pods
NAME                                         READY   STATUS    RESTARTS         AGE
grafana-7c96dc94cd-5hn42                     1/1     Running   0                24d
hlo-allocator-deployment-5f4dbcf5-rbjhp      1/1     Running   3 (24d ago)      46d
hlo-data-aggregator-7454bb88c4-d2qr7         1/1     Running   2 (24d ago)      28d
hlo-deployment-engine-ddff75f96-8sh8p        1/1     Running   2 (24d ago)      28d
hlo-frontend-76fc48bb64-zvm2n                1/1     Running   3 (24d ago)      46d
homeassistant-hacs-fsd7z                     1/1     Running   4 (24d ago)      126d
influxdb-66bf49cf84-z2c2h                    1/1     Running   0                24d
kapacitor-844f5c9fd4-qtlkv                   1/1     Running   2575 (65s ago)   24d
krakend-cb9b9ffbf-vwg5r                      1/1     Running   2 (24d ago)      28d
metallb-controller-6cb58c6c9b-rt84b          1/1     Running   17 (24d ago)     179d
metallb-speaker-97xgk                        4/4     Running   8 (131d ago)     165d
metallb-speaker-dxzds                        4/4     Running   75 (24d ago)     213d
metallb-speaker-rcx9c                        4/4     Running   40 (24d ago)     213d
mqtt-simple-6c6896dbb4-g7799                 1/1     Running   0                24d
orion-ld-broker-7c4d9f7f65-w4xsq             1/1     Running   0                21d
orion-ld-mongodb-0                           1/1     Running   0                21d
redpanda-0                                   2/2     Running   6 (24d ago)      46d
redpanda-console-6c8ddcff97-ksk74            1/1     Running   8 (24d ago)      46d
self-awareness-hardwareinfo-tcpkv            1/1     Running   3 (21d ago)      21d
self-awareness-hardwareinfo-zpdq7            1/1     Running   3 (21d ago)      21d
self-awareness-powerconsumptionamd64-krxtj   1/1     Running   3 (21d ago)      21d
self-awareness-powerconsumptionamd64-m47zj   1/1     Running   3 (21d ago)      21d
self-orchestrator-orchestrator-g95mj         1/1     Running   2 (24d ago)      46d
self-orchestrator-orchestrator-n9g7w         1/1     Running   3 (24d ago)      46d
telegraf-mqtt-54555fc8cb-fl9qm               1/1     Running   1 (21d ago)      24d
```

Figure 127: Pilot5 master node uptime

Pilot 5 – Business Process 1 – Activity - 30 (P5-BP1-VA30): Service Creation / Scalability

This validation activity evaluates the KPI 2.5.5 ‘Service Creation / Scalability’ as reported in D5.5. The validation activity involves measuring the time to deploy a new node (aeros-node’) in the KubeEdge cluster. As it can be seen in Figure 128, that depicts the logs of the dynamic deployment process of the aerOS node, the service creation time is 32 secs.

```
Start time:
2025-09-26 12:39:17
--2025-09-26 12:39:17--  https://github.com/containerd/containerd/releases/download/v1.7.11/containerd-1.7.11-linux-amd64.tar.gz
Resolving github.com (github.com)... 140.82.121.4
Connecting to github.com (github.com)|140.82.121.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://release-assets.githubusercontent.com/github-production-release-asset/46089560/a6b15b4a-6139-43e7-a0ad-0baf81494
c1b?sp=r&sv=2018-11-09&sr=b&spr=https&se=2025-09-26T13%3A17%3A46Z&rscd=attachment%3B+filename%3Dcontainerd-1.7.11-linux-amd64.tar
.gz&rsct=application%2Foctet-stream&skoid=96c2d410-5711-43a1-aedd-ab1947aa7ab0&sktid=398a6654-997b-47e9-b12b-9515b896b4de&skt=202
5-09-26T12%3A17%3A36Z&ske=2025-09-26T13%3A17%3A46Z&sks=b&skv=2018-11-09&sig=kF8QMnpl0GwHecqNpWPQw1pQ4SBRXHuvy1a3kghcxnk%3D&jwt=ey
J0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmVsZWFzZS1hc3NldHMuZ2l0aHVidXNlcmNvbnRlbnQuY29tIiwia2V5Ij
oia2V5MSIsImV4cCI6MTc1ODgwNDI1OCwibmJmIjoxNzU4ODAzOTU4LCJwYXRoIjoicmVsZWFzZWFzc2V0cHJvZHVjdGlvbi5ibG9iLmNvcmUud2luZG93cy5uZXQifQ.
sMBlEI1Yp4UnpI4e2GjIV7aFBxoOdCvubdACYkaQfmI&response-content-disposition=attachment%3B%20filename%3Dcontainerd-1.7.11-linux-amd64
.tar.gz&response-content-type=application%2Foctet-stream [following]
--2025-09-26 12:39:18--  https://release-assets.githubusercontent.com/github-production-release-asset/46089560/a6b15b4a-6139-43e7
-a0ad-0baf81494c1b?sp=r&sv=2018-11-09&sr=b&spr=https&se=2025-09-26T13%3A17%3A46Z&rscd=attachment%3B+filename%3Dcontainerd-1.7.11-
linux-amd64.tar.gz&rsct=application%2Foctet-stream&skoid=96c2d410-5711-43a1-aedd-ab1947aa7ab0&sktid=398a6654-997b-47e9-b12b-9515b
896b4de&skt=2025-09-26T12%3A17%3A36Z&ske=2025-09-26T13%3A17%3A46Z&sks=b&skv=2018-11-09&sig=kF8QMnpl0GwHecqNpWPQw1pQ4SBRXHuvy1a3kg
hcxnk%3D&jwt=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmVsZWFzZS1hc3NldHMuZ2l0aHVidXNlcmNvbnRlbnQ
uY29tIiwia2V5Ijoia2V5MSIsImV4cCI6MTc1ODgwNDI1OCwibmJmIjoxNzU4ODAzOTU4LCJwYXRoIjoicmVsZWFzZWFzc2V0cHJvZHVjdGlvbi5ibG9iLmNvcmUud2lu
ZG93cy5uZXQifQ.sMBlEI1Yp4UnpI4e2GjIV7aFBxoOdCvubdACYkaQfmI&response-content-disposition=attachment%3B%20filename%3Dcontainerd-1.7
.11-linux-amd64.tar.gz&response-content-type=application%2Foctet-stream
Resolving release-assets.githubusercontent.com (release-assets.githubusercontent.com)... 185.199.111.133, 185.199.108.133, 185.19
9.110.133, ...
Connecting to release-assets.githubusercontent.com (release-assets.githubusercontent.com)|185.199.111.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 47970788 (46M) [application/octet-stream]
Saving to: ‘containerd-1.7.11-linux-amd64.tar.gz’

containerd-1.7.11-linux- 100%[===============================>]  45.75M  24.8MB/s    in 1.8s

2025-09-26 12:39:20 (24.8 MB/s) - ‘containerd-1.7.11-linux-amd64.tar.gz’ saved [47970788/47970788]
```



```
--2025-09-26 12:39:27--  https://github.com/kubeedge/kubeedge/releases/download/v1.15.1/keadm-v1.15.1-linux-amd64.tar.gz
Resolving github.com (github.com)... 140.82.121.4
Connecting to github.com (github.com)|140.82.121.4|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://release-assets.githubusercontent.com/github-production-release-asset/150713223/d77b381d-b4ee-46df-97de-1f6802b8
1792?sp=r&sv=2018-11-09&sr=b&spr=https&se=2025-09-26T13%3A16%3A15Z&rscd=attachment%3B+filename%3Dkeadm-v1.15.1-linux-amd64.tar.gz
&rsct=application%2Foctet-stream&skoid=96c2d410-5711-43a1-aedd-ab1947aa7ab0&sktid=398a6654-997b-47e9-b12b-9515b896b4de&skt=2025-0
9-26T12%3A15%3A23Z&ske=2025-09-26T13%3A16%3A15Z&sks=b&skv=2018-11-09&sig=epXhavP59C0DmJFRRFdOUWGXylcM0%2Fi%2FO0MNwwTHbq4%3D&jwt=e
yJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmVsZWFzZS1hc3NldHMuZ2l0aHVidXNlcmNvbnRlbnQuY29tIiwia2V5I
joia2V5MSIsImV4cCI6MTc1ODgwNDI2OCwibmJmIjoxNzU4ODAzOTY4LCJwYXRoIjoicmVsZWFzZWFzc2V0cHJvZHVjdGlvbi5ibG9iLmNvcmUud2luZG93cy5uZXQifQ
.WVzsRQn0znMUcwqU6ihZnQn4DEDhCsPa9RWGj4XIvOY&response-content-disposition=attachment%3B%20filename%3Dkeadm-v1.15.1-linux-amd64.ta
r.gz&response-content-type=application%2Foctet-stream [following]
--2025-09-26 12:39:28--  https://release-assets.githubusercontent.com/github-production-release-asset/150713223/d77b381d-b4ee-46d
f-97de-1f6802b81792?sp=r&sv=2018-11-09&sr=b&spr=https&se=2025-09-26T13%3A16%3A15Z&rscd=attachment%3B+filename%3Dkeadm-v1.15.1-lin
ux-amd64.tar.gz&rsct=application%2Foctet-stream&skoid=96c2d410-5711-43a1-aedd-ab1947aa7ab0&sktid=398a6654-997b-47e9-b12b-9515b896
b4de&skt=2025-09-26T12%3A15%3A23Z&ske=2025-09-26T13%3A16%3A15Z&sks=b&skv=2018-11-09&sig=epXhavP59C0DmJFRRFdOUWGXylcM0%2Fi%2FO0MNw
wTHbq4%3D&jwt=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmVsZWFzZS1hc3NldHMuZ2l0aHVidXNlcmNvbnRlbn
QuY29tIiwia2V5Ijoia2V5MSIsImV4cCI6MTc1ODgwNDI2OCwibmJmIjoxNzU4ODAzOTY4LCJwYXRoIjoicmVsZWFzZWFzc2V0cHJvZHVjdGlvbi5ibG9iLmNvcmUud2l
uZG93cy5uZXQifQ.WVzsRQn0znMUcwqU6ihZnQn4DEDhCsPa9RWGj4XIvOY&response-content-disposition=attachment%3B%20filename%3Dkeadm-v1.15.1
-linux-amd64.tar.gz&response-content-type=application%2Foctet-stream
Resolving release-assets.githubusercontent.com (release-assets.githubusercontent.com)... 185.199.110.133, 185.199.108.133, 185.19
9.109.133, ...
Connecting to release-assets.githubusercontent.com (release-assets.githubusercontent.com)|185.199.110.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 27323963 (26M) [application/octet-stream]
Saving to: ‘keadm-v1.15.1-linux-amd64.tar.gz’

keadm-v1.15.1-linux-amd6 100%[===============================>]  26.06M  30.7MB/s    in 0.8s

2025-09-26 12:39:29 (30.7 MB/s) - ‘keadm-v1.15.1-linux-amd64.tar.gz’ saved [27323963/27323963]

keadm-v1.15.1-linux-amd64/
keadm-v1.15.1-linux-amd64/keadm/
keadm-v1.15.1-linux-amd64/keadm/keadm
keadm-v1.15.1-linux-amd64/version
I0925 12:39:30.397496   15559 command.go:901] 1. Check KubeEdge edgecore process status
I0925 12:39:30.488304   15559 command.go:901] 2. Check if the management directory is clean
I0925 12:39:30.488516   15559 join.go:107] 3. Create the necessary directories
I0925 12:39:30.494953   15559 join.go:184] 4. Pull Images
Pulling docker.io/kubeedge/installation-package:v1.15.1 ...
Successfully pulled docker.io/kubeedge/installation-package:v1.15.1
Pulling docker.io/kubeedge/pause:3.6 ...
Successfully pulled docker.io/kubeedge/pause:3.6
I0925 12:39:40.932905   15559 join.go:184] 5. Copy resources from the image to the management directory
I0925 12:39:46.110239   15559 join.go:107] 6. Generate systemd service file
I0925 12:39:46.110918   15559 join.go:107] 7. Generate EdgeCore default configuration
I0925 12:39:46.110980   15559 join.go:273] The configuration does not exist or the parsing fails, and the default configuration i
s generated
W0925 12:39:46.167699   15559 validation_others.go:24] NodeIP is empty , use default ip which can connect to cloud.
I0925 12:39:46.172189   15559 join.go:107] 8. Run EdgeCore daemon
I0925 12:39:48.199261   15559 join.go:438]
I0925 12:39:48.199303   15559 join.go:439] KubeEdge edgecore is running, For logs visit: journalctl -u edgecore.service -xe
End time:
2025-09-26 12:39:49
The script took 32 seconds to execute.
```

Figure 128: Time to deploy in aerOS node in KubeEdge

Pilot 5 – Business Process 1 – Activity - 31 (P5-BP1-VA31): Improvement of Air Quality

This validation activity evaluates the KPI 2.5.7 'Improvement of air quality' as reported in D5.5. The target is to demonstrate for the rooms of the pilot, and the specific demo situation, that the max CO2 is reduced approximately 20% and is lower than 1000 ppm in all cases. This is demonstrated by measuring CO2 during a working week without the aerOS pilot running (1-5/9/25) and with the aerOS pilot running (8-12/9/25). The related results for the COSMOTE building rooms 105, 106, 208, 209 that are part of the aerOS Pilot are depicted in the graphs below, respectively. It is evident from these diagrams that the CO2 measured was significantly improved with the aerOS system while the max CO2 was significantly below 1000 ppm in all cases. The datasets are available in https://gitlab.aeros-project.eu/pilots/pilot-5/data-pilot-5.git.

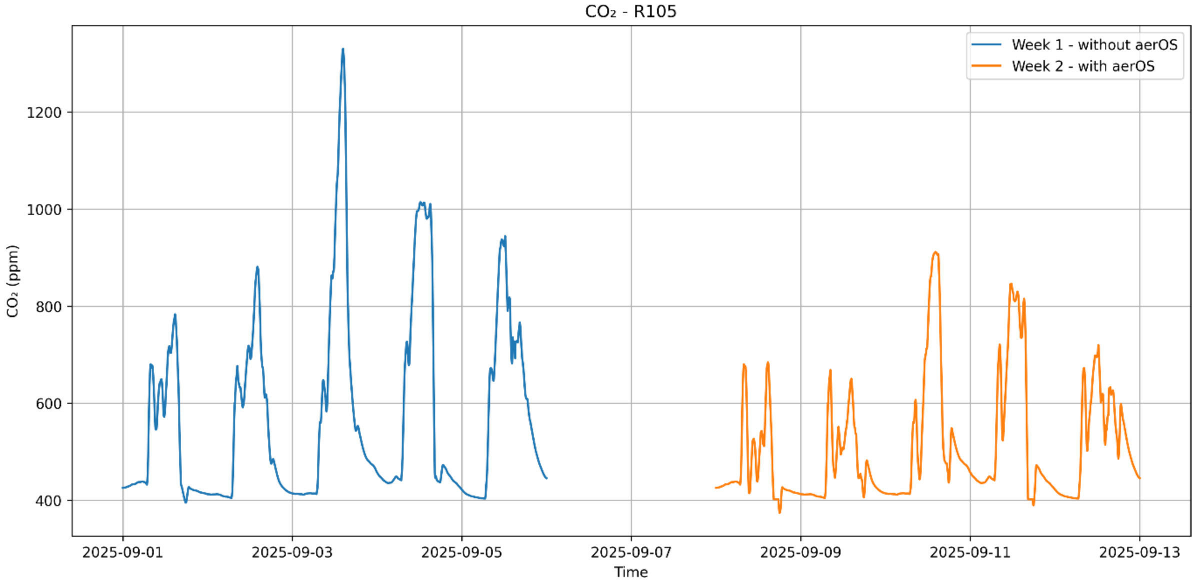


Figure 129: Daily (Day-of-Working-Week) CO2 (ppm) Before/After for Room 105

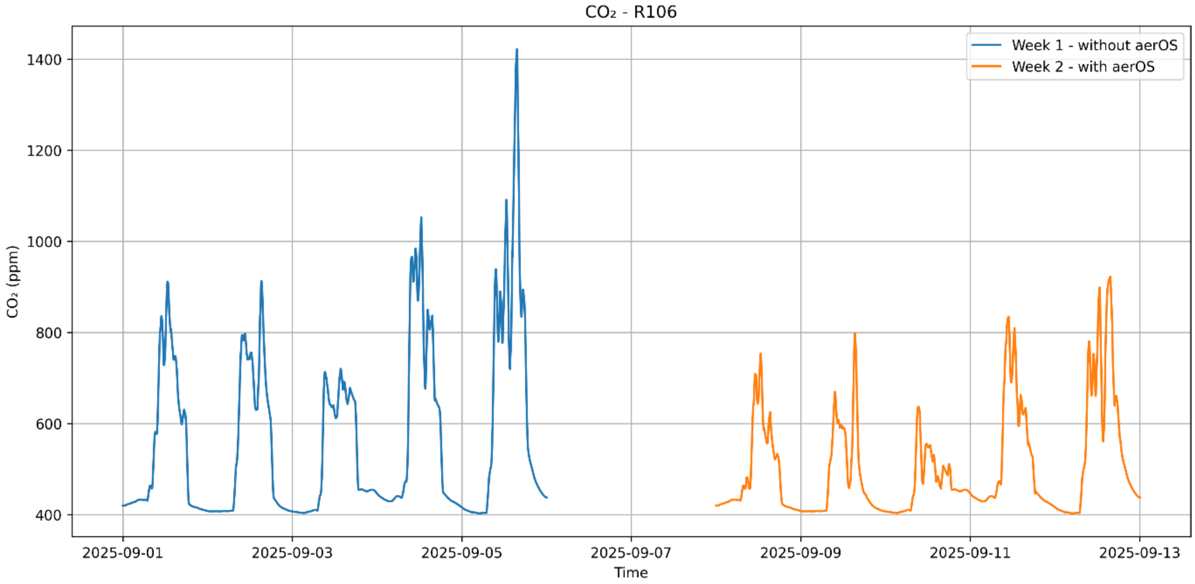


Figure 130: Daily (Day-of-Working-Week) CO2 (ppm) Before/After for Room 106

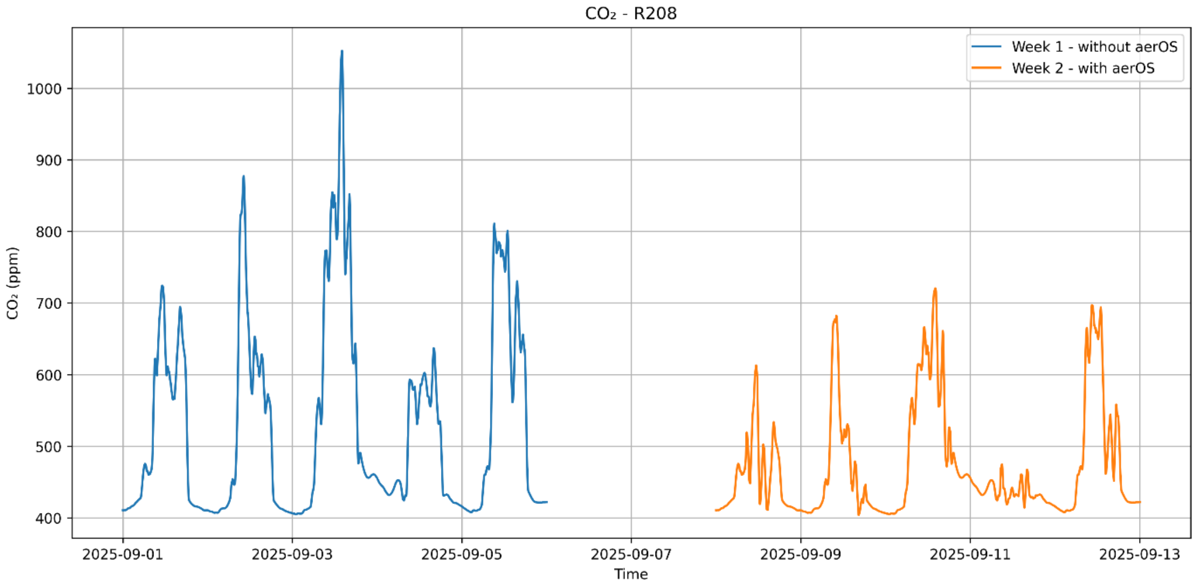


Figure 131: Daily (Day-of-Working-Week) CO2 (ppm) Before/After for Room 208

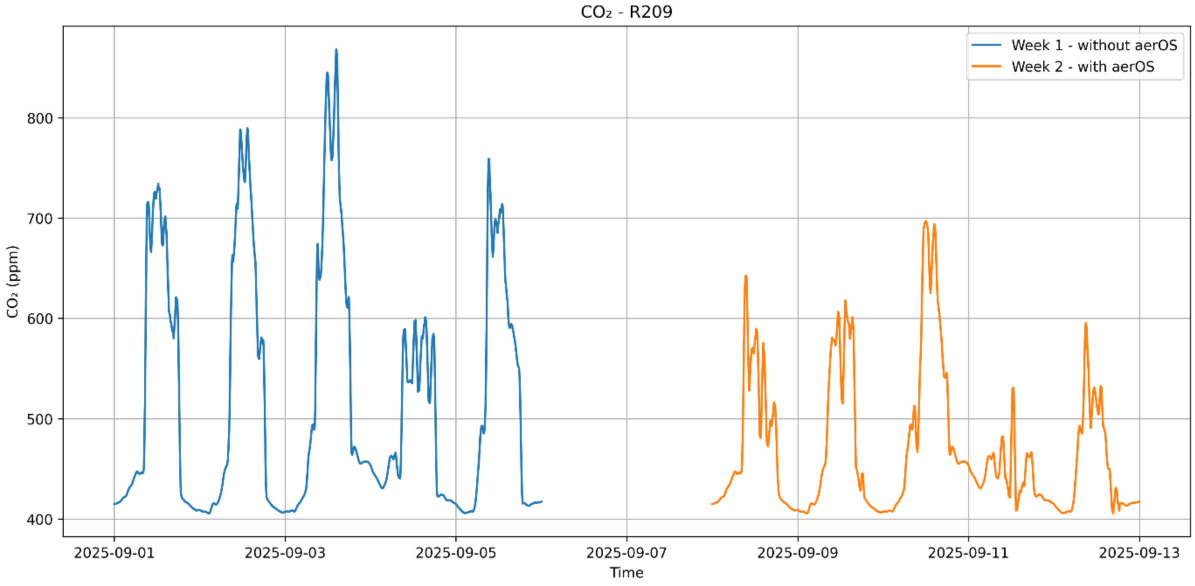


Figure 132: Daily (Day-of-Working-Week) CO2 (ppm) Before/After for Room 209

The Health Index AI estimates the overall indoor environmental quality by combining multiple parameters—temperature, humidity, CO_2, and particulate matter (PM1, PM2.5, PM10)—into a single, interpretable score ranging from 0 to 100. It uses XGBoost regression models trained on historical sensor data to forecast each environmental variable and assess whether future conditions remain within healthy ranges. By comparing predicted values against literature-based optimal thresholds, the Health Index quantifies how favorable the indoor environment is for human comfort and safety, enabling proactive management and early detection of unhealthy conditions.

Frugal AI approach has been applied to make the energy consumption/prediction all the environmental AI models more efficient by trying to find a model that is smaller yet as good quality as the original models, measured through the same or smaller RMSE and the same or bigger R-squared metrics. During the search procedure for each model a set of parameters was selected randomly from heuristically defined ranges, e.g., the number of estimators, the learning rate, etc. For each target variable, the search procedure prepared 500 different models which were trained and tested. An example behaviour of models of predicting PM10 levels *during the search* can be seen in the next figure. The method was successfully applied to all six of the models that predicted the environmental parameters. Therefore, for each original model, we managed to find a model that was smaller and equally as good.

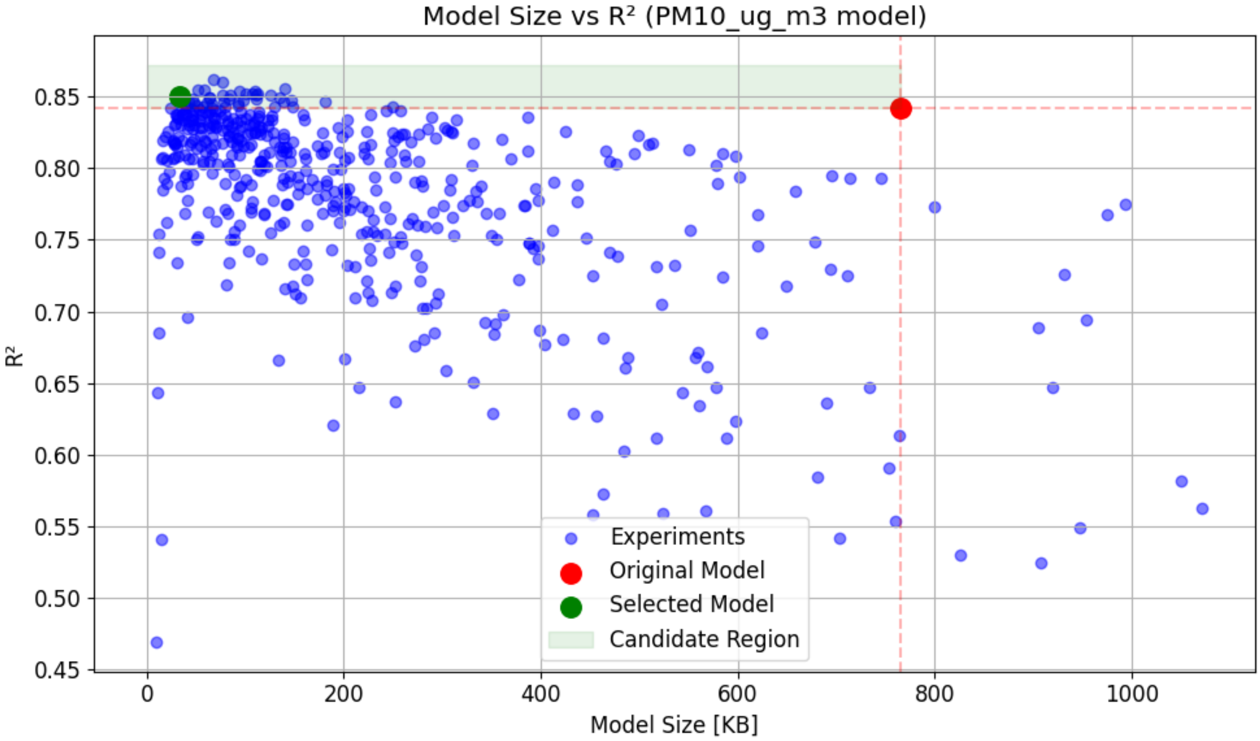


Figure 133: Visualization of the PM10 prediction model behaviour

Each blue dot is a test metric result (RMSE in this case) for a model trained with a set of hyperparameters. The green region contains models that are smaller and better than the original one. It can be sees that there are multiple candidates, however as per the pilot objective it has been chosen the smallest one that was, at least, as good as the original one.

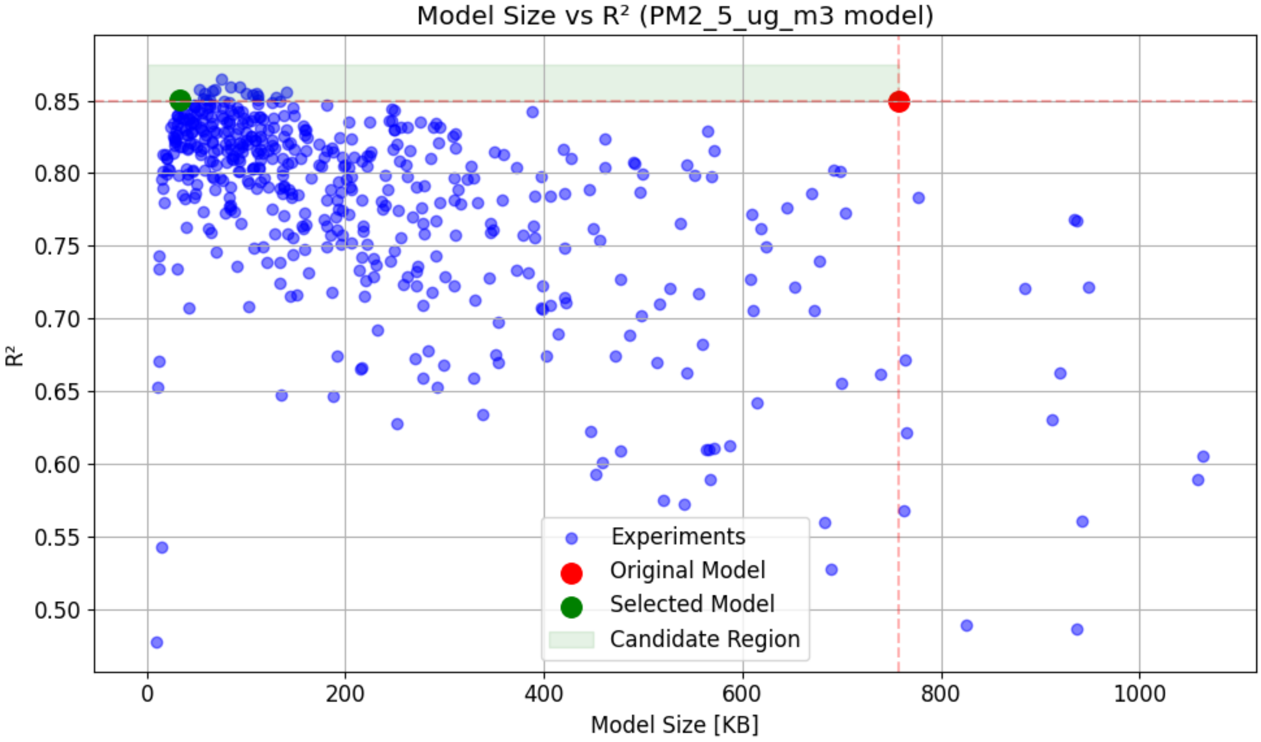


Figure 134: Visualization of the PM2.5 prediction model behaviour

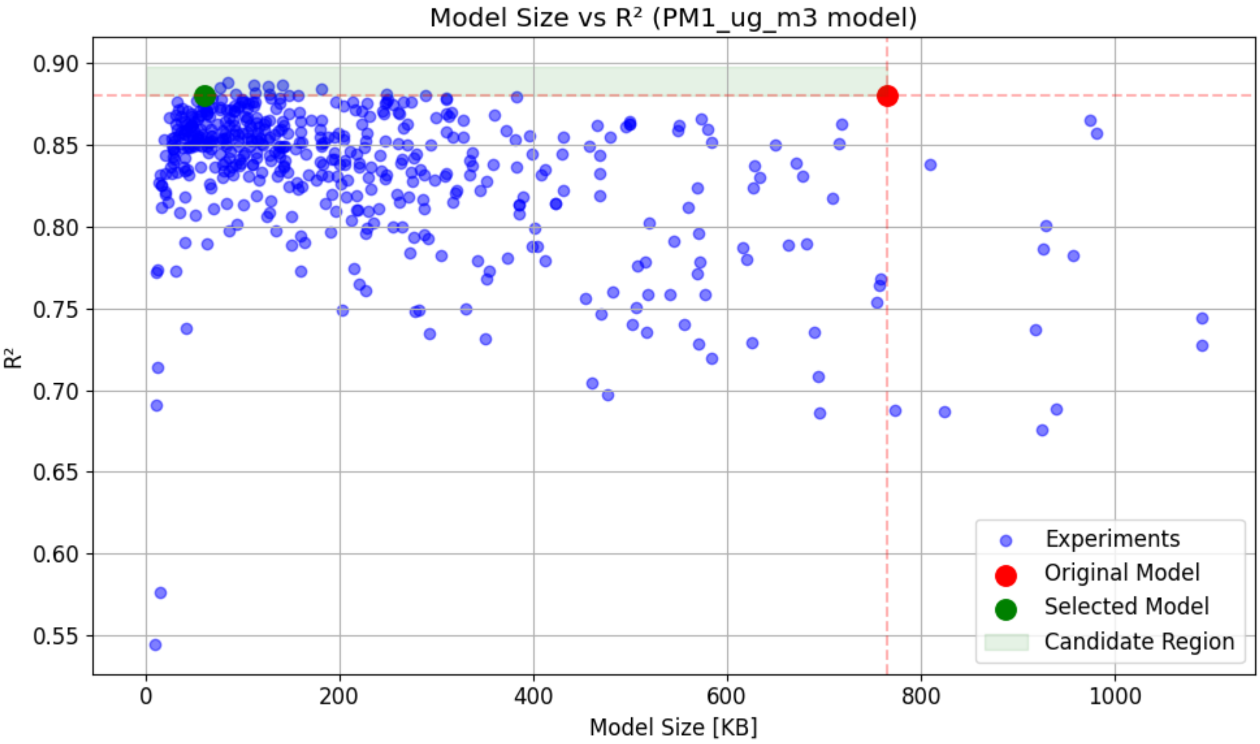


Figure 135: Visualization of the PM1 prediction model behaviour

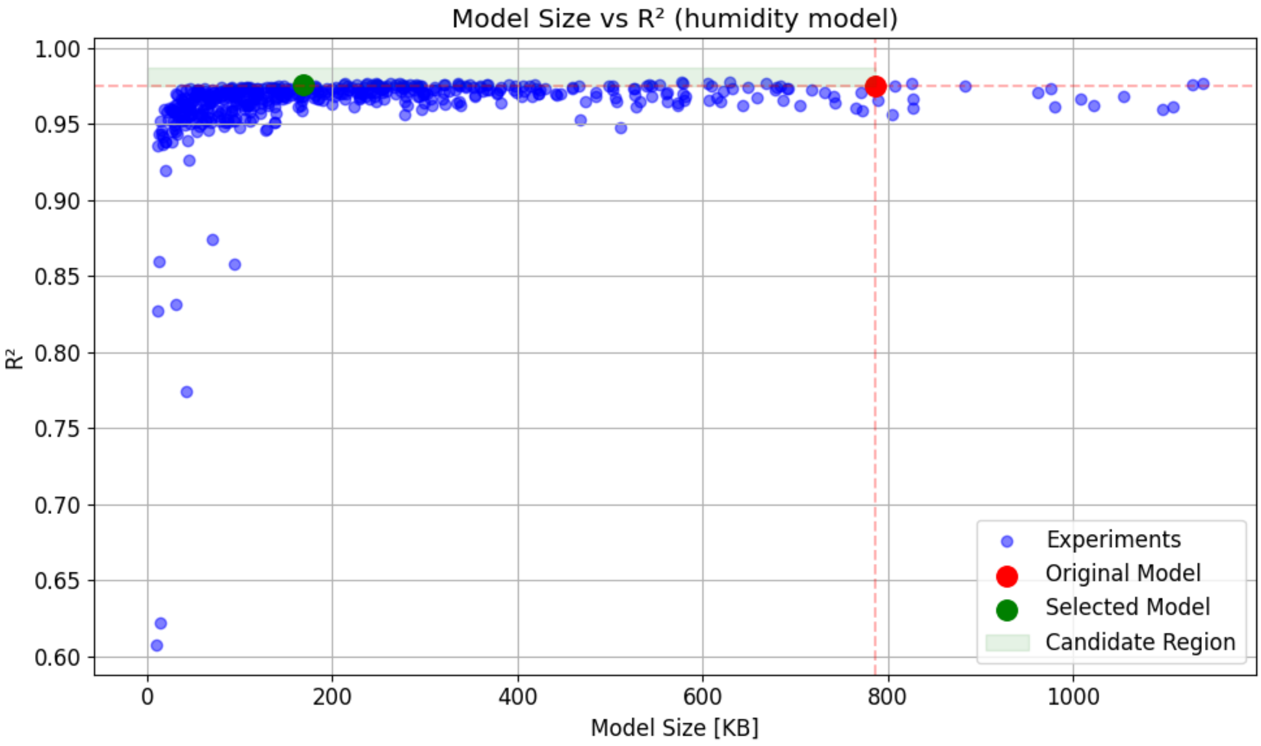


Figure 136: Visualization of the humidity prediction model behaviour

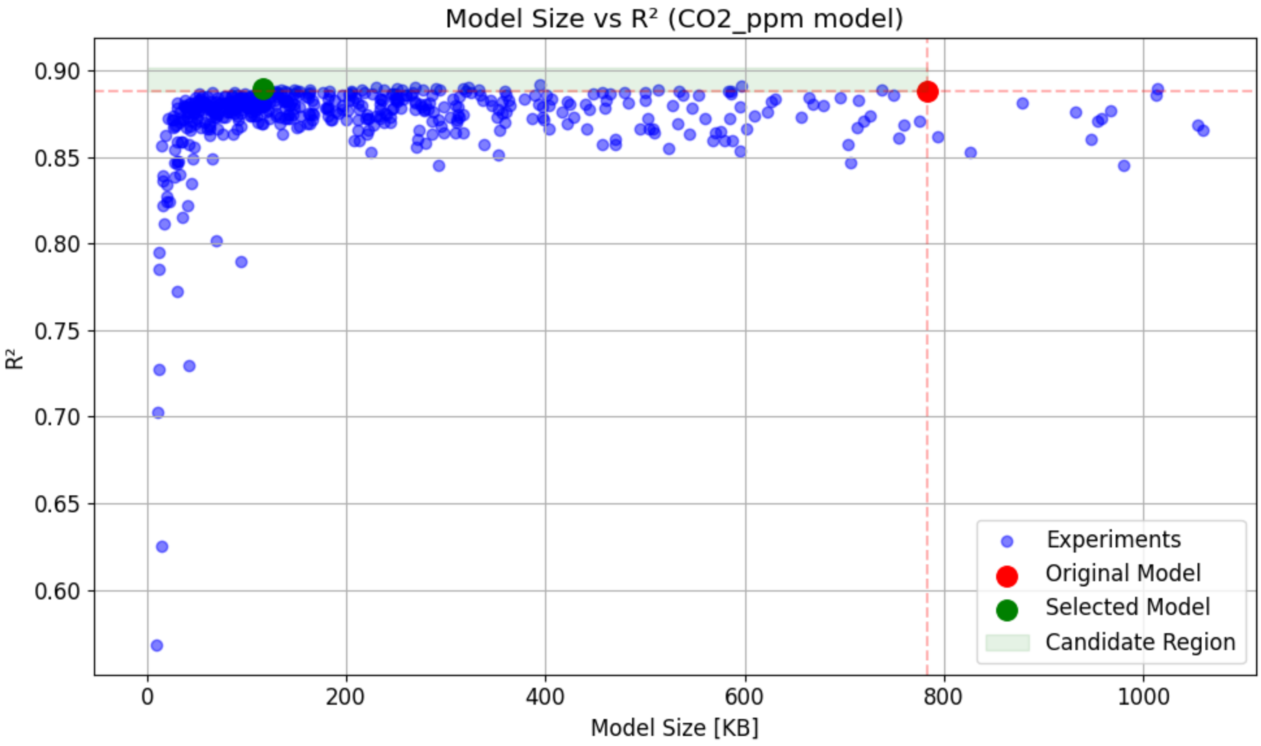


Figure 137: Visualization of CO2 prediction model behaviour

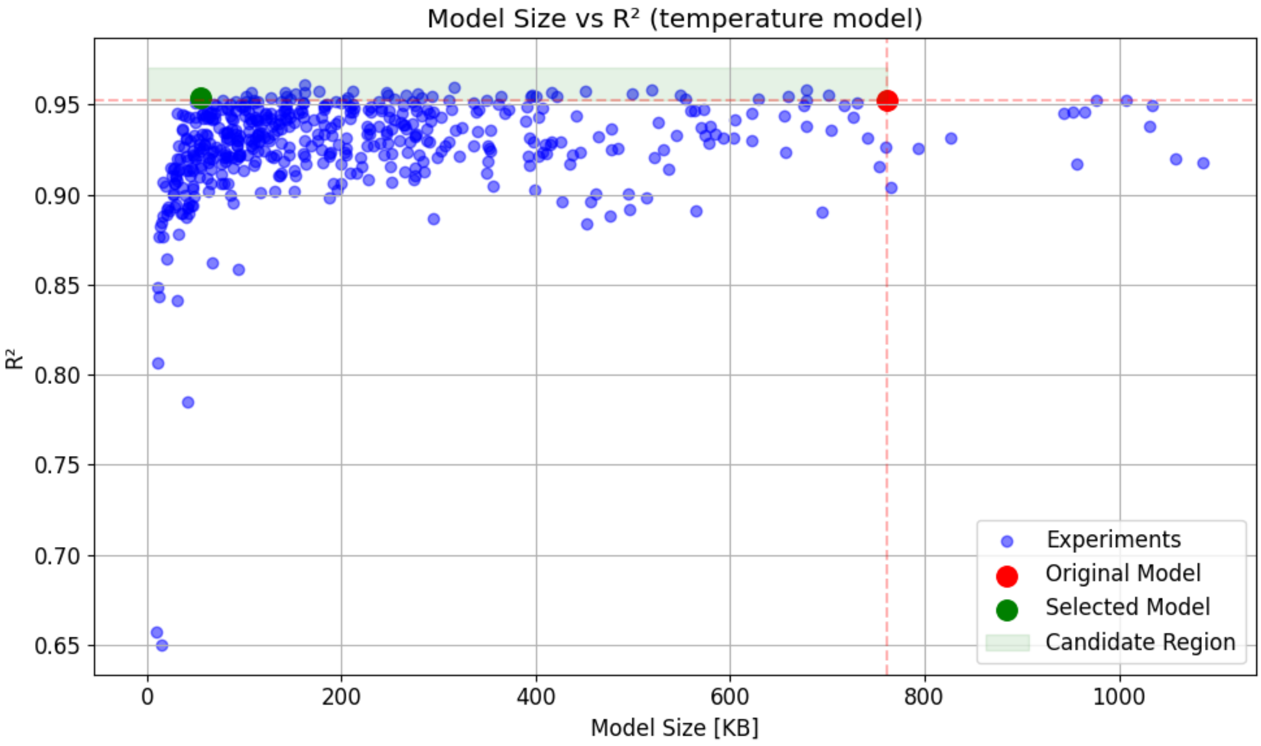


Figure 138: Visualization of the temperature prediction model behaviour

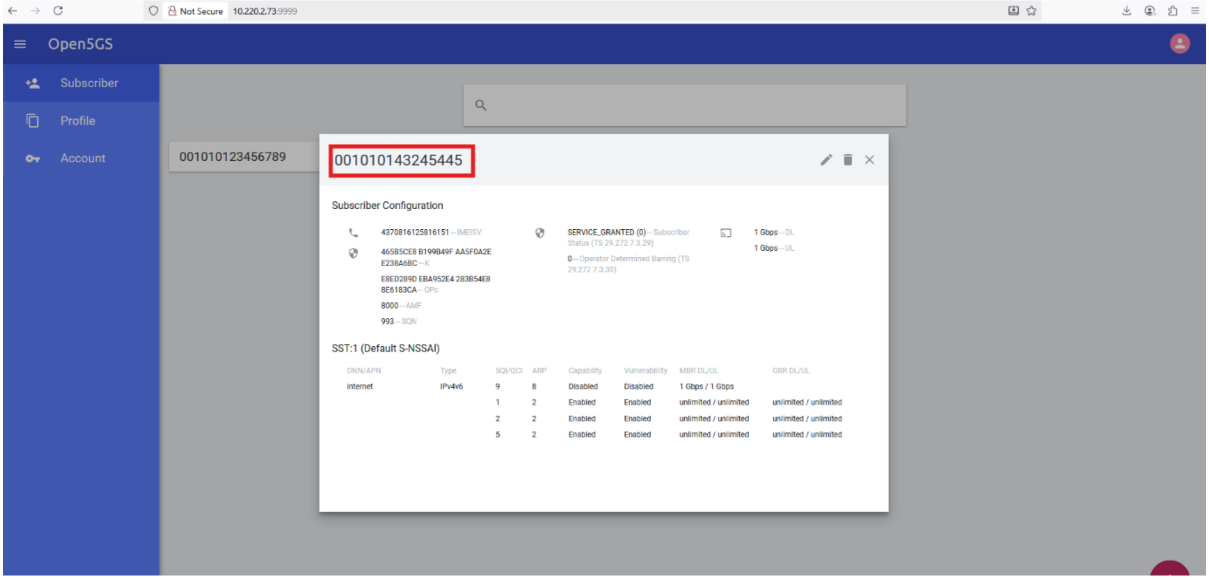
The code and the results from the experiments are in https://gitlab.aeros-project.eu/pilots/pilot-5/forecasting-health-energy/-/tree/dev/src/app/fai

Pilot 5 – Business Process 2 – Activity - 1 (P5-BP2-VA1): 5G E2E deployment validation with VNFs over aerOS (UERANSIM)

In D5.4, the full cycle for deploying Virtual Network Functions (VNFs) over aerOS was successfully executed and validated through interaction with UERANSIM, including the deployment of the User Plane Function (UPF) and the Network Exposure Function (NEF) for event monitoring and location reporting. This previous work demonstrated that aerOS could act as a MetaOS for orchestrating 5G E2E services across the continuum, managing both core network functions and exposure services at the edge.

To further validate the full functionality of aerOS as a continuum-supporting MetaOS, an additional validation activity was designed and executed. This activity focused on integrating a new NEF capability — AsSessionWithQoS — and deploying it through aerOS at the edge domain. The goal was to demonstrate not only that aerOS can manage the lifecycle of network functions, but also that newly introduced 5G features can be seamlessly integrated, exposed, and validated in real-world testbed conditions.

The validation started with the connection of the UERANSIM emulated device to the Open5GS core network. A UE was successfully registered, with signaling confirmed through both the UERANSIM console and the Open5GS graphical interface. At this stage, registration messages were observed at the AMF and associated core components, verifying that the emulated UE was fully integrated into the 5G system and available for subsequent QoS provisioning.



001010143245445

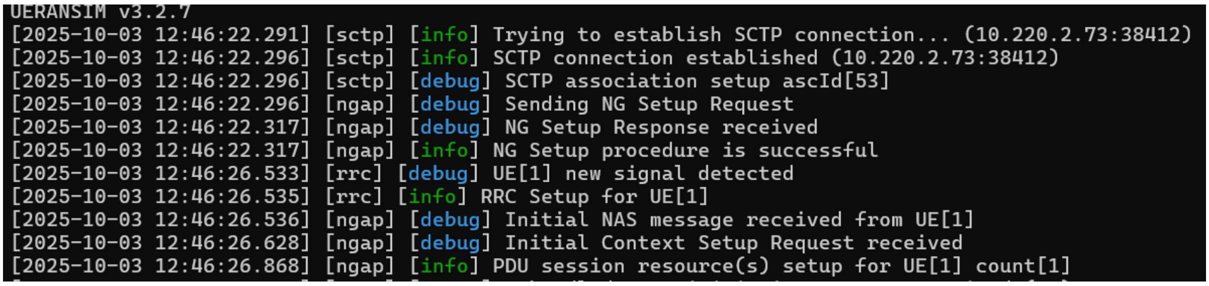
Subscriber Configuration

4370816125816151 --- IMEISV
465B5CE8 B199B49F AA5F0A2E E238A6BC --- K
E8ED289D EBA952E4 283B54E8 8E6183CA --- OPc
8000 --- AMF
993 --- SQN
SERVICE_GRANTED (0) --- Subscriber Status (TS 29.272 7.3.29)
0 --- Operator Determined Barring (TS 29.272 7.3.30)
1 Gbps --- DL
1 Gbps --- UL

SST:1 (Default S-NSSAI)

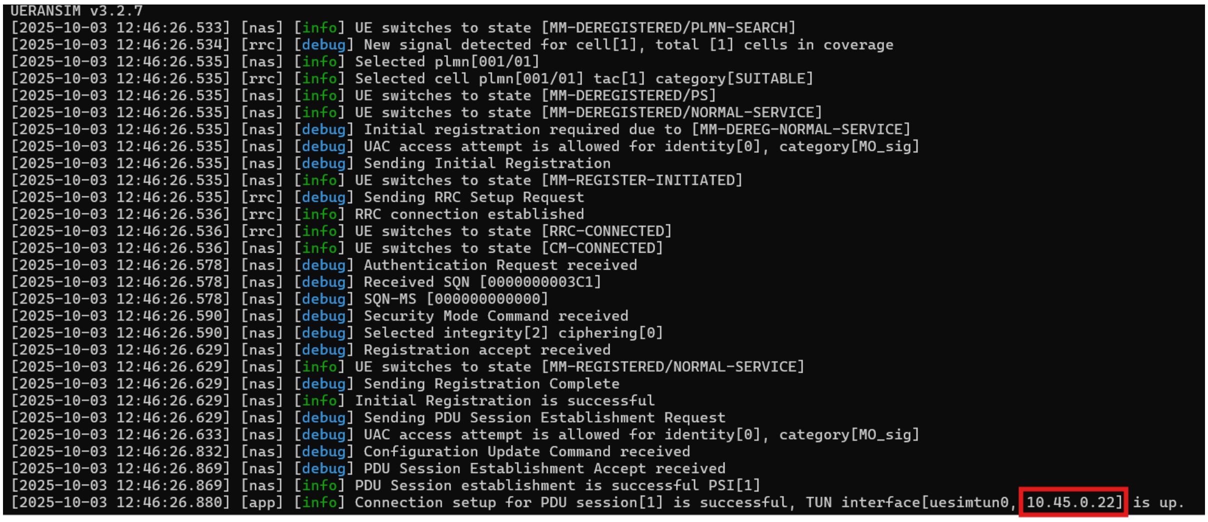
DNN/APN	Type	5QI/QCI	ARP	Capability	Vulnerability	MBR DL/UL	GBR DL/UL
internet	IPv4v6	9	8	Disabled	Disabled	1 Gbps / 1 Gbps	
		1	2	Enabled	Enabled	unlimited / unlimited	unlimited / unlimited
		2	2	Enabled	Enabled	unlimited / unlimited	unlimited / unlimited
		5	2	Enabled	Enabled	unlimited / unlimited	unlimited / unlimited

Figure 139: Registration of UE as a subscriber in the open5gs



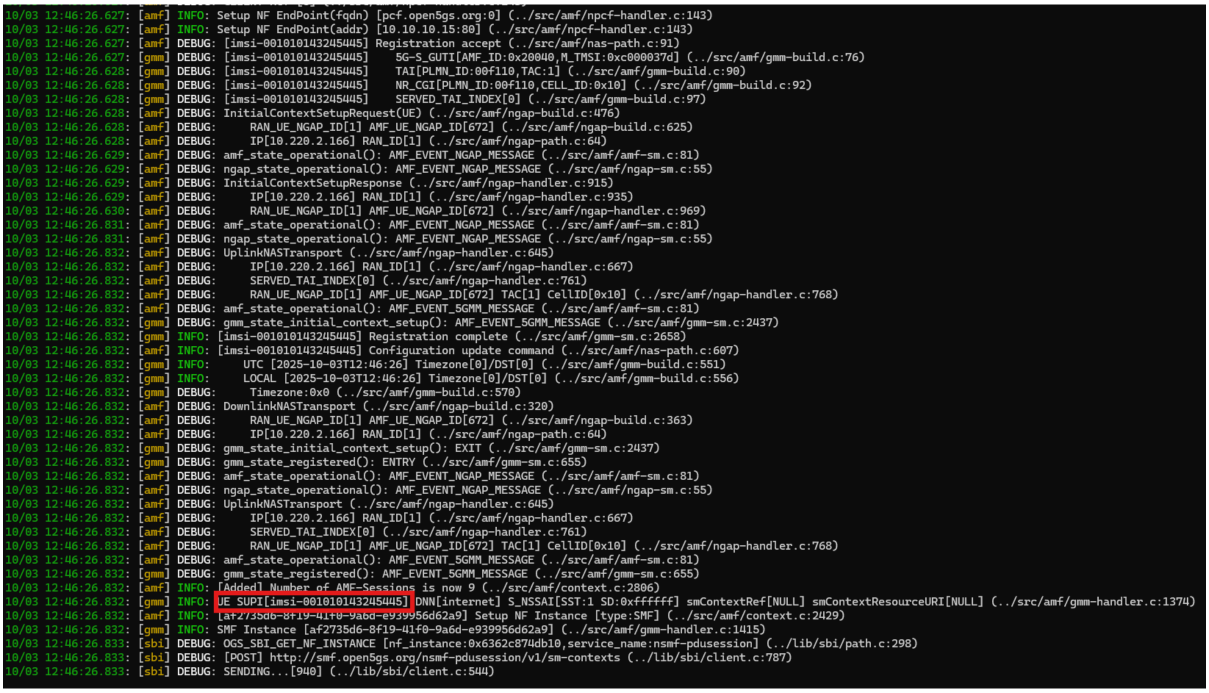
```
UERANSIM v3.2.7
[2025-10-03 12:46:22.291] [sctp] [info] Trying to establish SCTP connection... (10.220.2.73:38412)
[2025-10-03 12:46:22.296] [sctp] [info] SCTP connection established (10.220.2.73:38412)
[2025-10-03 12:46:22.296] [sctp] [debug] SCTP association setup ascId[53]
[2025-10-03 12:46:22.296] [ngap] [debug] Sending NG Setup Request
[2025-10-03 12:46:22.317] [ngap] [debug] NG Setup Response received
[2025-10-03 12:46:22.317] [ngap] [info] NG Setup procedure is successful
[2025-10-03 12:46:26.533] [rrc] [debug] UE[1] new signal detected
[2025-10-03 12:46:26.535] [rrc] [info] RRC Setup for UE[1]
[2025-10-03 12:46:26.536] [ngap] [debug] Initial NAS message received from UE[1]
[2025-10-03 12:46:26.628] [ngap] [debug] Initial Context Setup Request received
[2025-10-03 12:46:26.868] [ngap] [info] PDU session resource(s) setup for UE[1] count[1]
```

Figure 140: UERANSIM registered gNB to open5gs



```
UERANSIM v3.2.7
[2025-10-03 12:46:26.533] [nas] [info] UE switches to state [MM-DEREGISTERED/PLMN-SEARCH]
[2025-10-03 12:46:26.534] [rrc] [debug] New signal detected for cell[1], total [1] cells in coverage
[2025-10-03 12:46:26.535] [nas] [info] Selected plmn[001/01]
[2025-10-03 12:46:26.535] [rrc] [info] Selected cell plmn[001/01] tac[1] category[SUITABLE]
[2025-10-03 12:46:26.535] [nas] [info] UE switches to state [MM-DEREGISTERED/PS]
[2025-10-03 12:46:26.535] [nas] [info] UE switches to state [MM-DEREGISTERED/NORMAL-SERVICE]
[2025-10-03 12:46:26.535] [nas] [debug] Initial registration required due to [MM-DEREG-NORMAL-SERVICE]
[2025-10-03 12:46:26.535] [nas] [debug] UAC access attempt is allowed for identity[0], category[MO_sig]
[2025-10-03 12:46:26.535] [nas] [debug] Sending Initial Registration
[2025-10-03 12:46:26.535] [nas] [info] UE switches to state [MM-REGISTER-INITIATED]
[2025-10-03 12:46:26.535] [rrc] [debug] Sending RRC Setup Request
[2025-10-03 12:46:26.536] [rrc] [info] RRC connection established
[2025-10-03 12:46:26.536] [rrc] [info] UE switches to state [RRC-CONNECTED]
[2025-10-03 12:46:26.536] [nas] [info] UE switches to state [CM-CONNECTED]
[2025-10-03 12:46:26.578] [nas] [debug] Authentication Request received
[2025-10-03 12:46:26.578] [nas] [debug] Received SQN [0000000003C1]
[2025-10-03 12:46:26.578] [nas] [debug] SQN-MS [000000000000]
[2025-10-03 12:46:26.590] [nas] [debug] Security Mode Command received
[2025-10-03 12:46:26.590] [nas] [debug] Selected integrity[2] ciphering[0]
[2025-10-03 12:46:26.629] [nas] [debug] Registration accept received
[2025-10-03 12:46:26.629] [nas] [info] UE switches to state [MM-REGISTERED/NORMAL-SERVICE]
[2025-10-03 12:46:26.629] [nas] [debug] Sending Registration Complete
[2025-10-03 12:46:26.629] [nas] [info] Initial Registration is successful
[2025-10-03 12:46:26.629] [nas] [debug] Sending PDU Session Establishment Request
[2025-10-03 12:46:26.633] [nas] [debug] UAC access attempt is allowed for identity[0], category[MO_sig]
[2025-10-03 12:46:26.832] [nas] [debug] Configuration Update Command received
[2025-10-03 12:46:26.869] [nas] [debug] PDU Session Establishment Accept received
[2025-10-03 12:46:26.869] [nas] [info] PDU Session establishment is successful PSI[1]
[2025-10-03 12:46:26.880] [app] [info] Connection setup for PDU session[1] is successful, TUN interface[uesimtun0, 10.45.0.22] is up.
```

Figure 141: Registration of UE (UERANSIM provided) to open5gs



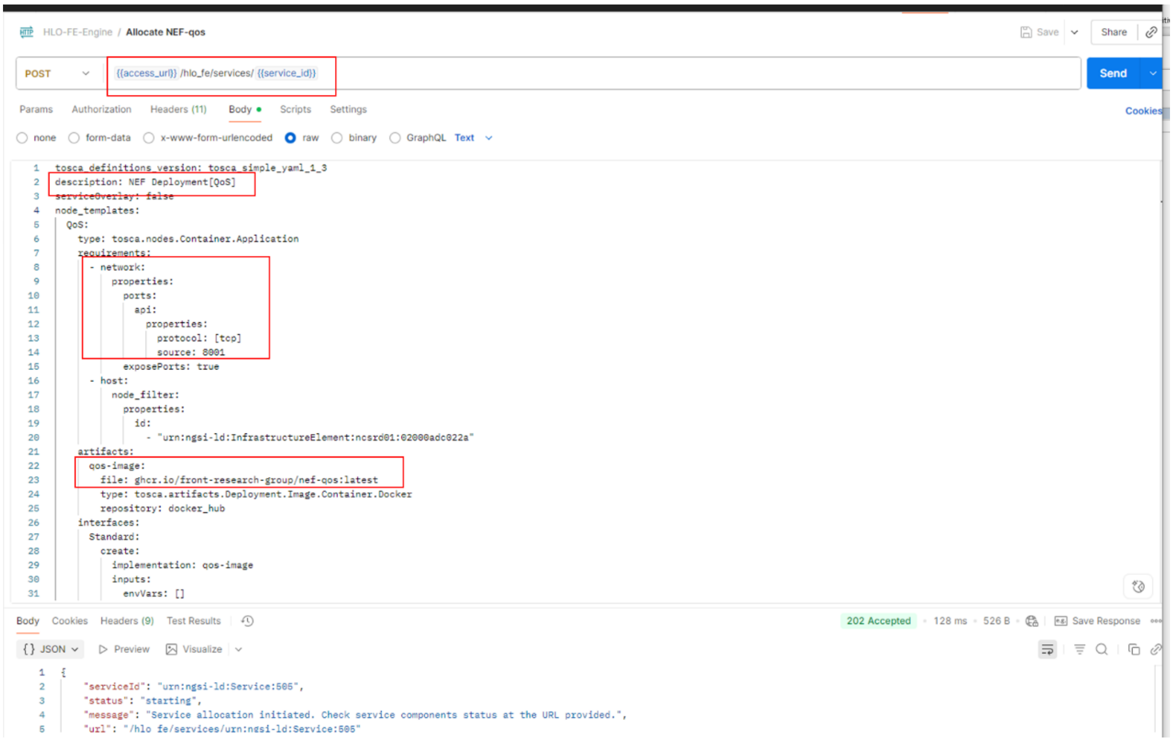
```
10/03 12:46:26.627: [amf] INFO: Setup NF EndPoint(fqdn) [pcf.open5gs.org:0] (../src/amf/npcf-handler.c:143)
10/03 12:46:26.627: [amf] INFO: Setup NF EndPoint(addr) [10.10.10.15:80] (../src/amf/npcf-handler.c:143)
10/03 12:46:26.627: [amf] DEBUG: [imsi-001010143245445] Registration accept (../src/amf/nas-path.c:91)
10/03 12:46:26.627: [gmm] DEBUG: [imsi-001010143245445]    5G-S_GUTI[AMF_ID:0x20040,M_TMSI:0xc000037d] (../src/amf/gmm-build.c:76)
10/03 12:46:26.628: [gmm] DEBUG: [imsi-001010143245445]    TAI[PLMN_ID:00f110,TAC:1] (../src/amf/gmm-build.c:90)
10/03 12:46:26.628: [gmm] DEBUG: [imsi-001010143245445]    NR_CGI[PLMN_ID:00f110,CELL_ID:0x10] (../src/amf/gmm-build.c:92)
10/03 12:46:26.628: [gmm] DEBUG: [imsi-001010143245445]    SERVED_TAI_INDEX[0] (../src/amf/gmm-build.c:97)
10/03 12:46:26.628: [amf] DEBUG: InitialContextSetupRequest(UE) (../src/amf/ngap-build.c:476)
10/03 12:46:26.628: [amf] DEBUG:     RAN_UE_NGAP_ID[1] AMF_UE_NGAP_ID[672] (../src/amf/ngap-build.c:625)
10/03 12:46:26.628: [amf] DEBUG:     IP[10.220.2.166] RAN_ID[1] (../src/amf/ngap-path.c:64)
10/03 12:46:26.629: [amf] DEBUG: amf_state_operational(): AMF_EVENT_NGAP_MESSAGE (../src/amf/amf-sm.c:81)
10/03 12:46:26.629: [amf] DEBUG: ngap_state_operational(): AMF_EVENT_NGAP_MESSAGE (../src/amf/ngap-sm.c:55)
10/03 12:46:26.629: [amf] DEBUG: InitialContextSetupResponse (../src/amf/ngap-handler.c:915)
10/03 12:46:26.629: [amf] DEBUG:     IP[10.220.2.166] RAN_ID[1] (../src/amf/ngap-handler.c:935)
10/03 12:46:26.630: [amf] DEBUG:     RAN_UE_NGAP_ID[1] AMF_UE_NGAP_ID[672] (../src/amf/ngap-handler.c:969)
10/03 12:46:26.831: [amf] DEBUG: amf_state_operational(): AMF_EVENT_NGAP_MESSAGE (../src/amf/amf-sm.c:81)
10/03 12:46:26.831: [amf] DEBUG: ngap_state_operational(): AMF_EVENT_NGAP_MESSAGE (../src/amf/ngap-sm.c:55)
10/03 12:46:26.832: [amf] DEBUG: UplinkNASTransport (../src/amf/ngap-handler.c:645)
10/03 12:46:26.832: [amf] DEBUG:     IP[10.220.2.166] RAN_ID[1] (../src/amf/ngap-handler.c:667)
10/03 12:46:26.832: [amf] DEBUG:     SERVED_TAI_INDEX[0] (../src/amf/ngap-handler.c:761)
10/03 12:46:26.832: [amf] DEBUG:     RAN_UE_NGAP_ID[1] AMF_UE_NGAP_ID[672] TAC[1] CellID[0x10] (../src/amf/ngap-handler.c:768)
10/03 12:46:26.832: [amf] DEBUG: amf_state_operational(): AMF_EVENT_5GMM_MESSAGE (../src/amf/amf-sm.c:81)
10/03 12:46:26.832: [gmm] DEBUG: gmm_state_initial_context_setup(): AMF_EVENT_5GMM_MESSAGE (../src/amf/gmm-sm.c:2437)
10/03 12:46:26.832: [gmm] INFO: [imsi-001010143245445] Registration complete (../src/amf/gmm-sm.c:2658)
10/03 12:46:26.832: [amf] INFO: [imsi-001010143245445] Configuration update command (../src/amf/nas-path.c:607)
10/03 12:46:26.832: [gmm] INFO:     UTC [2025-10-03T12:46:26] Timezone[0]/DST[0] (../src/amf/gmm-build.c:551)
10/03 12:46:26.832: [gmm] INFO:     LOCAL [2025-10-03T12:46:26] Timezone[0]/DST[0] (../src/amf/gmm-build.c:556)
10/03 12:46:26.832: [gmm] DEBUG:     Timezone:0x0 (../src/amf/gmm-build.c:570)
10/03 12:46:26.832: [amf] DEBUG: DownlinkNASTransport (../src/amf/ngap-build.c:320)
10/03 12:46:26.832: [amf] DEBUG:     RAN_UE_NGAP_ID[1] AMF_UE_NGAP_ID[672] (../src/amf/ngap-build.c:363)
10/03 12:46:26.832: [amf] DEBUG:     IP[10.220.2.166] RAN_ID[1] (../src/amf/ngap-path.c:64)
10/03 12:46:26.832: [gmm] DEBUG: gmm_state_initial_context_setup(): EXIT (../src/amf/gmm-sm.c:2437)
10/03 12:46:26.832: [gmm] DEBUG: gmm_state_registered(): ENTRY (../src/amf/gmm-sm.c:655)
10/03 12:46:26.832: [amf] DEBUG: amf_state_operational(): AMF_EVENT_NGAP_MESSAGE (../src/amf/amf-sm.c:81)
10/03 12:46:26.832: [amf] DEBUG: ngap_state_operational(): AMF_EVENT_NGAP_MESSAGE (../src/amf/ngap-sm.c:55)
10/03 12:46:26.832: [amf] DEBUG: UplinkNASTransport (../src/amf/ngap-handler.c:645)
10/03 12:46:26.832: [amf] DEBUG:     IP[10.220.2.166] RAN_ID[1] (../src/amf/ngap-handler.c:667)
10/03 12:46:26.832: [amf] DEBUG:     SERVED_TAI_INDEX[0] (../src/amf/ngap-handler.c:761)
10/03 12:46:26.832: [amf] DEBUG:     RAN_UE_NGAP_ID[1] AMF_UE_NGAP_ID[672] TAC[1] CellID[0x10] (../src/amf/ngap-handler.c:768)
10/03 12:46:26.832: [amf] DEBUG: amf_state_operational(): AMF_EVENT_5GMM_MESSAGE (../src/amf/amf-sm.c:81)
10/03 12:46:26.832: [gmm] DEBUG: gmm_state_registered(): AMF_EVENT_5GMM_MESSAGE (../src/amf/gmm-sm.c:655)
10/03 12:46:26.832: [amf] INFO: [Added] Number of AMF-Sessions is now 9 (../src/amf/context.c:2806)
10/03 12:46:26.832: [gmm] INFO: UE SUPI[imsi-001010143245445] DNN[internet] S_NSSAI[SST:1 SD:0xffffff] smContextRef[NULL] smContextResourceURI[NULL] (../src/amf/gmm-handler.c:1374)
10/03 12:46:26.832: [amf] INFO: [af2735d6-8f19-41f0-9a6d-e939956d62a9] Setup NF Instance [type:SMF] (../src/amf/context.c:2429)
10/03 12:46:26.832: [gmm] INFO: SMF Instance [af2735d6-8f19-41f0-9a6d-e939956d62a9] (../src/amf/gmm-handler.c:1415)
10/03 12:46:26.833: [sbi] DEBUG: OGS_SBI_GET_NF_INSTANCE [nf_instance:0x6362c874db10,service_name:nsmf-pdusession] (../lib/sbi/path.c:298)
10/03 12:46:26.833: [sbi] DEBUG: [POST] http://smf.open5gs.org/nsmf-pdusession/v1/sm-contexts (../lib/sbi/client.c:787)
10/03 12:46:26.833: [sbi] DEBUG: SENDING...[940] (../lib/sbi/client.c:544)
```

Figure 142: AMF (open5gs) logs with messages during UE (UERANSIM) registration in the 5G network

Following UE registration, the AsSessionWithQoS component was deployed as a new VNF through the aerOS orchestration layer. Deployment was performed using a TOSCA descriptor, ensuring declarative and automated lifecycle management.

The aerOS deployment logs confirmed the instantiation of the NEF-QoS VNF, while k9s inspection showed the relevant pods and services correctly running in the edge cluster. Furthermore, the exposed Swagger interface of the QoS API verified that the functionality was available to external consumers.

This step demonstrated the ability of aerOS to dynamically instantiate new VNFs at the edge, expose their APIs, and make them ready for interaction with the 5G core.

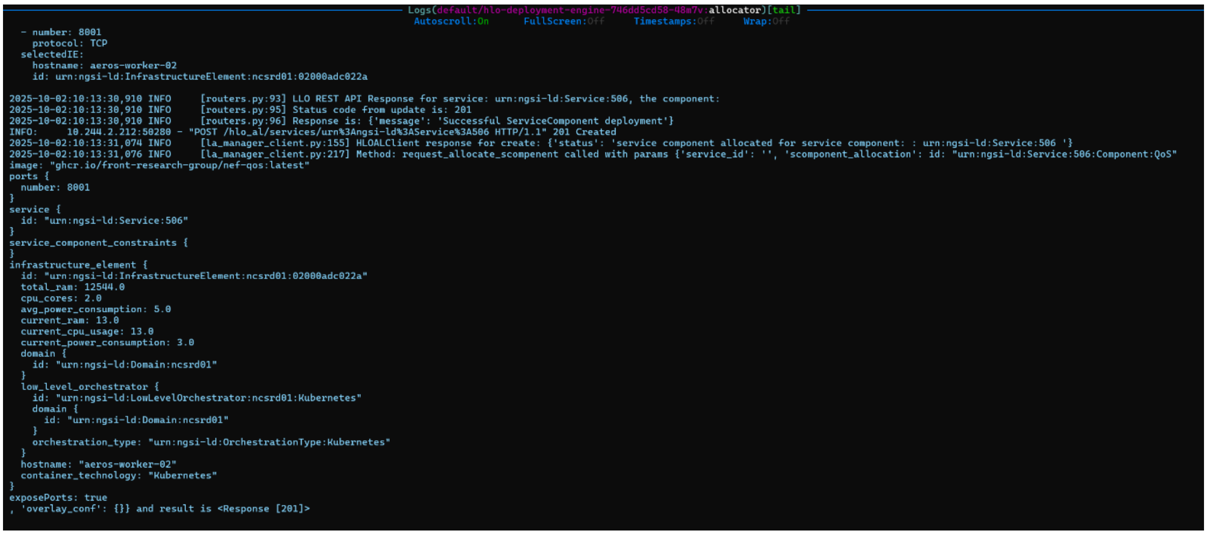


POST {{access_url}} /hlo_fe/services/ {{service_id}}

```
tosca_definitions_version: tosca_simple_yaml_1_3
description: NEF Deployment[QoS]
serviceOverlay: false
node_templates:
  QoS:
    type: tosca.nodes.Container.Application
    requirements:
      - network:
          properties:
            ports:
              api:
                properties:
                  protocol: [tcp]
                  source: 8001
            exposePorts: true
      - host:
          node_filter:
            properties:
              id:
                - "urn:ngsi-ld:InfrastructureElement:ncsrd01:02000adc022a"
    artifacts:
      qos-image:
        file: ghcr.io/front-research-group/nef-qos:latest
        type: tosca.artifacts.Deployment.Image.Container.Docker
        repository: docker_hub
    interfaces:
      Standard:
        create:
          implementation: qos-image
          inputs:
            envVars: []
```

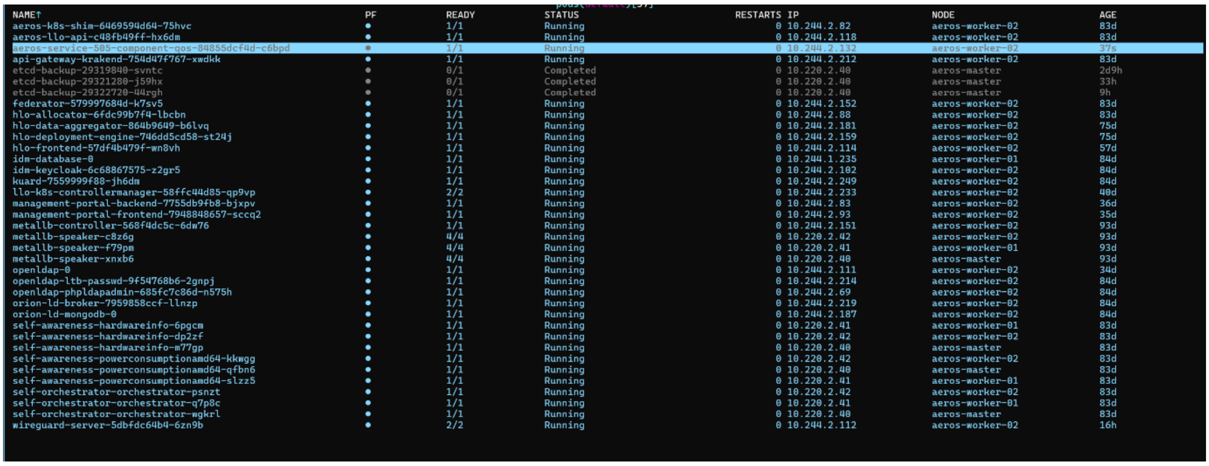
```
{
    "serviceId": "urn:ngsi-ld:Service:505",
    "status": "starting",
    "message": "Service allocation initiated. Check service components status at the URL provided.",
    "url": "/hlo fe/services/urn:ngsi-ld:Service:505"
```

Figure 143: aerOS descriptor (TOSCA based) for QoS service deployment



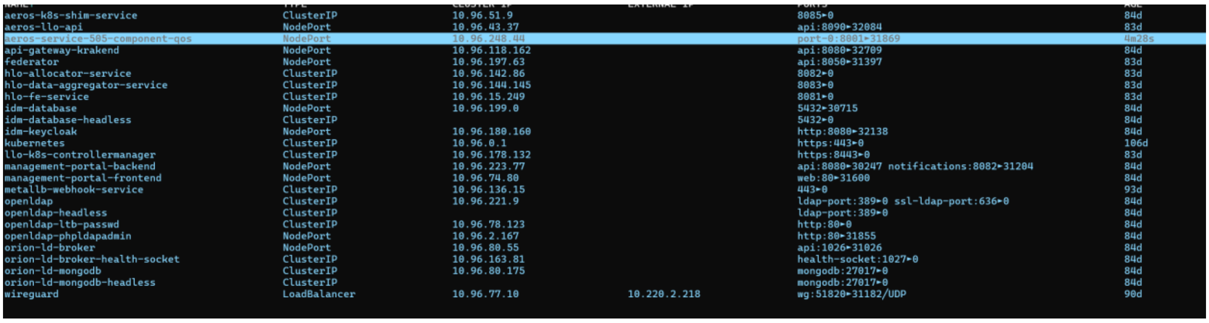
```
                                        Logs(default/hlo-deployment-engine-746dd5cd58-48m7v:allocator)[tail]
                                         Autoscroll:On      FullScreen:Off      Timestamps:Off      Wrap:Off
  - number: 8001
    protocol: TCP
  selectedIE:
    hostname: aeros-worker-02
    id: urn:ngsi-ld:InfrastructureElement:ncsrd01:02000adc022a

2025-10-02:10:13:30,910 INFO     [routers.py:93] LLO REST API Response for service: urn:ngsi-ld:Service:506, the component:
2025-10-02:10:13:30,910 INFO     [routers.py:95] Status code from update is: 201
2025-10-02:10:13:30,910 INFO     [routers.py:96] Response is: {'message': 'Successful ServiceComponent deployment'}
INFO:     10.244.2.212:50280 - "POST /hlo_al/services/urn%3Angsi-ld%3AService%3A506 HTTP/1.1" 201 Created
2025-10-02:10:13:31,074 INFO     [la_manager_client.py:155] HLOALClient response for create: {'status': 'service component allocated for service component: : urn:ngsi-ld:Service:506 '}
2025-10-02:10:13:31,076 INFO     [la_manager_client.py:217] Method: request_allocate_scompenent called with params {'service_id': '', 'scomponent_allocation': id: "urn:ngsi-ld:Service:506:Component:QoS"
image: "ghcr.io/front-research-group/nef-qos:latest"
ports {
  number: 8001
}
service {
  id: "urn:ngsi-ld:Service:506"
}
service_component_constraints {
}
infrastructure_element {
  id: "urn:ngsi-ld:InfrastructureElement:ncsrd01:02000adc022a"
  total_ram: 12544.0
  cpu_cores: 2.0
  avg_power_consumption: 5.0
  current_ram: 13.0
  current_cpu_usage: 13.0
  current_power_consumption: 3.0
  domain {
    id: "urn:ngsi-ld:Domain:ncsrd01"
  }
  low_level_orchestrator {
    id: "urn:ngsi-ld:LowLevelOrchestrator:ncsrd01:Kubernetes"
    domain {
      id: "urn:ngsi-ld:Domain:ncsrd01"
    }
    orchestration_type: "urn:ngsi-ld:OrchestrationType:Kubernetes"
  }
  hostname: "aeros-worker-02"
  container_technology: "Kubernetes"
}
exposePorts: true
, 'overlay_conf': {}} and result is <Response [201]>
```

Figure 144: aerOS HLO proceeding to service deployment



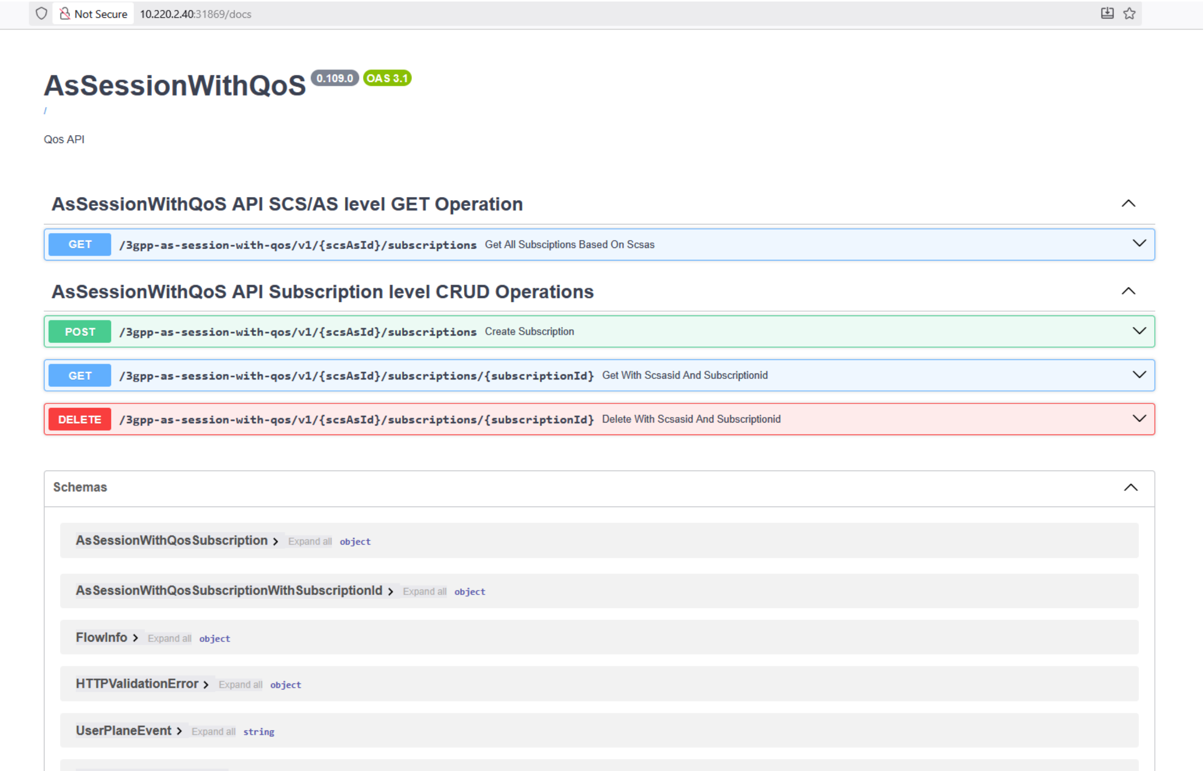
```
NAME↑                                         PF      READY     STATUS                 RESTARTS IP               NODE                AGE
aeros-k8s-shim-6469594d64-75hvc               ●       1/1       Running                       0 10.244.2.82      aeros-worker-02     83d
aeros-llo-api-c48fb49ff-hx6dm                 ●       1/1       Running                       0 10.244.2.118     aeros-worker-02     83d
aeros-service-505-component-qos-84855dcf4d-c6bpd ●    1/1       Running                       0 10.244.2.132     aeros-worker-02     37s
api-gateway-krakend-754d47f767-xwdkk          ●       1/1       Running                       0 10.244.2.212     aeros-worker-02     83d
etcd-backup-29319840-svntc                    ●       0/1       Completed                     0 10.220.2.40      aeros-master        2d9h
etcd-backup-29321280-j59hx                    ●       0/1       Completed                     0 10.220.2.40      aeros-master        33h
etcd-backup-29322720-44rgh                    ●       0/1       Completed                     0 10.220.2.40      aeros-master        9h
federator-579997684d-k7sv5                    ●       1/1       Running                       0 10.244.2.152     aeros-worker-02     83d
hlo-allocator-6fdc99b7f4-lbcbn                ●       1/1       Running                       0 10.244.2.88      aeros-worker-02     83d
hlo-data-aggregator-864b9649-b6lvq            ●       1/1       Running                       0 10.244.2.181     aeros-worker-02     75d
hlo-deployment-engine-746dd5cd58-st24j        ●       1/1       Running                       0 10.244.2.159     aeros-worker-02     75d
hlo-frontend-57df4b479f-wn8vh                 ●       1/1       Running                       0 10.244.2.114     aeros-worker-02     57d
idm-database-0                                ●       1/1       Running                       0 10.244.1.235     aeros-worker-01     84d
idm-keycloak-6c68867575-z2gr5                 ●       1/1       Running                       0 10.244.2.102     aeros-worker-02     84d
kuard-7559999f88-jh6dm                        ●       1/1       Running                       0 10.244.2.249     aeros-worker-02     84d
llo-k8s-controllermanager-58ffc44d85-qp9vp    ●       2/2       Running                       0 10.244.2.233     aeros-worker-02     40d
management-portal-backend-7755db9fb8-bjxpv    ●       1/1       Running                       0 10.244.2.83      aeros-worker-02     36d
management-portal-frontend-7948848657-sccq2   ●       1/1       Running                       0 10.244.2.93      aeros-worker-02     35d
metallb-controller-568f4dc5c-6dw76            ●       1/1       Running                       0 10.244.2.151     aeros-worker-02     93d
metallb-speaker-c8z6g                         ●       4/4       Running                       0 10.220.2.42      aeros-worker-02     93d
metallb-speaker-f79pm                         ●       4/4       Running                       0 10.220.2.41      aeros-worker-01     93d
metallb-speaker-xnxb6                         ●       4/4       Running                       0 10.220.2.40      aeros-master        93d
openldap-0                                    ●       1/1       Running                       0 10.244.2.111     aeros-worker-02     34d
openldap-ltb-passwd-9f54768b6-2gnpj           ●       1/1       Running                       0 10.244.2.214     aeros-worker-02     84d
openldap-phpldapadmin-685fc7c86d-n575h        ●       1/1       Running                       0 10.244.2.69      aeros-worker-02     84d
orion-ld-broker-7959858ccf-llnzp              ●       1/1       Running                       0 10.244.2.219     aeros-worker-02     84d
orion-ld-mongodb-0                            ●       1/1       Running                       0 10.244.2.187     aeros-worker-02     84d
self-awareness-hardwareinfo-6pgcm             ●       1/1       Running                       0 10.220.2.41      aeros-worker-01     83d
self-awareness-hardwareinfo-dp2zf             ●       1/1       Running                       0 10.220.2.42      aeros-worker-02     83d
self-awareness-hardwareinfo-m77gp             ●       1/1       Running                       0 10.220.2.40      aeros-master        83d
self-awareness-powerconsumptionamd64-kkwgg    ●       1/1       Running                       0 10.220.2.42      aeros-worker-02     83d
self-awareness-powerconsumptionamd64-qfbn6    ●       1/1       Running                       0 10.220.2.40      aeros-master        83d
self-awareness-powerconsumptionamd64-slzz5    ●       1/1       Running                       0 10.220.2.41      aeros-worker-01     83d
self-orchestrator-orchestrator-psnzt          ●       1/1       Running                       0 10.220.2.42      aeros-worker-02     83d
self-orchestrator-orchestrator-q7p8c          ●       1/1       Running                       0 10.220.2.41      aeros-worker-01     83d
self-orchestrator-orchestrator-wgkrl          ●       1/1       Running                       0 10.220.2.40      aeros-master        83d
wireguard-server-5dbfdc64b4-6zn9b             ●       2/2       Running                       0 10.244.2.112     aeros-worker-02     16h
```

Figure 145: QoS application (pod in aerOS domain)



```
aeros-k8s-shim-service              ClusterIP      10.96.51.9                        8085►0                                        84d
aeros-llo-api                       NodePort       10.96.43.37                       api:8090►32084                                83d
aeros-service-505-component-qos     NodePort       10.96.248.44                      port-0:8001►31869                             4m28s
api-gateway-krakend                 NodePort       10.96.118.162                     api:8080►32709                                84d
federator                           NodePort       10.96.197.63                      api:8050►31397                                83d
hlo-allocator-service               ClusterIP      10.96.142.86                      8082►0                                        83d
hlo-data-aggregator-service         ClusterIP      10.96.144.145                     8083►0                                        83d
hlo-fe-service                      ClusterIP      10.96.15.249                      8081►0                                        83d
idm-database                        NodePort       10.96.199.0                       5432►30715                                    84d
idm-database-headless               ClusterIP                                        5432►0                                        84d
idm-keycloak                        NodePort       10.96.180.160                     http:8080►32138                               84d
kubernetes                          ClusterIP      10.96.0.1                         https:443►0                                   106d
llo-k8s-controllermanager           ClusterIP      10.96.178.132                     https:8443►0                                  83d
management-portal-backend           NodePort       10.96.223.77                      api:8080►30247 notifications:8082►31204       84d
management-portal-frontend          NodePort       10.96.74.80                       web:80►31600                                  84d
metallb-webhook-service             ClusterIP      10.96.136.15                      443►0                                         93d
openldap                            ClusterIP      10.96.221.9                       ldap-port:389►0 ssl-ldap-port:636►0           84d
openldap-headless                   ClusterIP                                        ldap-port:389►0                               84d
openldap-ltb-passwd                 ClusterIP      10.96.78.123                      http:80►0                                     84d
openldap-phpldapadmin               NodePort       10.96.2.167                       http:80►31855                                 84d
orion-ld-broker                     NodePort       10.96.80.55                       api:1026►31026                                84d
orion-ld-broker-health-socket       ClusterIP      10.96.163.81                      health-socket:1027►0                          84d
orion-ld-mongodb                    ClusterIP      10.96.80.175                      mongodb:27017►0                               84d
orion-ld-mongodb-headless           ClusterIP                                        mongodb:27017►0                               84d
wireguard                           LoadBalancer   10.96.77.10      10.220.2.218     wg:51820►31182/UDP                            90d
```

Figure 146: QoS service (service in aerOS domain) exposing network port as requested



Not Secure 10.220.2.40:31869/docs

AsSessionWithQoS 0.109.0 OAS 3.1

/

Qos API

AsSessionWithQoS API SCS/AS level GET Operation

GET /3gpp-as-session-with-qos/v1/{scsAsId}/subscriptions Get All Subsciptions Based On Scsas

AsSessionWithQoS API Subscription level CRUD Operations

POST /3gpp-as-session-with-qos/v1/{scsAsId}/subscriptions Create Subscription

GET /3gpp-as-session-with-qos/v1/{scsAsId}/subscriptions/{subscriptionId} Get With Scsasid And Subscriptionid

DELETE /3gpp-as-session-with-qos/v1/{scsAsId}/subscriptions/{subscriptionId} Delete With Scsasid And Subscriptionid

Schemas

AsSessionWithQosSubscription > Expand all object

AsSessionWithQosSubscriptionWithSubscriptionId > Expand all object

FlowInfo > Expand all object

HTTPValidationError > Expand all object

UserPlaneEvent > Expand all string

Figure 147: QoS service API exposed over aerOS

The final step consisted of calling the QoS API exposed by aerOS (via NodePort). Using the API, a 5QI value was registered for the UE that had been previously connected through UERANSIM.

API responses confirmed the acceptance of the request, and the logs of the QoS API showed the processing of the session registration. In parallel, PCF logs confirmed that the QoS policy event was propagated correctly to the 5G core. This provided end-to-end evidence that the newly deployed NEF-QoS component interacted successfully with both aerOS orchestration and the 5G core functions.

Curl

```
curl -X 'POST' \
  'http://10.220.2.73:8585/3gpp-as-session-with-qos/v1/example1/subscriptions' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "flowInfo": [
    {
      "flowDescriptions": [
        "permit in ip from 10.45.0.22 to any",
        "permit out ip from any to 10.45.0.22"
      ],
      "flowId": 1
    }
  ],
  "notificationDestination": "https://example.com/callback",
  "qosReference": "qod_2",
  "supportedFeatures": "003C",
  "ueIpv4Addr": "10.45.0.22"
}
```

Request URL

```
http://10.220.2.73:8585/3gpp-as-session-with-qos/v1/example1/subscriptions
```

Server response

Code Details

201

Response body

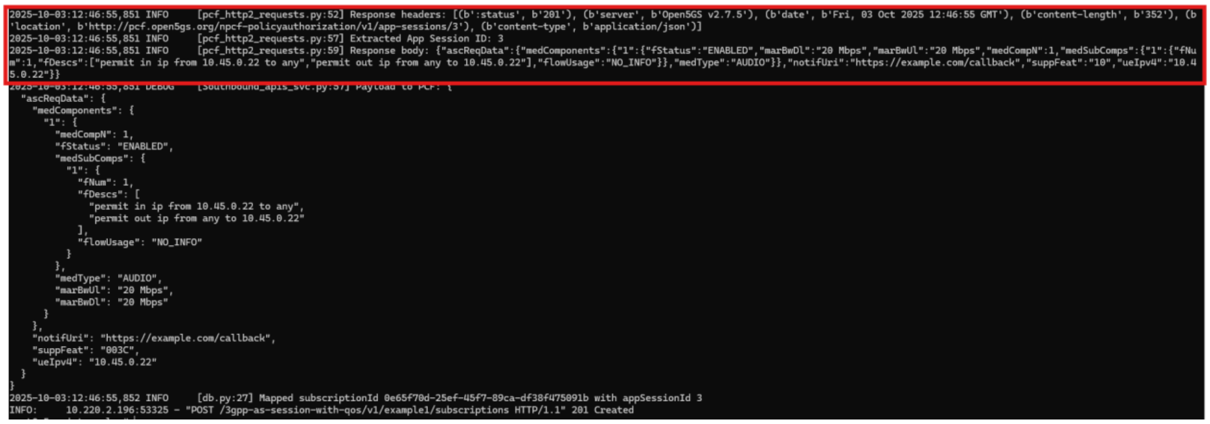
```
{
  "notificationDestination": "https://example.com/callback",
  "supportedFeatures": "003C",
  "qosReference": "qod_2",
  "ueIpv4Addr": "10.45.0.22",
  "flowInfo": [
    {
      "flowId": 1,
      "flowDescriptions": [
        "permit in ip from 10.45.0.22 to any",
        "permit out ip from any to 10.45.0.22"
      ]
    }
  ],
  "subscriptionId": "0e65f70d-25ef-45f7-89ca-df38f475091b"
}
```

Response headers

```
content-length: 314
content-type: application/json
date: Fri,03 Oct 2025 12:46:54 GMT
location: /3gpp-as-session-with-qos/v1/example1/subscriptions/0e65f70d-25ef-45f7-89ca-df38f475091b
server: uvicorn
```

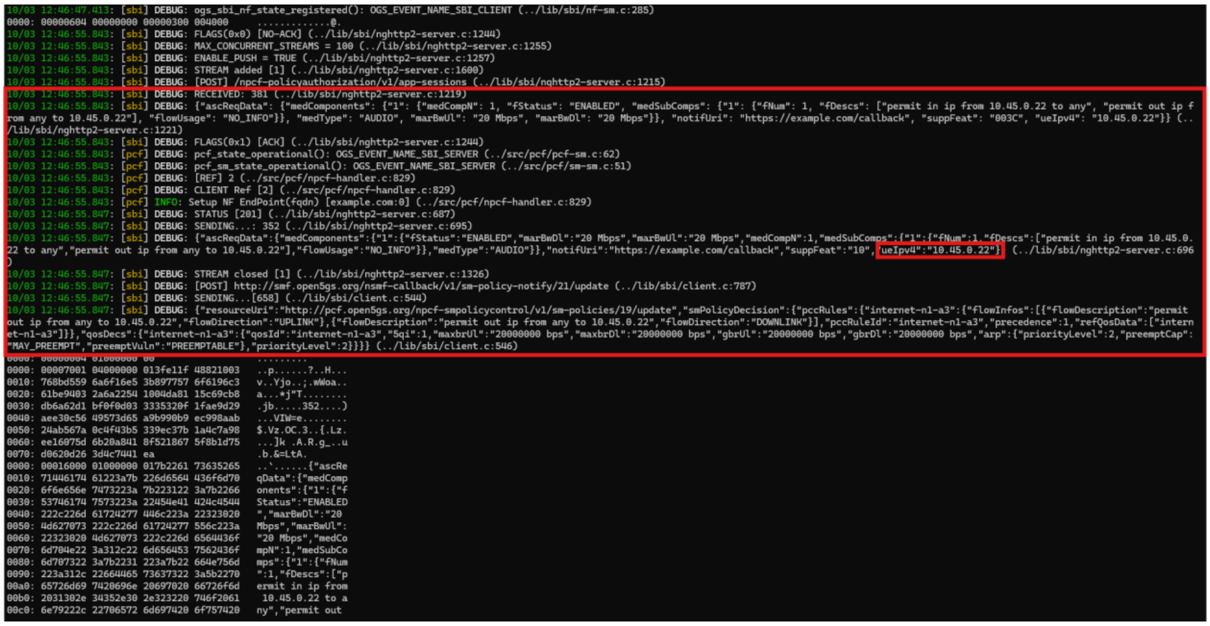
Responses

Figure 148: Calling QoS API for registering policy for UE (UERANSIM provided)



```
2025-10-03:12:46:55,851 INFO     [pcf_http2_requests.py:52] Response headers: [(b':status', b'201'), (b'server', b'Open5GS v2.7.5'), (b'date', b'Fri, 03 Oct 2025 12:46:55 GMT'), (b'content-length', b'352'), (b
'location', b'http://pcf.open5gs.org/npcf-policyauthorization/v1/app-sessions/3'), (b'content-type', b'application/json')]
2025-10-03:12:46:55,851 INFO     [pcf_http2_requests.py:57] Extracted App Session ID: 3
2025-10-03:12:46:55,851 INFO     [pcf_http2_requests.py:59] Response body: {"ascReqData":{"medComponents":{"1":{"fStatus":"ENABLED","marBwDl":"20 Mbps","marBwUl":"20 Mbps","medCompN":1,"medSubComps":{"1":{"fNu
m":1,"fDescs":["permit in ip from 10.45.0.22 to any","permit out ip from any to 10.45.0.22"],"flowUsage":"NO_INFO"}},"medType":"AUDIO"}},"notifUri":"https://example.com/callback","suppFeat":"10","ueIpv4":"10.4
5.0.22"}}
2025-10-03:12:46:55,851 DEBUG    [southbound_apis_svc.py:57] Payload to PCF: {
  "ascReqData": {
    "medComponents": {
      "1": {
        "medCompN": 1,
        "fStatus": "ENABLED",
        "medSubComps": {
          "1": {
            "fNum": 1,
            "fDescs": [
              "permit in ip from 10.45.0.22 to any",
              "permit out ip from any to 10.45.0.22"
            ],
            "flowUsage": "NO_INFO"
          }
        },
        "medType": "AUDIO",
        "marBwUl": "20 Mbps",
        "marBwDl": "20 Mbps"
      }
    },
    "notifUri": "https://example.com/callback",
    "suppFeat": "003C",
    "ueIpv4": "10.45.0.22"
  }
}
2025-10-03:12:46:55,852 INFO     [db.py:27] Mapped subscriptionId 0e65f70d-25ef-45f7-89ca-df38f475091b with appSessionId 3
INFO:     10.220.2.196:53325 - "POST /3gpp-as-session-with-qos/v1/example1/subscriptions HTTP/1.1" 201 Created
```

Figure 149: QoS API received request

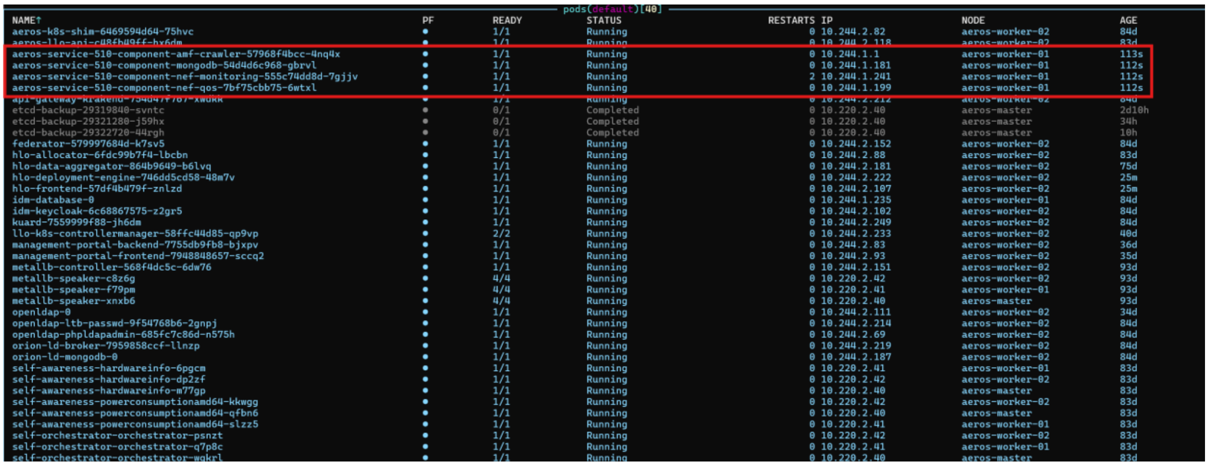


```
10/03 12:46:47.413: [sbi] DEBUG: ogs_sbi_nf_state_registered(): OGS_EVENT_NAME_SBI_CLIENT (../lib/sbi/nf-sm.c:285)
0000: 00000604 00000000 00000300 004000     ..............@.
10/03 12:46:55.843: [sbi] DEBUG: FLAGS(0x0) [NO-ACK] (../lib/sbi/nghttp2-server.c:1244)
10/03 12:46:55.843: [sbi] DEBUG: MAX_CONCURRENT_STREAMS = 100 (../lib/sbi/nghttp2-server.c:1255)
10/03 12:46:55.843: [sbi] DEBUG: ENABLE_PUSH = TRUE (../lib/sbi/nghttp2-server.c:1257)
10/03 12:46:55.843: [sbi] DEBUG: STREAM added [1] (../lib/sbi/nghttp2-server.c:1600)
10/03 12:46:55.843: [sbi] DEBUG: [POST] /npcf-policyauthorization/v1/app-sessions (../lib/sbi/nghttp2-server.c:1215)
10/03 12:46:55.843: [sbi] DEBUG: RECEIVED: 381 (../lib/sbi/nghttp2-server.c:1219)
10/03 12:46:55.843: [sbi] DEBUG: {"ascReqData": {"medComponents": {"1": {"medCompN": 1, "fStatus": "ENABLED", "medSubComps": {"1": {"fNum": 1, "fDescs": ["permit in ip from 10.45.0.22 to any", "permit out ip f
rom any to 10.45.0.22"], "flowUsage": "NO_INFO"}}, "medType": "AUDIO", "marBwUl": "20 Mbps", "marBwDl": "20 Mbps"}}, "notifUri": "https://example.com/callback", "suppFeat": "003C", "ueIpv4": "10.45.0.22"}} (..
/lib/sbi/nghttp2-server.c:1221)
10/03 12:46:55.843: [sbi] DEBUG: FLAGS(0x1) [ACK] (../lib/sbi/nghttp2-server.c:1244)
10/03 12:46:55.843: [pcf] DEBUG: pcf_state_operational(): OGS_EVENT_NAME_SBI_SERVER (../src/pcf/pcf-sm.c:62)
10/03 12:46:55.843: [pcf] DEBUG: pcf_sm_state_operational(): OGS_EVENT_NAME_SBI_SERVER (../src/pcf/sm-sm.c:51)
10/03 12:46:55.843: [pcf] DEBUG: [REF] 2 (../src/pcf/npcf-handler.c:829)
10/03 12:46:55.843: [pcf] DEBUG: CLIENT Ref [2] (../src/pcf/npcf-handler.c:829)
10/03 12:46:55.843: [pcf] INFO: Setup NF EndPoint(fqdn) [example.com:0] (../src/pcf/npcf-handler.c:829)
10/03 12:46:55.847: [sbi] DEBUG: STATUS [201] (../lib/sbi/nghttp2-server.c:687)
10/03 12:46:55.847: [sbi] DEBUG: SENDING...: 352 (../lib/sbi/nghttp2-server.c:695)
10/03 12:46:55.847: [sbi] DEBUG: {"ascReqData":{"medComponents":{"1":{"fStatus":"ENABLED","marBwDl":"20 Mbps","marBwUl":"20 Mbps","medCompN":1,"medSubComps":{"1":{"fNum":1,"fDescs":["permit in ip from 10.45.0.
22 to any","permit out ip from any to 10.45.0.22"],"flowUsage":"NO_INFO"}},"medType":"AUDIO"}},"notifUri":"https://example.com/callback","suppFeat":"10","ueIpv4":"10.45.0.22"}} (../lib/sbi/nghttp2-server.c:696
)
10/03 12:46:55.847: [sbi] DEBUG: STREAM closed [1] (../lib/sbi/nghttp2-server.c:1326)
10/03 12:46:55.847: [sbi] DEBUG: [POST] http://smf.open5gs.org/nsmf-callback/v1/sm-policy-notify/21/update (../lib/sbi/client.c:787)
10/03 12:46:55.847: [sbi] DEBUG: SENDING...[658] (../lib/sbi/client.c:544)
10/03 12:46:55.847: [sbi] DEBUG: {"resourceUri":"http://pcf.open5gs.org/npcf-smpolicycontrol/v1/sm-policies/19/update","smPolicyDecision":{"pccRules":{"internet-n1-a3":{"flowInfos":[{"flowDescription":"permit
out ip from any to 10.45.0.22","flowDirection":"UPLINK"},{"flowDescription":"permit out ip from any to 10.45.0.22","flowDirection":"DOWNLINK"}],"pccRuleId":"internet-n1-a3","precedence":1,"refQosData":["intern
et-n1-a3"]}},"qosDecs":{"internet-n1-a3":{"qosId":"internet-n1-a3","5qi":1,"maxbrUl":"20000000 bps","maxbrDl":"20000000 bps","gbrUl":"20000000 bps","gbrDl":"20000000 bps","arp":{"priorityLevel":2,"preemptCap":
"MAY_PREEMPT","preemptVuln":"PREEMPTABLE"},"priorityLevel":2}}}} (../lib/sbi/client.c:546)
0000: 00000004 01000000 00                  .........
0000: 00007001 04000000 013fe11f 48821003   ..p......?..H...
0010: 768bd559 6a6f16e5 3b897757 6f6196c3   v..Yjo..;.wWoa..
0020: 61be9403 2a6a2254 1004da81 15c69cb8   a...*j"T........
0030: db6a62d1 bf0f0d03 3335320f 1fae9d29   .jb.....352....)
0040: aee30c56 49573d65 a9b990b9 ec998aab   ...VIW=e........
0050: 24ab567a 0c4f43b5 339ec37b 1a4c7a98   $.Vz.OC.3..{.Lz.
0060: ee16075d 6b20a841 8f521867 5f8b1d75   ...]k .A.R.g_..u
0070: d0620d26 3d4c7441 ea                  .b.&=LtA.
0000: 00016000 01000000 017b2261 73635265   ..`......{"ascRe
0010: 71446174 61223a7b 226d6564 436f6d70   qData":{"medComp
0020: 6f6e656e 7473223a 7b223122 3a7b2266   onents":{"1":{"f
0030: 53746174 7573223a 22454e41 424c4544   Status":"ENABLED
0040: 222c226d 61724277 446c223a 22323020   ","marBwDl":"20 
0050: 4d627073 222c226d 61724277 556c223a   Mbps","marBwUl":
0060: 22323020 4d627073 222c226d 6564436f   "20 Mbps","medCo
0070: 6d704e22 3a312c22 6d656453 7562436f   mpN":1,"medSubCo
0080: 6d707322 3a7b2231 223a7b22 664e756d   mps":{"1":{"fNum
0090: 223a312c 22664465 73637322 3a5b2270   ":1,"fDescs":["p
00a0: 65726d69 7420696e 20697020 66726f6d   ermit in ip from
00b0: 2031302e 34352e30 2e323220 746f2061    10.45.0.22 to a
00c0: 6e79222c 22706572 6d697420 6f757420   ny","permit out 
```

Figure 150: Request received, and accepted, at PCF (open5gs)

The validation activities confirmed that the emulated UE was successfully registered through UERANSIM and fully integrated with the Open5GS core, demonstrating correct attach and signaling procedures. Building on this foundation, aerOS was able to dynamically deploy the AsSessionWithQoS component using a TOSCA descriptor, with its successful instantiation verified through deployment logs, the status of pods and services, and the availability of the exposed Swagger interface. Finally, the invocation of the QoS API established a 5QI value for the registered UE, with the interaction clearly recorded in the QoS API logs and corroborated by entries in the PCF component of the 5G core. Taken together, these results provide strong evidence that aerOS can manage the full lifecycle of advanced 5G VNFs at the edge and ensure their seamless interaction with the 5G core, thereby reinforcing its role as a flexible MetaOS for supporting end-to-end 5G deployments across the continuum.

In conclusion, the following figure illustrates the deployment of VNFs (NEF Event Monitoring and QoS) over aerOS, showcasing the platform's ability to support the seamless integration of 5G network functions with continuum computing and edge infrastructure capabilities.



```
                                              pods(default)[40]
NAME↑                                                  PF    READY    STATUS         RESTARTS IP            NODE               AGE
aeros-k8s-shim-6469594d64-75hvc                        ●     1/1      Running               0 10.244.2.82   aeros-worker-02    84d
aeros-llo-api-c48fb49ff-hx6dm                          ●     1/1      Running               0 10.244.2.118  aeros-worker-02    83d
aeros-service-510-component-amf-crawler-57968f4bcc-4nq4x ●   1/1      Running               0 10.244.1.1    aeros-worker-01    113s
aeros-service-510-component-mongodb-54d4d6c968-gbrvl   ●     1/1      Running               0 10.244.1.181  aeros-worker-01    112s
aeros-service-510-component-nef-monitoring-555c74dd8d-7gjjv ● 1/1     Running               2 10.244.1.241  aeros-worker-01    112s
aeros-service-510-component-nef-qos-7bf75cbb75-6wtxl   ●     1/1      Running               0 10.244.1.199  aeros-worker-01    112s
api-gateway-krakend-754d47f767-xwdkk                   ●     1/1      Running               0 10.244.2.212  aeros-worker-02    84d
etcd-backup-29319840-svntc                             ●     0/1      Completed             0 10.220.2.40   aeros-master       2d10h
etcd-backup-29321280-j59hx                             ●     0/1      Completed             0 10.220.2.40   aeros-master       34h
etcd-backup-29322720-44rgh                             ●     0/1      Completed             0 10.220.2.40   aeros-master       10h
federator-579997684d-k7sv5                             ●     1/1      Running               0 10.244.2.152  aeros-worker-02    84d
hlo-allocator-6fdc99b7f4-lbcbn                         ●     1/1      Running               0 10.244.2.88   aeros-worker-02    83d
hlo-data-aggregator-864b9649-b6lvq                     ●     1/1      Running               0 10.244.2.181  aeros-worker-02    75d
hlo-deployment-engine-746dd5cd58-48m7v                 ●     1/1      Running               0 10.244.2.222  aeros-worker-02    25m
hlo-frontend-57df4b479f-znlzd                          ●     1/1      Running               0 10.244.2.107  aeros-worker-02    25m
idm-database-0                                         ●     1/1      Running               0 10.244.1.235  aeros-worker-01    84d
idm-keycloak-6c68867575-z2gr5                          ●     1/1      Running               0 10.244.2.102  aeros-worker-02    84d
kuard-7559999f88-jh6dm                                 ●     1/1      Running               0 10.244.2.249  aeros-worker-02    84d
llo-k8s-controllermanager-58ffc44d85-qp9vp             ●     2/2      Running               0 10.244.2.233  aeros-worker-02    40d
management-portal-backend-7755db9fb8-bjxpv             ●     1/1      Running               0 10.244.2.83   aeros-worker-02    36d
management-portal-frontend-7948848657-sccq2            ●     1/1      Running               0 10.244.2.93   aeros-worker-02    35d
metallb-controller-568f4dc5c-6dw76                     ●     1/1      Running               0 10.244.2.151  aeros-worker-02    93d
metallb-speaker-c8z6g                                  ●     4/4      Running               0 10.220.2.42   aeros-worker-02    93d
metallb-speaker-f79pm                                  ●     4/4      Running               0 10.220.2.41   aeros-worker-01    93d
metallb-speaker-xnxb6                                  ●     4/4      Running               0 10.220.2.40   aeros-master       93d
openldap-0                                             ●     1/1      Running               0 10.244.2.111  aeros-worker-02    34d
openldap-ltb-passwd-9f54768b6-2gnpj                    ●     1/1      Running               0 10.244.2.214  aeros-worker-02    84d
openldap-phpldapadmin-685fc7c86d-n575h                 ●     1/1      Running               0 10.244.2.69   aeros-worker-02    84d
orion-ld-broker-7959858ccf-llnzp                       ●     1/1      Running               0 10.244.2.219  aeros-worker-02    84d
orion-ld-mongodb-0                                     ●     1/1      Running               0 10.244.2.187  aeros-worker-02    84d
self-awareness-hardwareinfo-6pgcm                      ●     1/1      Running               0 10.220.2.41   aeros-worker-01    83d
self-awareness-hardwareinfo-dp2zf                      ●     1/1      Running               0 10.220.2.42   aeros-worker-02    83d
self-awareness-hardwareinfo-m77gp                      ●     1/1      Running               0 10.220.2.40   aeros-master       83d
self-awareness-powerconsumptionamd64-kkwgg             ●     1/1      Running               0 10.220.2.42   aeros-worker-02    83d
self-awareness-powerconsumptionamd64-qfbn6             ●     1/1      Running               0 10.220.2.40   aeros-master       83d
self-awareness-powerconsumptionamd64-slzz5             ●     1/1      Running               0 10.220.2.41   aeros-worker-01    83d
self-orchestrator-orchestrator-psnzt                   ●     1/1      Running               0 10.220.2.42   aeros-worker-02    83d
self-orchestrator-orchestrator-q7p8c                   ●     1/1      Running               0 10.220.2.41   aeros-worker-01    83d
self-orchestrator-orchestrator-wqkrl                   ●     1/1      Running               0 10.220.2.40   aeros-master       83d
```

Figure 151: VNFs deployment over aerOS continuum

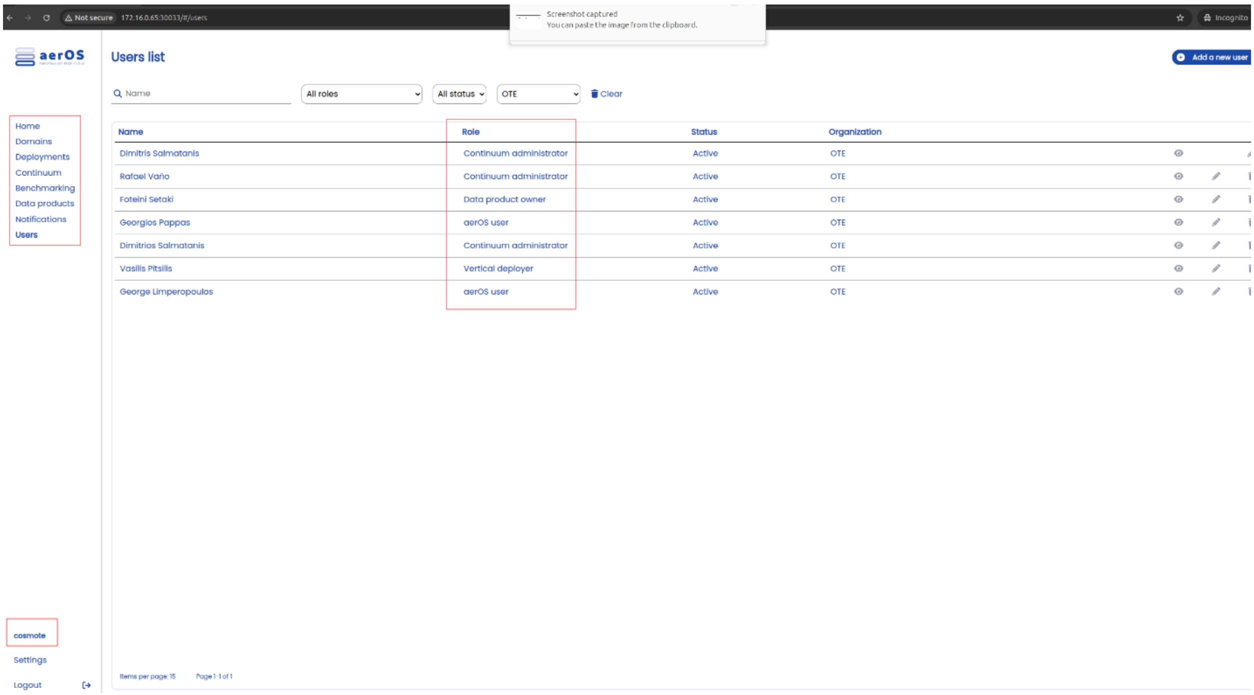
Pilot 5 – Business Process 2 – Activity - 2 (P5-BP2-VA2): Access Control based on established RBAC Rules

The validation phase aimed to confirm that the role-based access control (RBAC) mechanisms integrated in aerOS are effectively enforcing the defined policies across the continuum. Building on the development and integration activities, this phase demonstrated that roles configured in **LDAP** and **Keycloak**, and enforced by **KrakenD**, resulted in the expected access behaviors for all identified user categories. The validation activities provided evidence that only authorized roles were able to access or modify protected resources, while unauthorized attempts were consistently rejected.

The validation process followed a structured sequence of steps covering identity management, token provisioning, policy enforcement, and logging.

Step 1 – Verification of users and roles

The first step confirmed that all relevant actors were correctly registered in the aerOS identity management system and associated with the intended roles. Using the aerOS portal, each user, registered, was linked to one of the predefined roles (Continuum Administrator, aerOS User, Data Product Owner, External User, Vertical Deployer). This ensured that the role definitions created during integration were properly instantiated in the live environment. The following screenshots display some of the users created their assigned roles and the differentiations in portal provided capabilities based on their roles.



172.16.0.65:30033/#/users

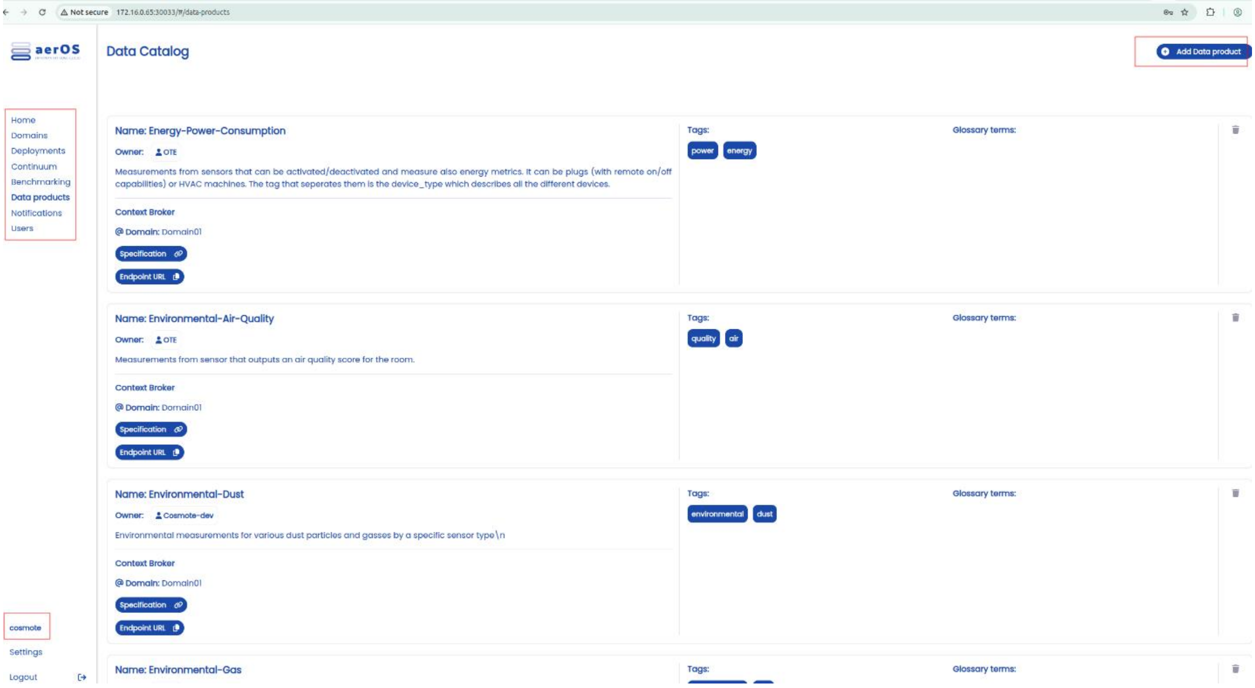
Screenshot captured
You can paste the image from the clipboard.

Users list

Name	Role	Status	Organization
Dimitris Salmatanis	Continuum administrator	Active	OTE
Rafael Vaño	Continuum administrator	Active	OTE
Foteini Setaki	Data product owner	Active	OTE
Georgios Pappas	aerOS user	Active	OTE
Dimitrios Salmatanis	Continuum administrator	Active	OTE
Vasilis Pitsilis	Vertical deployer	Active	OTE
George Limperopoulos	aerOS user	Active	OTE

Items per page: 15 Page 1-1 of 1

Figure 152: aerOS Users and assigned roles, as seen from administrator cosmote user) dashboard



172.16.0.65:30033/#/data-products

Data Catalog

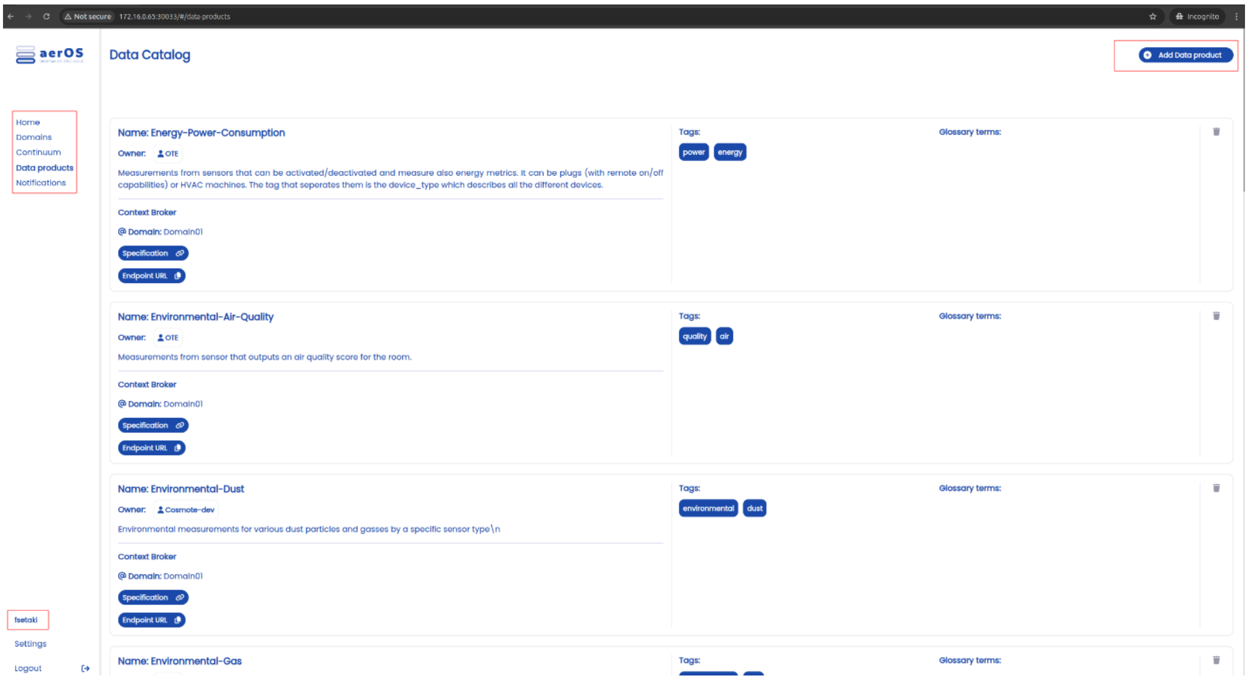
Name: Energy-Power-Consumption
Owner: OTE
Measurements from sensors that can be activated/deactivated and measure also energy metrics. It can be plugs (with remote on/off capabilities) or HVAC machines. The tag that seperates them is the device_type which describes all the different devices.
Context Broker
@ Domain: Domain01
Tags: power energy
Glossary terms:

Name: Environmental-Air-Quality
Owner: OTE
Measurements from sensor that outputs an air quality score for the room.
Context Broker
@ Domain: Domain01
Tags: quality air
Glossary terms:

Name: Environmental-Dust
Owner: Cosmote-dev
Environmental measurements for various dust particles and gasses by a specific sensor type\n
Context Broker
@ Domain: Domain01
Tags: environmental dust
Glossary terms:

Name: Environmental-Gas
Tags:
Glossary terms:

Figure 153: aerOS continuum administrator ("cosmote" user) has full access to all actions



Data Catalog

Name: Energy-Power-Consumption

Owner: OTE

Measurements from sensors that can be activated/deactivated and measure also energy metrics. It can be plugs (with remote on/off capabilities) or HVAC machines. The tag that seperates them is the device_type which describes all the different devices.

Context Broker

@ Domain: Domain01

Specification

Endpoint URL

Tags: power energy

Glossary terms:

Name: Environmental-Air-Quality

Owner: OTE

Measurements from sensor that outputs an air quality score for the room.

Context Broker

@ Domain: Domain01

Specification

Endpoint URL

Tags: quality air

Glossary terms:

Name: Environmental-Dust

Owner: Cosmote-dev

Environmental measurements for various dust particles and gasses by a specific sensor type\n

Context Broker

@ Domain: Domain01

Specification

Endpoint URL

Tags: environmental dust

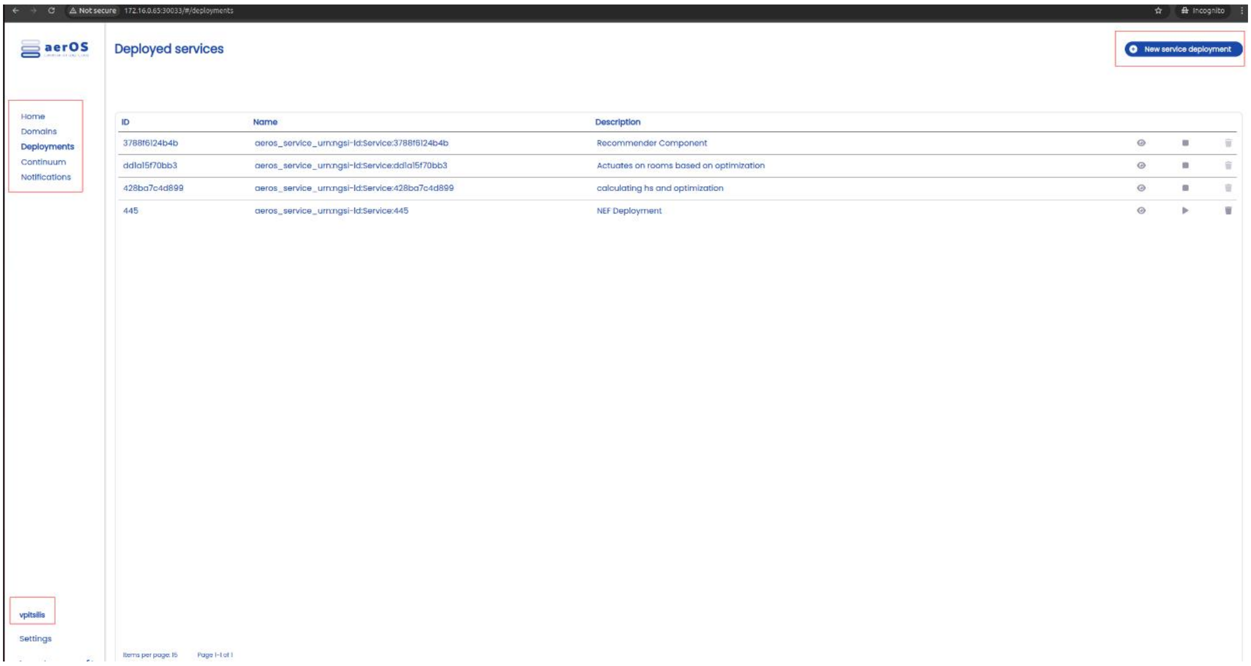
Glossary terms:

Name: Environmental-Gas

Tags:

Glossary terms:

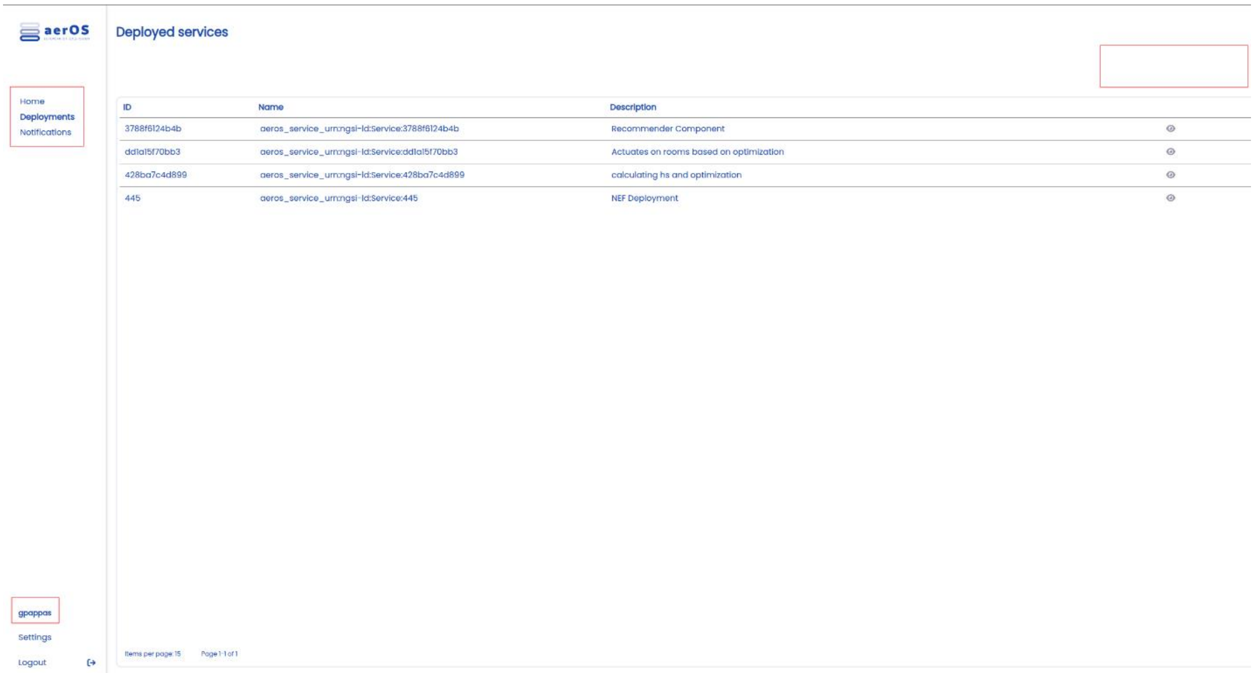
Figure 154: aerOS product owner ("fsetaki" user) can create and see data products but cannot access service deployments



Deployed services

ID	Name	Description
3788f6124b4b	aeros_service_urn:ngsi-ld:Service:3788f6124b4b	Recommender Component
dd1a15f70bb3	aeros_service_urn:ngsi-ld:Service:dd1a15f70bb3	Actuates on rooms based on optimization
428ba7c4d899	aeros_service_urn:ngsi-ld:Service:428ba7c4d899	calculating hs and optimization
445	aeros_service_urn:ngsi-ld:Service:445	NEF Deployment

Figure 155: aerOS vertical deployer user ("vpitsilis" user) can see and create service deployments



Deployed services

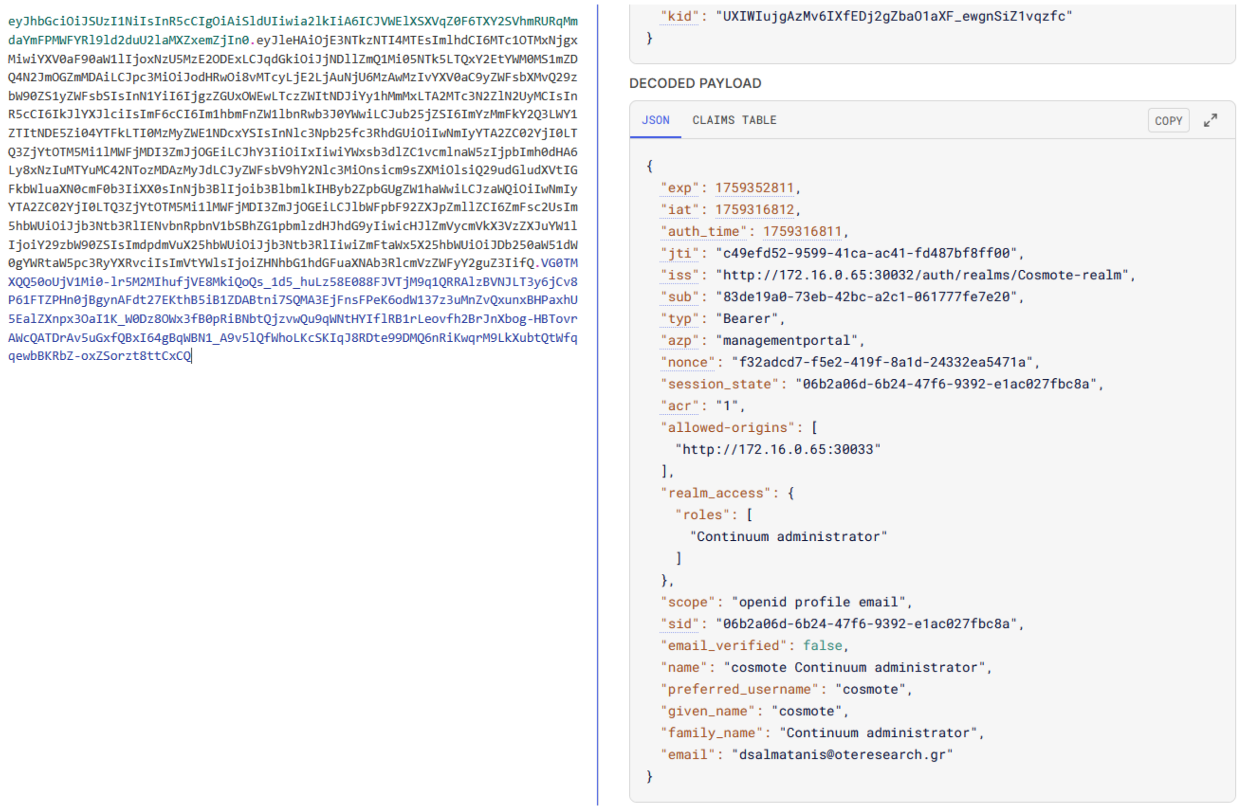
ID	Name	Description
3788f6124b4b	aeros_service_urn:ngsi-ld:Service:3788f6124b4b	Recommender Component
dd1a15f70bb3	aeros_service_urn:ngsi-ld:Service:dd1a15f70bb3	Actuates on rooms based on optimization
428ba7c4d899	aeros_service_urn:ngsi-ld:Service:428ba7c4d899	calculating hs and optimization
445	aeros_service_urn:ngsi-ld:Service:445	NEF Deployment

Items per page: 15 Page 1-1 of 1

Figure 156: aerOS user ("gpappas" user) can see service deployments

Step 2 – Token retrieval and inspection

Once user-role assignments were verified, authentication tokens were retrieved for each role. Using **Postman**, authentication requests were sent to Keycloak, which issued the corresponding JWTs. The tokens were decoded and inspected with **jwt.io** to validate that the embedded claims correctly reflected the assigned roles, issuer, audience, and expiry. This confirmed the **propagation of role attributes** from the identity store into the access tokens. The following screenshots from jwt.io depict the roles embedding in authorization tokens.



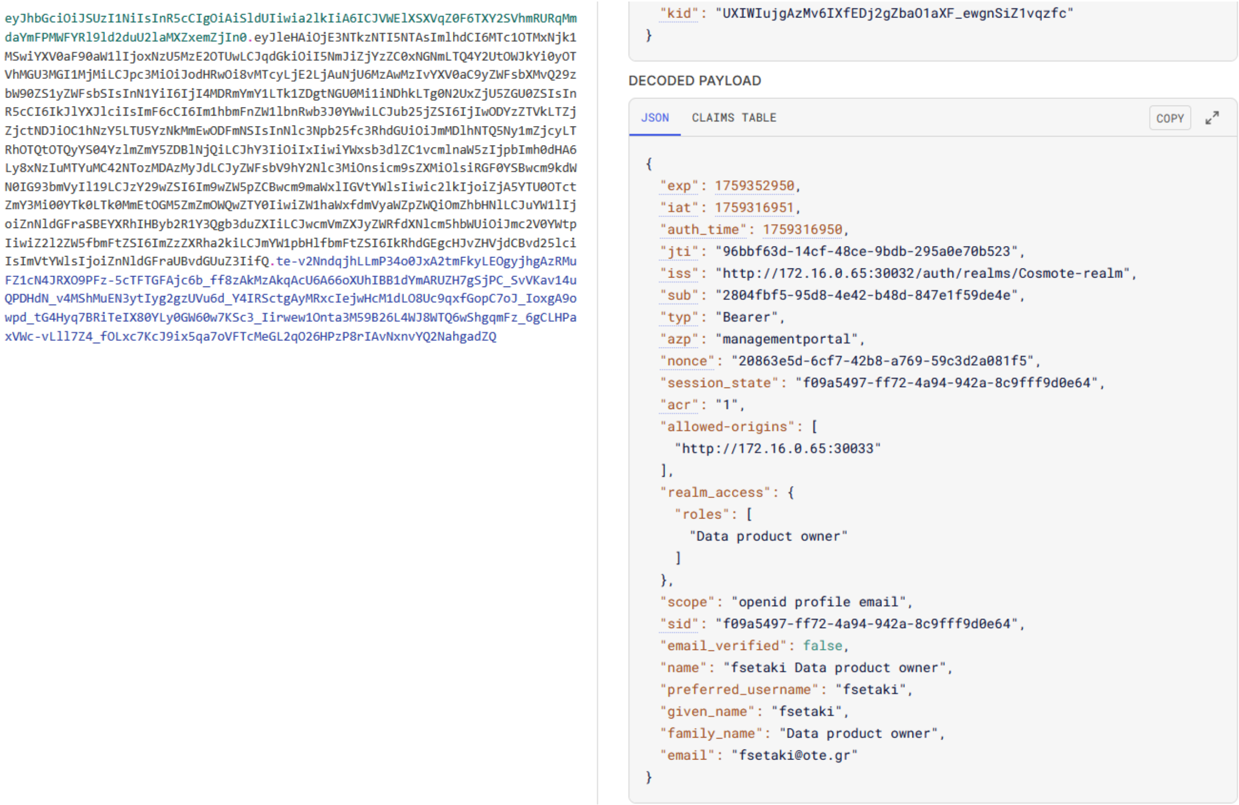
```
eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJVWElXSXVqZ0F6TXY2SVhmRURqMm
daYmFPMWFYRl9ld2duU2laMXZxemZjIn0.eyJleHAiOjE3NTkzNTI4MTEsImlhdCI6MTc1OTMxNjgx
MiwiYXV0aF90aW1lIjoxNzU5MzE2ODExLCJqdGkiOiJjNDllZmQ1Mi05NTk5LTQxY2EtYWM0MS1mZD
Q4N2JmOGZmMDAiLCJpc3MiOiJodHRwOi8vMTcyLjE2LjAuNjU6MzAwMzIvYXV0aC9yZWFsbXMvQ29z
bW90ZS1yZWFsbSIsInN1YiI6IjgzZGUxOWEwLTczZWItNDJiYy1hMmMxLTA2MTc3N2ZlN2UyMCIsIn
R5cCI6IkJlYXJlciIsImF6cCI6Im1hbmFnZW1lbnRwb3J0YWwiLCJub25jZSI6ImYzMmFkY2Q3LWY1
ZTItNDE5Zi04YTFkLTI0MzMyZWE1NDcxYSIsInNlc3Npb25fc3RhdGUiOiIwNmIyYTA2ZC02YjI0LT
Q3ZjYtOTM5Mi1lMWFjMDI3ZmJjOGEiLCJhY3IiOiIxIiwiYWxsb3dlZC1vcmlnaW5zIjpbImh0dHA6
Ly8xNzIuMTYuMC42NTozMDAzMyJdLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsiQ29udGludXVtIG
FkbWluaXN0cmF0b3IiXX0sInNjb3BlIjoib3BlbmlkIHByb2ZpbGUgZW1haWwiLCJzaWQiOiIwNmIy
YTA2ZC02YjI0LTQ3ZjYtOTM5Mi1lMWFjMDI3ZmJjOGEiLCJlbWFpbF92ZXJpZmllZCI6ZmFsc2UsIm
5hbWUiOiJjb3Ntb3RlIENvbnRpbnV1bSBhZG1pbmlzdHJhdG9yIiwicHJlZmVycmVkX3VzZXJuYW1l
IjoiY29zbW90ZSIsImdpdmVuX25hbWUiOiJjb3Ntb3RlIiwiZmFtaWx5X25hbWUiOiJDb250aW51dW
0gYWRtaW5pc3RyYXRvciIsImVtYWlsIjoiZHNhbG1hdGFuaXNAb3RlcmVzZWFyY2guZ3IifQ.VG0TM
XQQ50oUjV1Mi0-lr5M2MIhufjVE8MkiQoQs_1d5_huLz58E088FJVTjM9q1QRRAlzBVNJLT3y6jCv8
P61FTZPHn0jBgynAFdt27EKthB5iB1ZDABtni7SQMA3EjFnsFPeK6odW137z3uMnZvQxunxBHPaxhU
5EalZXnpx3OaI1K_W0Dz8OWx3fB0pRiBNbtQjzvwQu9qWNtHYIflRB1rLeovfh2BrJnXbog-HBTovr
AWcQATDrAv5uGxfQBxI64gBqWBN1_A9v5lQfWhoLKcSKIqJ8RDte99DMQ6nRiKwqrM9LkXubtQtWfq
qewbBKRbZ-oxZSorzt8ttCxCQ
```

```
  "kid": "UXIWIujgAzMv6IXfEDj2gZba01aXF_ewgnSiZ1vqzfc"
}
```

DECODED PAYLOAD

```
{
  "exp": 1759352811,
  "iat": 1759316812,
  "auth_time": 1759316811,
  "jti": "c49efd52-9599-41ca-ac41-fd487bf8ff00",
  "iss": "http://172.16.0.65:30032/auth/realms/Cosmote-realm",
  "sub": "83de19a0-73eb-42bc-a2c1-061777fe7e20",
  "typ": "Bearer",
  "azp": "managementportal",
  "nonce": "f32adcd7-f5e2-419f-8a1d-24332ea5471a",
  "session_state": "06b2a06d-6b24-47f6-9392-e1ac027fbc8a",
  "acr": "1",
  "allowed-origins": [
    "http://172.16.0.65:30033"
  ],
  "realm_access": {
    "roles": [
      "Continuum administrator"
    ]
  },
  "scope": "openid profile email",
  "sid": "06b2a06d-6b24-47f6-9392-e1ac027fbc8a",
  "email_verified": false,
  "name": "cosmote Continuum administrator",
  "preferred_username": "cosmote",
  "given_name": "cosmote",
  "family_name": "Continuum administrator",
  "email": "dsalmatanis@oteresearch.gr"
}
```

Figure 157: aerOS "cosmote" with "Continuum administrator" role embedded in token



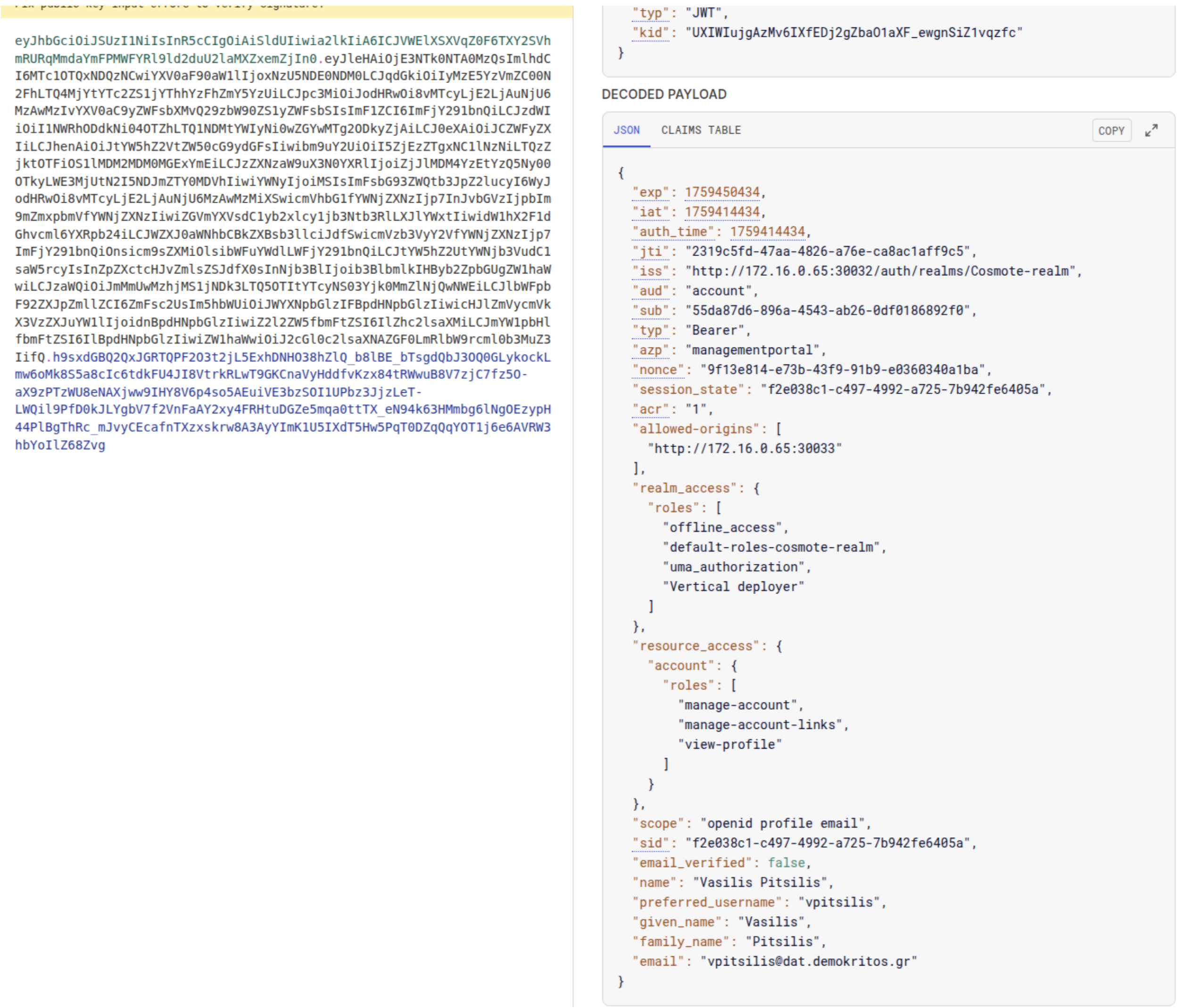
```
eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJVWElXSXVqZ0F6TXY2SVhmRURqMm
daYmFPMWFYRl9ld2duU2laMXZxemZjIn0.eyJleHAiOjE3NTkzNTI5NTAsImlhdCI6MTc1OTMxNjk1
MSwiYXV0aF90aW1lIjoxNzU5MzE2OTUwLCJqdGkiOiI5NmJiZjYzZC0xNGNmLTQ4Y2UtOWJkYi0yOT
VhMGU3MGI1MjMiLCJpc3MiOiJodHRwOi8vMTcyLjE2LjAuNjU6MzAwMzIvYXV0aC9yZWFsbXMvQ29z
bW90ZS1yZWFsbSIsInN1YiI6IjI4MDRmYmY1LTk1ZDgtNGU0Mi1iNDhkLTg0N2UxZjU5ZGU0ZSIsIn
R5cCI6IkJlYXJlciIsImF6cCI6Im1hbmFnZW1lbnRwb3J0YWwiLCJub25jZSI6IjIwODYzZTVkLTZj
ZjctNDJiOC1hNzY5LTU5YzNkMmEwODFmNSIsInNlc3Npb25fc3RhdGUiOiJmMDlhNTQ5Ny1mZjcyLT
RhOTQtOTQyYS04YzlmZmY5ZDBlNjQiLCJhY3IiOiIxIiwiYWxsb3dlZC1vcmlnaW5zIjpbImh0dHA6
Ly8xNzIuMTYuMC42NTozMDAzMyJdLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsiRGF0YSBwcm9kdW
N0IG93bmVyIl19LCJzY29wZSI6Im9wZW5pZCBwcm9maWxlIGVtYWlsIiwic2lkIjoiZjA5YTU0OTct
ZmY3Mi00YTk0LTk0MmEtOGM5ZmZmOWQwZTY0IiwiZW1haWxfdmVyaWZpZWQiOmZhbHNlLCJuYW1lIj
oiZnNldGFraSBEYXRhIHByb2R1Y3Qgb3duZXIiLCJwcmVmZXJyZWRfdXNlcm5hbWUiOiJmc2V0YWtp
IiwiZ2l2ZW5fbmFtZSI6ImZzZXRha2kiLCJmYW1pbHlfbmFtZSI6IkRhdGEgcHJvZHVjdCBvd25lci
IsImVtYWlsIjoiZnNldGFraUBvdGUuZ3IifQ.te-v2NndqjhLLmP34o0JxA2tmFkyLEOgyjhgAzRMu
FZ1cN4JRXO9PFz-5cTFTGFAjc6b_ff8zAkMzAkqAcU6A66oXUhIBB1dYmARUZH7gSjPC_SvVKav14u
QPDHdN_v4MShMuEN3ytIyg2gzUVu6d_Y4IRSctgAyMRxcIejwHcM1dLO8Uc9qxfGopC7oJ_IoxgA9o
wpd_tG4Hyq7BRiTeIX80YLy0GW60w7KSc3_Iirwew1Onta3M59B26L4WJ8WTQ6wShgqmFz_6gCLHPa
xVWc-vLll7Z4_fOLxc7KcJ9ix5qa7oVFTcMeGL2qO26HPzP8rIAvNxnvYQ2NahgadZQ
```

```
  "kid": "UXIWIujgAzMv6IXfEDj2gZba01aXF_ewgnSiZ1vqzfc"
}
```

DECODED PAYLOAD

```
{
  "exp": 1759352950,
  "iat": 1759316951,
  "auth_time": 1759316950,
  "jti": "96bbf63d-14cf-48ce-9bdb-295a0e70b523",
  "iss": "http://172.16.0.65:30032/auth/realms/Cosmote-realm",
  "sub": "2804fbf5-95d8-4e42-b48d-847e1f59de4e",
  "typ": "Bearer",
  "azp": "managementportal",
  "nonce": "20863e5d-6cf7-42b8-a769-59c3d2a081f5",
  "session_state": "f09a5497-ff72-4a94-942a-8c9fff9d0e64",
  "acr": "1",
  "allowed-origins": [
    "http://172.16.0.65:30033"
  ],
  "realm_access": {
    "roles": [
      "Data product owner"
    ]
  },
  "scope": "openid profile email",
  "sid": "f09a5497-ff72-4a94-942a-8c9fff9d0e64",
  "email_verified": false,
  "name": "fsetaki Data product owner",
  "preferred_username": "fsetaki",
  "given_name": "fsetaki",
  "family_name": "Data product owner",
  "email": "fsetaki@ote.gr"
}
```

Figure 158: aerOS "fsetaki" with "Data product owner" role embedded in token



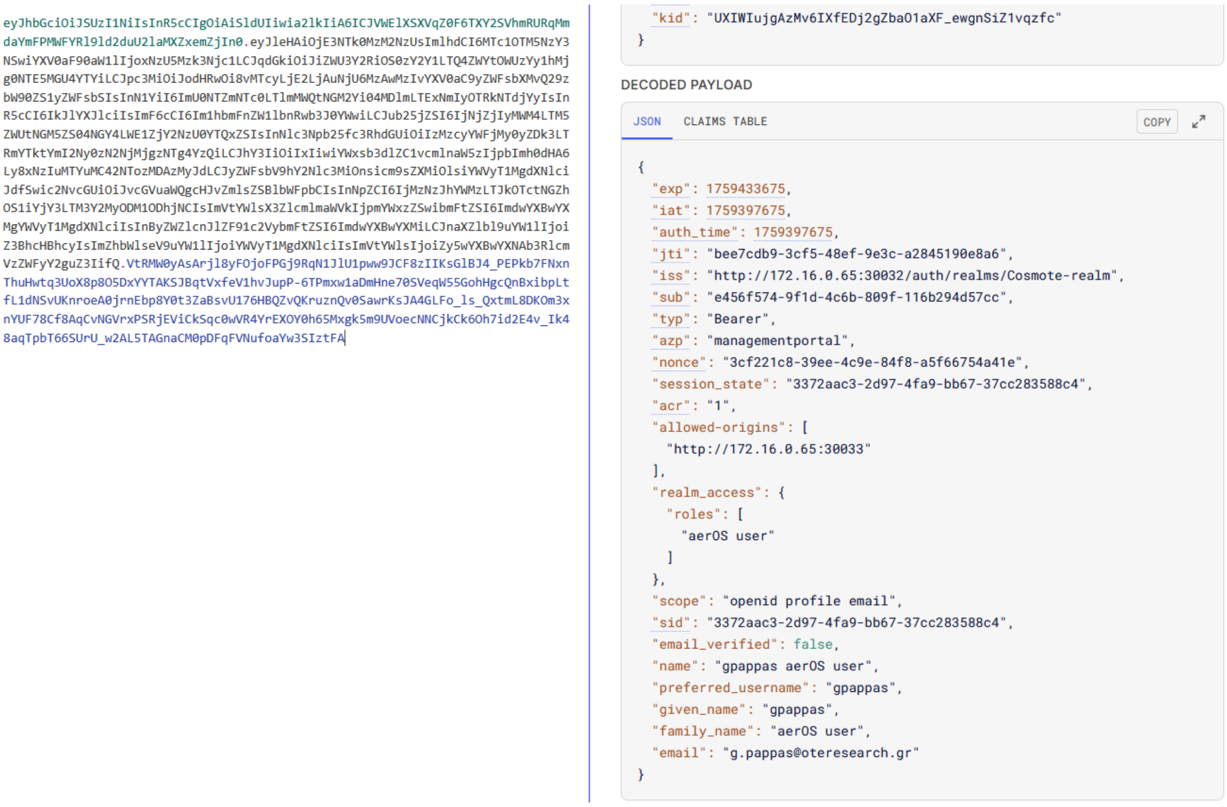
```
eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJVWElXSXVqZ0F6TXY2SVh
mRURqMmdaYmFPMWFYRl9ld2duU2laMXZxemZjIn0.eyJleHAiOjE3NTk0NTA0MzQsImlhdC
I6MTc1OTQxNDQzNCwiYXV0aF90aW1lIjoxNzU5NDE0NDM0LCJqdGkiOiIyMzE5YzVmZC00N
2FhLTQ4MjYtYTc2ZS1jYThhYzFhZmY5YzUiLCJpc3MiOiJodHRwOi8vMTcyLjE2LjAuNjU6
MzAwMzIvYXV0aC9yZWFsbXMvQ29zbW90ZS1yZWFsbSIsImF1ZCI6ImFjY291bnQiLCJzdWI
iOiI1NWRhODdkNi04OTZhLTQ1NDMtYWIyNi0wZGYwMTg2ODkyZjAiLCJ0eXAiOiJCZWFyZX
IiLCJhenAiOiJtYW5hZ2VtZW50cG9ydGFsIiwibm9uY2UiOiI5ZjEzZTgxNC1lNzNiLTQzZ
jktOTFiOS1lMDM2MDM0MGExYmEiLCJzZXNzaW9uX3N0YXRlIjoiZjJlMDM4YzEtYzQ5Ny00
OTkyLWE3MjUtN2I5NDJmZTY0MDVhIiwiYWNyIjoiMSIsImFsbG93ZWQtb3JpZ2lucyI6WyJ
odHRwOi8vMTcyLjE2LjAuNjU6MzAwMzMiXSwicmVhbG1fYWNjZXNzIjp7InJvbGVzIjpbIm
9mZmxpbmVfYWNjZXNzIiwiZGVmYXVsdC1yb2xlcy1jb3Ntb3RlLXJlYWxtIiwidW1hX2F1d
Ghvcml6YXRpb24iLCJWZXJ0aWNhbCBkZXBsb3llciJdfSwicmVzb3VyY2VfYWNjZXNzIjp7
ImFjY291bnQiOnsicm9sZXMiOlsibWFuYWdlLWFjY291bnQiLCJtYW5hZ2UtYWNjb3VudC1
saW5rcyIsInZpZXctcHJvZmlsZSJdfX0sInNjb3BlIjoib3BlbmlkIHByb2ZpbGUgZW1haW
wiLCJzaWQiOiJmMmUwMzhjMS1jNDk3LTQ5OTItYTcyNS03Yjk0MmZlNjQwNWEiLCJlbWFpb
F92ZXJpZmllZCI6ZmFsc2UsIm5hbWUiOiJWYXNpbGlzIFBpdHNpbGlzIiwicHJlZmVycmVk
X3VzZXJuYW1lIjoidnBpdHNpbGlzIiwiZ2l2ZW5fbmFtZSI6IlZhc2lsaXMiLCJmYW1pbHl
fbmFtZSI6IlBpdHNpbGlzIiwiZW1haWwiOiJ2cGl0c2lsaXNAZGF0LmRlbW9rcml0b3MuZ3
IifQ.h9sxdGBQ2QxJGRTQPF2O3t2jL5ExhDNHO38hZlQ_b8lBE_bTsgdQbJ3OQ0GLykockL
mw6oMk8S5a8cIc6tdkFU4JI8VtrkRLwT9GKCnaVyHddfvKzx84tRWwuB8V7zjC7fz5O-
aX9zPTzWU8eNAXjww9IHY8V6p4so5AEuiVE3bzSOI1UPbz3JjzLeT-
LWQil9PfD0kJLYgbV7f2VnFaAY2xy4FRHtuDGZe5mqa0ttTX_eN94k63HMmbg6lNgOEzypH
44PlBgThRc_mJvyCEcafnTXzxskrw8A3AyYImK1U5IXdT5Hw5PqT0DZqQqYOT1j6e6AVRW3
hbYoIlZ68Zvg
```

```
  "typ": "JWT",
  "kid": "UXIWIujgAzMv6IXfEDj2gZba01aXF_ewgnSiZ1vqzfc"
}
```

DECODED PAYLOAD

```
{
  "exp": 1759450434,
  "iat": 1759414434,
  "auth_time": 1759414434,
  "jti": "2319c5fd-47aa-4826-a76e-ca8ac1aff9c5",
  "iss": "http://172.16.0.65:30032/auth/realms/Cosmote-realm",
  "aud": "account",
  "sub": "55da87d6-896a-4543-ab26-0df0186892f0",
  "typ": "Bearer",
  "azp": "managementportal",
  "nonce": "9f13e814-e73b-43f9-91b9-e0360340a1ba",
  "session_state": "f2e038c1-c497-4992-a725-7b942fe6405a",
  "acr": "1",
  "allowed-origins": [
    "http://172.16.0.65:30033"
  ],
  "realm_access": {
    "roles": [
      "offline_access",
      "default-roles-cosmote-realm",
      "uma_authorization",
      "Vertical deployer"
    ]
  },
  "resource_access": {
    "account": {
      "roles": [
        "manage-account",
        "manage-account-links",
        "view-profile"
      ]
    }
  },
  "scope": "openid profile email",
  "sid": "f2e038c1-c497-4992-a725-7b942fe6405a",
  "email_verified": false,
  "name": "Vasilis Pitsilis",
  "preferred_username": "vpitsilis",
  "given_name": "Vasilis",
  "family_name": "Pitsilis",
  "email": "vpitsilis@dat.demokritos.gr"
}
```

Figure 159: aerOS "vpitsilis" with "Vertical deployer" role embedded in token



```
eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSldUIiwia2lkIiA6ICJVWElXSXVqZ0F6TXY2SVhmRURqMm
daYmFPMWFYRl9ld2duU2laMXZxemZjIn0.eyJleHAiOjE3NTk0MzM2NzUsImlhdCI6MTc1OTM5NzY3
NSwiYXV0aF90aW1lIjoxNzU5Mzk3Njc1LCJqdGkiOiJiZWU3Y2RiOS0zY2Y1LTQ4ZWYtOWUzYy1hMj
g0NTE5MGU4YTYiLCJpc3MiOiJodHRwOi8vMTcyLjE2LjAuNjU6MzAwMzIvYXV0aC9yZWFsbXMvQ29z
bW90ZS1yZWFsbSIsInN1YiI6ImU0NTZmNTc0LTlmMWQtNGM2Yi04MDlmLTExNmIyOTRkNTdjYyIsIn
R5cCI6IkJlYXJlciIsImF6cCI6Im1hbmFnZW1lbnRwb3J0YWwiLCJub25jZSI6IjNjZjIyMWM4LTM5
ZWUtNGM5ZS04NGY4LWE1ZjY2NzU0YTQxZSIsInNlc3Npb25fc3RhdGUiOiIzMzcyYWFjMy0yZDk3LT
RmYTktYmI2Ny0zN2NjMjgzNTg4YzQiLCJhY3IiOiIxIiwiYWxsb3dlZC1vcmlnaW5zIjpbImh0dHA6
Ly8xNzIuMTYuMC42NTozMDAzMyJdLCJyZWFsbV9hY2Nlc3MiOnsicm9sZXMiOlsiYWVyT1MgdXNlci
JdfSwic2NvcGUiOiJvcGVuaWQgcHJvZmlsZSBlbWFpbCIsInNpZCI6IjMzNzJhYWMzLTJkOTctNGZh
OS1iYjY3LTM3Y2MyODM1ODhjNCIsImVtYWlsX3ZlcmlmaWVkIjpmYWxzZSwibmFtZSI6ImdwYXBwYX
MgYWVyT1MgdXNlciIsInByZWZlcnJlZF91c2VybmFtZSI6ImdwYXBwYXMiLCJnaXZlbl9uYW1lIjoi
Z3BhcHBhcyIsImZhbWlseV9uYW1lIjoiYWVyT1MgdXNlciIsImVtYWlsIjoiZy5wYXBwYXNAb3Rlcm
VzZWFyY2guZ3IifQ.VtRMW0yAsArjl8yFOjoFPGj9RqN1JlU1pww9JCF8zIIKsGlBJ4_PEPkb7FNxn
ThuHwtq3UoX8p8O5DxYYTAKSJBqtVxfeV1hvJupP-6TPmxw1aDmHne70SVeqW55GohHgcQnBxibpLt
fL1dNSvUKnroeA0jrnEbp8Y0t3ZaBsvU176HBQZvQKruznQv0SawrKsJA4GLFo_ls_QxtmL8DKOm3x
nYUF78Cf8AqCvNGVrxPSRjEViCkSqc0wVR4YrEXOY0h65Mxgk5m9UVoecNNCjkCk6Oh7id2E4v_Ik4
8aqTpbT66SUrU_w2AL5TAGnaCM0pDFqFVNufoaYw3SIztFA
```

```
  "kid": "UXIWIujgAzMv6IXfEDj2gZba01aXF_ewgnSiZ1vqzfc"
}
```

```
{
  "exp": 1759433675,
  "iat": 1759397675,
  "auth_time": 1759397675,
  "jti": "bee7cdb9-3cf5-48ef-9e3c-a2845190e8a6",
  "iss": "http://172.16.0.65:30032/auth/realms/Cosmote-realm",
  "sub": "e456f574-9f1d-4c6b-809f-116b294d57cc",
  "typ": "Bearer",
  "azp": "managementportal",
  "nonce": "3cf221c8-39ee-4c9e-84f8-a5f66754a41e",
  "session_state": "3372aac3-2d97-4fa9-bb67-37cc283588c4",
  "acr": "1",
  "allowed-origins": [
    "http://172.16.0.65:30033"
  ],
  "realm_access": {
    "roles": [
      "aerOS user"
    ]
  },
  "scope": "openid profile email",
  "sid": "3372aac3-2d97-4fa9-bb67-37cc283588c4",
  "email_verified": false,
  "name": "gpappas aerOS user",
  "preferred_username": "gpappas",
  "given_name": "gpappas",
  "family_name": "aerOS user",
  "email": "g.pappas@oteresearch.gr"
}
```

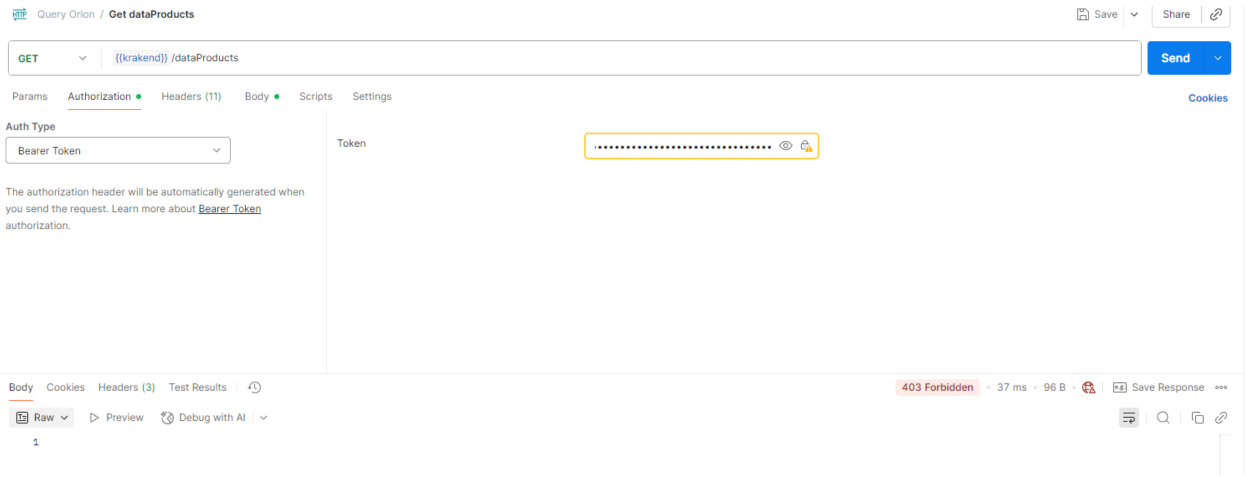
Figure 160: aerOS "gpappas" with "aerOS user" role embedded in token

Step 3 – Execution of API requests

With valid tokens available, a series of API calls was executed against representative aerOS endpoints. These endpoints were chosen because they map directly to role responsibilities:

- **Data product management** (*/dataProducts, /dataCatalog*) reflects the tasks of **Data Product Owners**, who must be able to create and delete their own products.
- **Service deployment and removal** (*/hlo_fe/services/{{service_id}}*) corresponds to **Vertical Deployers**, who must both deploy and remove IoT services, and **Administrators**, who retain global authority.
- **Resource discovery** (*/entities?type=InfrastructureElement)* demonstrates read-only operations available to all roles for monitoring and assessment.
- **Negative scenarios** (expired or tampered tokens) confirm that invalid credentials are correctly rejected.

Requests were made with tokens belonging to each role. Positive cases (authorized actions) returned 200 or 201 HTTP status codes, while negative cases (unauthorized attempts) returned 403 Forbidden or 401 Unauthorized. This validated the **correct enforcement of RBAC rules at the gateway level**, ensuring each role was restricted to its expected scope of actions.



Query Orion / Get dataProducts

GET {{krakend}} /dataProducts

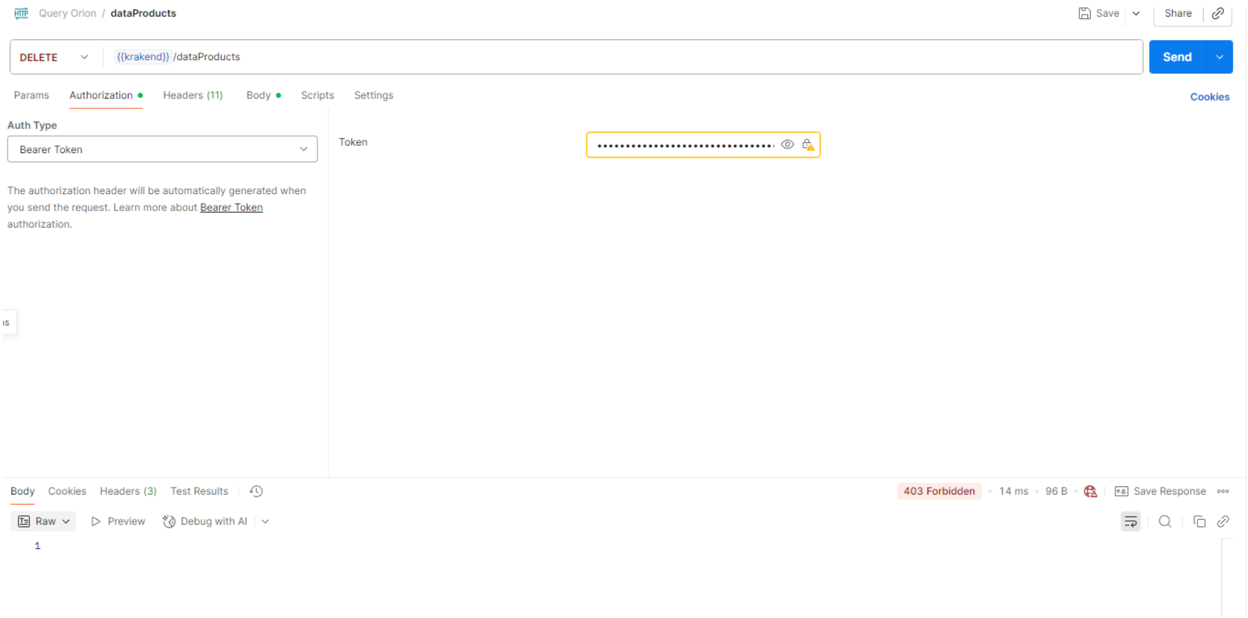
Auth Type: Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about Bearer Token authorization.

Token: ••••••••••••••••••••••••••••••••••••

Body: 403 Forbidden · 37 ms · 96 B

Figure 161: User "gpappas" who is "aerOS user" calling GET /dataProducts and denied with 403 HTTP response code



Query Orion / dataProducts

DELETE {{krakend}} /dataProducts

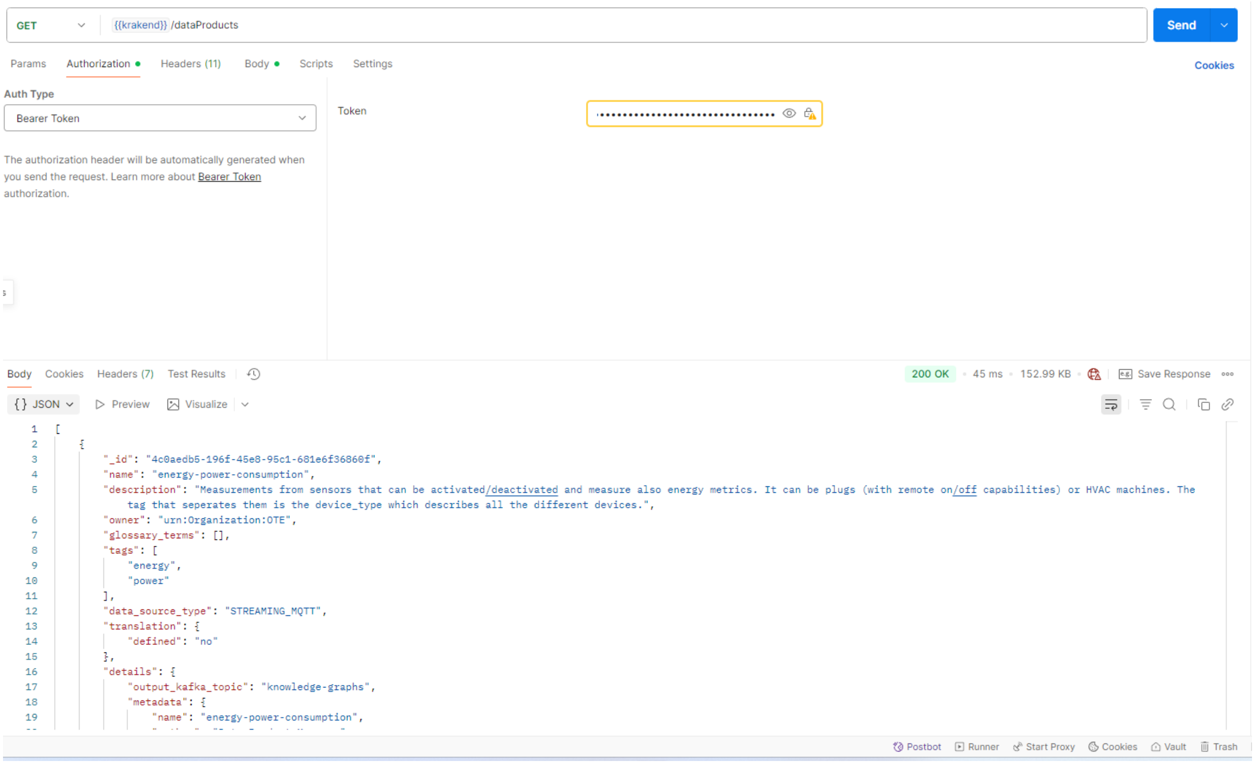
Auth Type: Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about Bearer Token authorization.

Token: ••••••••••••••••••••••••••••••••••••

Body: 403 Forbidden · 14 ms · 96 B

Figure 162: User "gpappas" who is "aerOS user" calling DELETE /dataProducts and denied with 403 response code



Auth Type: Bearer Token

The authorization header will be automatically generated when you send the request. Learn more about Bearer Token authorization.

Token: ••••••••••••••••••••••••••••••••

```
[
    {
        "_id": "4c0aedb5-196f-45e8-95c1-681e6f36860f",
        "name": "energy-power-consumption",
        "description": "Measurements from sensors that can be activated/deactivated and measure also energy metrics. It can be plugs (with remote on/off capabilities) or HVAC machines. The
            tag that seperates them is the device_type which describes all the different devices.",
        "owner": "urn:Organization:OTE",
        "glossary_terms": [],
        "tags": [
            "energy",
            "power"
        ],
        "data_source_type": "STREAMING_MQTT",
        "translation": {
            "defined": "no"
        },
        "details": {
            "output_kafka_topic": "knowledge-graphs",
            "metadata": {
                "name": "energy-power-consumption",
```

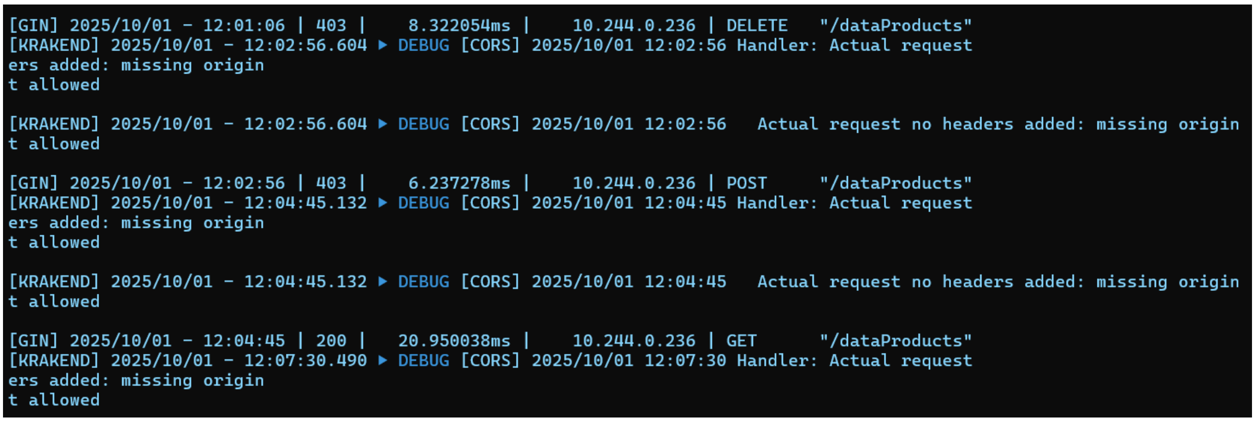
Figure 163: User "fsetaki" who is "Product owner" calling GET /dataProducts and receiving 200 response code

Please note that for the execution of test cases, Postman was used to issue requests with tokens corresponding to different aerOS roles. While Postman provides clear evidence of the request and response (e.g., 200 OK, 403 Forbidden), it does not directly display the user identity or role associated with the token, as these are encoded within the JWT. Therefore, the screenshots included here are illustrative and should be interpreted in conjunction with the validation matrix and the token inspection results. In some cases, KrakenD log excerpts are also provided to establish a traceable link between the user identity, role claims, and the authorization outcome.

Step 4 – Validation through gateway logs

In addition to Postman request/response evidence, validation activities were cross-checked against logs produced by the KrakenD API Gateway. These logs record the enforcement decision taken for each request, including the accessed endpoint, the method invoked, the response code issued, and timing information. For example, in the validation of the */dataProducts* endpoint, the logs clearly show that unauthorized requests to create or delete resources were blocked with 403 Forbidden, while authorized read operations were permitted with 200 OK.

As with the Postman screenshots, the log entries are not self-explanatory in terms of *who* issued the request, since user identifiers and roles are embedded in the JWT and not echoed in plain text by default. For this reason, log evidence is presented in conjunction with the validation matrix and token inspection results, which together provide a complete trace from *role assignment → token claims → gateway decision → observed outcome*.



```
[GIN] 2025/10/01 - 12:01:06 | 403 |    8.322054ms |    10.244.0.236 | DELETE   "/dataProducts"
[KRAKEND] 2025/10/01 - 12:02:56.604 ▶ DEBUG [CORS] 2025/10/01 12:02:56 Handler: Actual request
ers added: missing origin
t allowed

[KRAKEND] 2025/10/01 - 12:02:56.604 ▶ DEBUG [CORS] 2025/10/01 12:02:56   Actual request no headers added: missing origin
t allowed

[GIN] 2025/10/01 - 12:02:56 | 403 |    6.237278ms |    10.244.0.236 | POST     "/dataProducts"
[KRAKEND] 2025/10/01 - 12:04:45.132 ▶ DEBUG [CORS] 2025/10/01 12:04:45 Handler: Actual request
ers added: missing origin
t allowed

[KRAKEND] 2025/10/01 - 12:04:45.132 ▶ DEBUG [CORS] 2025/10/01 12:04:45   Actual request no headers added: missing origin
t allowed

[GIN] 2025/10/01 - 12:04:45 | 200 |   20.950038ms |    10.244.0.236 | GET      "/dataProducts"
[KRAKEND] 2025/10/01 - 12:07:30.490 ▶ DEBUG [CORS] 2025/10/01 12:07:30 Handler: Actual request
ers added: missing origin
t allowed
```

Figure 164: Krakend logs corresponding to the previous presented Postman requests

Summarizing these, indicative, logs we see how authorized and unauthorized responses are logged for the following endpoints (while results were similar, and correct, for all endpoints and tokens):

403 | DELETE "/dataProducts"

403 | POST "/dataProducts"

200 | GET "/dataProducts"

This evidence demonstrates that KrakenD consistently enforced the RBAC policies configured, denying unauthorized actions and allowing permitted ones.

Step 5 – Negative and edge case testing

Several edge cases were tested to validate robustness. These included expired tokens, tampered JWTs, and requests with incorrect audience claims. In all cases, the system correctly rejected requests with 401 Unauthorized. This confirmed that the system is **resilient to common misuse and security violations** in addition to enforcing the nominal RBAC rules.

Validation Matrix

The following table demonstrates some of the validation activities carried out across representative aerOS endpoints, mapping each role to its expected access permissions and the observed outcomes.

Test ID	Endpoint	Method	Required Role(s)	Continuum Administrator	aerOS User	Data Product Owner	Vertical Deployer	Observed / Evidence
VA-01	/dataProducts	GET	All roles	☑ 200	✗ 403	☑ 200	✗ 403	Data products restricted to product owners
VA-02	/dataProducts/{dataProductId}	GET	All roles	☑ 200	✗ 403	☑ 200	✗ 403	Retrieval restricted
VA-03	/dataProducts	POST	Data Product Owner, Admin	☑ 201	✗ 403	☑ 201	✗ 403	Creation restricted
VA-04	/dataProducts/{dataProductId}	DELETE	Data Product Owner, Admin	☑ 200	✗ 403	☑ 200	✗ 403	Lifecycle managed by owner/admin
VA-05	/dataCatalog	POST	Data Product Owner, Admin	☑ 201	✗ 403	☑ 201	✗ 403	Registration restricted
VA-06	/dataCatalog/{dataProductId}	DELETE	Data Product Owner, Admin	☑ 200	✗ 403	☑ 200	✗ 403	Owner/admin removal confirmed

VA-07	/hlo_fe/services/{{service_id}}	POST	Vertical Deployer, Admin	☑ 201	✗ 403	✗ 403	☑ 201	Deployment limited to deployers
VA-08	/hlo_fe/services/{{service_id}}	DELETE	Vertical Deployer, Admin	☑ 200	✗ 403	✗ 403	☑ 200	Deployer/admin service removal
VA-09	/entities?type=InfrastructureElement	GET	All roles	☑ 200	☑ 200	☑ 200	☑ 200	Resource discovery available
VA-10	/entities?type=Service	GET	All roles	☑ 200	☑ 200	✗ 401	☑ 200	Resource discovery available
VA-11	Any endpoint (expired/tampered token)	Any	None	✗ 401	✗ 401	✗ 401	✗ 401	Invalid tokens consistently rejected

Results Summary

The validation confirmed that RBAC enforcement in aerOS works as intended:

- **Continuum Administrators** retain full privileges across all endpoints.
- **aerOS Users** are restricted to read-only operations.
- **Data Product Owners** can both create and delete their own data products and catalog entries, ensuring lifecycle ownership.
- **Vertical Deployers** can both deploy and remove IoT services, providing autonomy in managing their applications.
- Unauthorized requests, expired tokens, and tampered tokens are consistently denied, ensuring robustness against misuse.

Overall, the system demonstrates **least privilege enforcement, clear separation of roles, and auditability** across the continuum.

B. Appendix B. Technical KPIs

aerOS network and compute fabric

KPI 1.1.1 Response time for the orchestration of IoT applications (KVI-1.1)

Table 23: KPI 1.1.1 Response time for the orchestration of IoT applications (KV-1.1)

KPI ID number and partner resp.	**KPI 1.1.1**
KPI Name	**Response time for the orchestration of IoT applications**
Description	This KPI measures the time the orchestration system takes to achieve the target state of the blueprint of the IoT applications
Motivation	Whether achieving the initial state or transitioning states due to external conditions changes, the orchestration system should provide responsiveness for the IoT applications. A less responsive system would hinder the usefulness of such autonomous service for the end user and makes it less reactive to changing conditions.
Target value	<15% baseline → 8.5 s
Prerequisites	aerOS installation ready in the concerned domains. aerOS installation implies here that, at least, self-awareness and self-orchestrator elements are functional in several IEs, that these (IEs) are organized in one (or more) domain(s) and that the HLO is capable of receiving implementation blueprints and allocating computing workloads.
aerOS components (task)	HLO (T3.3), LLO (T3.3), self-awareness (T3.5), self-orchestrator (T3.5)
Evaluation means	The evaluation process leverages the status of Data Fabric service components to monitor the deployment time effectively. This monitoring is crucial for understanding the time taken for various components to become operational. Additionally, the deployment time can be assessed more accurately from the Management Portal, providing a precise measure of response time from the user's perspective. At M24, given the current state of the demonstrator, the deployment time from the Management Portal that excludes the latency introduced by the HLO (High-Level Orchestration) AI service is being measured. This focused approach allows us to gather baseline data on deployment efficiency without the additional complexity of AI processing delays. For D5.6, a more comprehensive measurement mechanism has been implemented. This advanced system encompasses all aspects of the deployment process, including HLO AI latency and other potential delays, ensuring a thorough and accurate evaluation of the deployment time across the entire service framework.

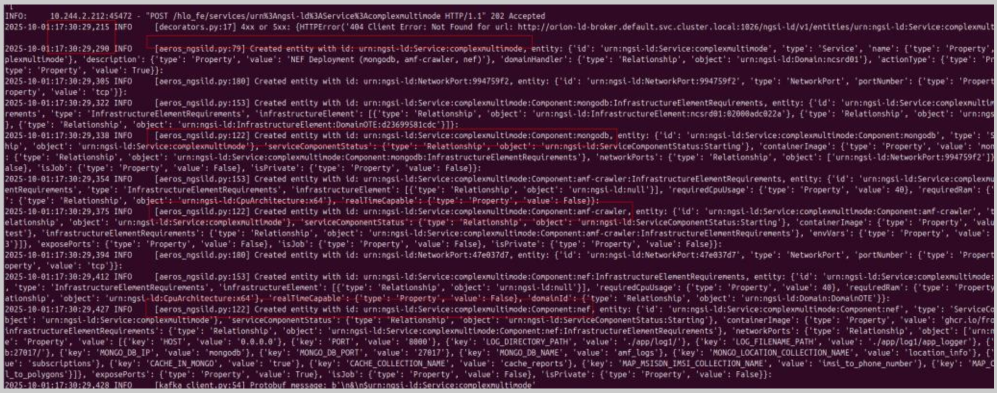
	To assess the response time for the orchestration of IoT applications, multiple deployments were carried out. The deployments in the following table were designed to capture the latency associated with service instantiation and orchestration under real conditions. A set of diversified experiments was performed, reflecting different pilot contexts and configurations, to provide a representative picture of the system's performance. The measured response times are summarized in the following table.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***10 s[3]***	***5 s (160%)***	***4.8 s (166%)***
Outcome elaboration (M38)	The variation in response time for service component orchestration across the continuum is mainly explained by the orchestration mode, the number of components, and whether deployments remain local or span remote domains. In manual orchestration, the deployment is triggered by a direct selection of the instance environment (IE), which involves minimal processing by the orchestrator itself. This explains why manual cases show the lowest latency despite requiring human intervention. By contrast, automated orchestration leaves the full decision-making process to the orchestrator, which must analyze requirements, select resources, and resolve dependencies. This additional processing overhead increases deployment latency. Semi-automated orchestration lies in between, where part of the selection is guided by the user, but the orchestrator still performs optimization tasks, resulting in intermediate delays. The number of service components deployed further affects latency: single-component services can be instantiated quickly, while multi-component applications require additional coordination and dependency resolution between components, which extends the overall deployment time. Finally, the deployment scope plays an important role. If all components are placed within the local domain, latency remains relatively low, but once orchestration extends to remote domains, extra time is consumed by cross-domain communication, synchronization, and network overlay establishment. In summary, the measured latencies reflect the combined effect of these factors. Tools used to get timestamps and measure latencies where k9s and kubectl logs.		

---

[3] The baseline is taken from the worst case in usual IoT applications in the literature [3]

Service Deployment Description	t_hlo_fe	t_HLO_de	t_llo	t_running	Total Latency
NEF-API Service • 1 Service component • Manual Deployment	15:30:55	15:30:56	15:30:56	15:30:57	2 sec
NEF-API Service • 1 Service component • Semi-automated Deployment	15:44:56	15:44:58	15:44:59	15:45:00	4 sec
NEF-API Service • 1 Service component • Automated Deployment	16:08:55	16:08:58	16:08:58	16:08:59	4 sec
NEF application Service • 3 Service components (nef, amf-crawler-mongodb) • Manual Deployment • 1 domain selected	16:37:34	16:37:36	16:37:38	16:37:39	5 sec
NEF application Service Complex • 3 Service components (nef, amf-crawler-mongodb) • Manual/Semi/Automated Deployment (per component) • 2 domains selected • Network overlay requested	17:30:29	17:30:32 (local domain) 17:30:33 (remote domain)	17:30:33 (local domain) 17:30:33 (local domain)	17:30:34 (local domain) 17:30:34 (remote domain)	5 sec
Network Performance application • 4 Service components (iperf-server/client, influxdb, controller) • SemiAutomated Deployment • 1 domain selected	11:23:24	11:23:29	11:23:30	11:23:31	7 sec
Network Performance application • 4 Service components (iperf-server/client, influxdb, controller) • SemiAutomated Deployment • 2 domains selected • Network overlay requested	13:35:08	13:35:12 (local domain) 13:35:13 (remote domain)	13:35:14 (local domain) 13:35:14 (remote domain)	13:35:15 (local domain) 13:35:15 (remote domain)	7 sec
				Average	**4.8 sec**

The following screenshot is from the HLO-FE of the 5th case:

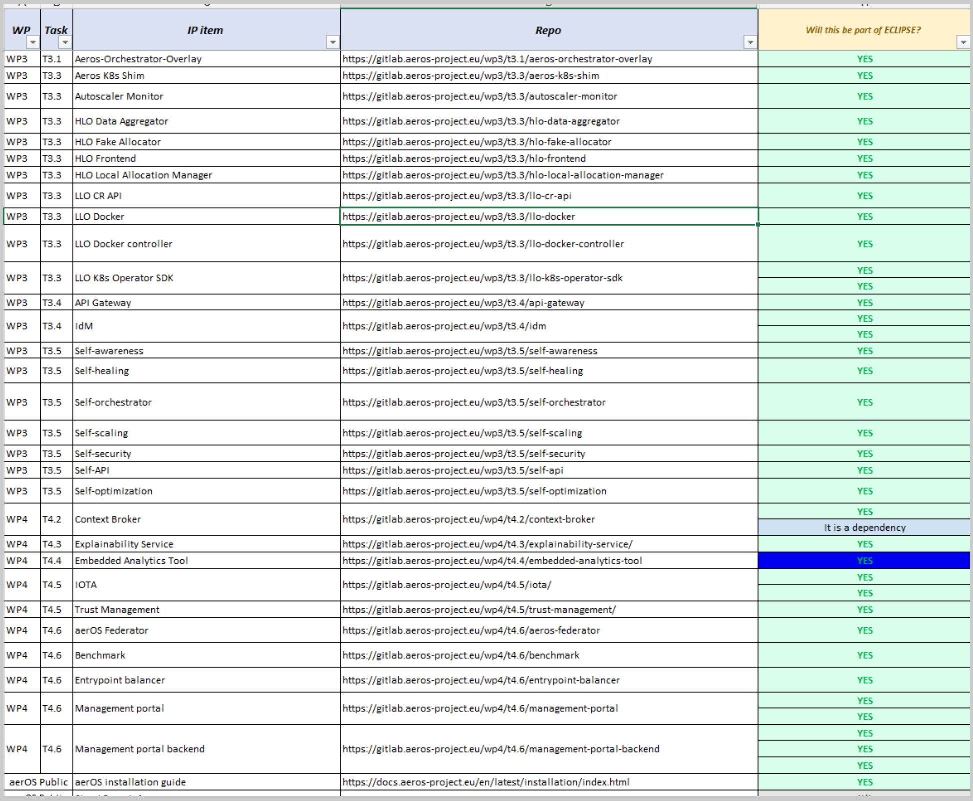


```
INFO:     10.244.2.212:45472 - "POST /hlo_fe/services/urn%3Angsi-ld%3AService%3Acomplexmultimode HTTP/1.1" 202 Accepted
2025-10-01:17:30:29,215 INFO     [decorators.py:17] 4xx or 5xx: {HTTPError('404 Client Error: Not Found for url: http://orion-ld-broker.default.svc.cluster.local:1026/ngsi-ld/v1/entities/urn:ngsi-ld:Service:complexmult

2025-10-01:17:30:29,290 INFO     [aeros_ngsild.py:79] Created entity with id: urn:ngsi-ld:Service:complexmultimode, entity: {'id': 'urn:ngsi-ld:Service:complexmultimode', 'type': 'Service', 'name': {'type': 'Property',
plexmultimode'}, 'description': {'type': 'Property', 'value': 'NEF Deployment (mongodb, amf-crawler, nef)'}, 'domainHandler': {'type': 'Relationship', 'object': 'urn:ngsi-ld:Domain:ncsrd01'}, 'actionType': {'type': 'Pr
type': 'Property', 'value': True}}:
2025-10-01:17:30:29,305 INFO     [aeros_ngsild.py:180] Created entity with id: urn:ngsi-ld:NetworkPort:994759f2, entity: {'id': 'urn:ngsi-ld:NetworkPort:994759f2', 'type': 'NetworkPort', 'portNumber': {'type': 'Propert
roperty', 'value': 'tcp'}}:
2025-10-01:17:30:29,322 INFO     [aeros_ngsild.py:153] Created entity with id: urn:ngsi-ld:Service:complexmultimode:Component:mongodb:InfrastructureElementRequirements, entity: {'id': 'urn:ngsi-ld:Service:complexmultim
rements', 'type': 'InfrastructureElementRequirements', 'infrastructureElement': [{'type': 'Relationship', 'object': 'urn:ngsi-ld:InfrastructureElement:ncsrd01:02000adc022a'}, {'type': 'Relationship', 'object': 'urn:ngs
}, {'type': 'Relationship', 'object': 'urn:ngsi-ld:InfrastructureElement:DomainOTE:d23699581cdc'}]}:
2025-10-01:17:30:29,338 INFO     [aeros_ngsild.py:122] Created entity with id: urn:ngsi-ld:Service:complexmultimode:Component:mongodb, entity: {'id': 'urn:ngsi-ld:Service:complexmultimode:Component:mongodb', 'type': 'S
hip', 'object': 'urn:ngsi-ld:Service:complexmultimode'}, 'serviceComponentStatus': {'type': 'Relationship', 'object': 'urn:ngsi-ld:ServiceComponentStatus:Starting'}, 'containerImage': {'type': 'Property', 'value': 'mon
: {'type': 'Relationship', 'object': 'urn:ngsi-ld:Service:complexmultimode:Component:mongodb:InfrastructureElementRequirements'}, 'networkPorts': {'type': 'Relationship', 'object': ['urn:ngsi-ld:NetworkPort:994759f2']}
alse}, 'isJob': {'type': 'Property', 'value': False}, 'isPrivate': {'type': 'Property', 'value': False}}:
2025-10-01:17:30:29,354 INFO     [aeros_ngsild.py:153] Created entity with id: urn:ngsi-ld:Service:complexmultimode:Component:amf-crawler:InfrastructureElementRequirements, entity: {'id': 'urn:ngsi-ld:Service:complexmu
entRequirements', 'type': 'InfrastructureElementRequirements', 'infrastructureElement': [{'type': 'Relationship', 'object': 'urn:ngsi-ld:null'}], 'requiredCpuUsage': {'type': 'Property', 'value': 40}, 'requiredRam': {'
': {'type': 'Relationship', 'object': 'urn:ngsi-ld:CpuArchitecture:x64'}, 'realTimeCapable': {'type': 'Property', 'value': False}}:
2025-10-01:17:30:29,375 INFO     [aeros_ngsild.py:122] Created entity with id: urn:ngsi-ld:Service:complexmultimode:Component:amf-crawler, entity: {'id': 'urn:ngsi-ld:Service:complexmultimode:Component:amf-crawler', 't
elationship', 'object': 'urn:ngsi-ld:Service:complexmultimode'}, 'serviceComponentStatus': {'type': 'Relationship', 'object': 'urn:ngsi-ld:ServiceComponentStatus:Starting'}, 'containerImage': {'type': 'Property', 'valu
test'}, 'infrastructureElementRequirements': {'type': 'Relationship', 'object': 'urn:ngsi-ld:Service:complexmultimode:Component:amf-crawler:InfrastructureElementRequirements'}, 'envVars': {'type': 'Property', 'value':
3'}]}, 'exposePorts': {'type': 'Property', 'value': False}, 'isJob': {'type': 'Property', 'value': False}, 'isPrivate': {'type': 'Property', 'value': False}}:
2025-10-01:17:30:29,394 INFO     [aeros_ngsild.py:180] Created entity with id: urn:ngsi-ld:NetworkPort:47e037d7, entity: {'id': 'urn:ngsi-ld:NetworkPort:47e037d7', 'type': 'NetworkPort', 'portNumber': {'type': 'Propert
operty', 'value': 'tcp'}}:
2025-10-01:17:30:29,412 INFO     [aeros_ngsild.py:153] Created entity with id: urn:ngsi-ld:Service:complexmultimode:Component:nef:InfrastructureElementRequirements, entity: {'id': 'urn:ngsi-ld:Service:complexmultimode:
, 'type': 'InfrastructureElementRequirements', 'infrastructureElement': [{'type': 'Relationship', 'object': 'urn:ngsi-ld:null'}], 'requiredCpuUsage': {'type': 'Property', 'value': 40}, 'requiredRam': {'type': 'Property
ationship', 'object': 'urn:ngsi-ld:CpuArchitecture:x64'}, 'realTimeCapable': {'type': 'Property', 'value': False}, 'domainId': {'type': 'Relationship', 'object': 'urn:ngsi-ld:Domain:DomainOTE'}}:
2025-10-01:17:30:29,427 INFO     [aeros_ngsild.py:122] Created entity with id: urn:ngsi-ld:Service:complexmultimode:Component:nef, entity: {'id': 'urn:ngsi-ld:Service:complexmultimode:Component:nef', 'type': 'ServiceCo
bject': 'urn:ngsi-ld:Service:complexmultimode'}, 'serviceComponentStatus': {'type': 'Relationship', 'object': 'urn:ngsi-ld:ServiceComponentStatus:Starting'}, 'containerImage': {'type': 'Property', 'value': 'ghcr.io/fro
infrastructureElementRequirements': {'type': 'Relationship', 'object': 'urn:ngsi-ld:Service:complexmultimode:Component:nef:InfrastructureElementRequirements'}, 'networkPorts': {'type': 'Relationship', 'object': ['urn:n
e': 'Property', 'value': [{'key': 'HOST', 'value': '0.0.0.0'}, {'key': 'PORT', 'value': '8000'}, {'key': 'LOG_DIRECTORY_PATH', 'value': './app/log1/'}, {'key': 'LOG_FILENAME_PATH', 'value': './app/log1/app_logger'}, {'
b:27017/'}, {'key': 'MONGO_DB_IP', 'value': 'mongodb'}, {'key': 'MONGO_DB_PORT', 'value': '27017'}, {'key': 'MONGO_DB_NAME', 'value': 'amf_logs'}, {'key': 'MONGO_LOCATION_COLLECTION_NAME', 'value': 'location_info'}, {'
ue': 'subscriptions'}, {'key': 'CACHE_IN_MONGO', 'value': 'true'}, {'key': 'CACHE_COLLECTION_NAME', 'value': 'cache_reports'}, {'key': 'MAP_MSISDN_IMSI_COLLECTION_NAME', 'value': 'imsi_to_phone_number'}, {'key': 'MAP_C
l_to_polygons'}]}, 'exposePorts': {'type': 'Property', 'value': True}, 'isJob': {'type': 'Property', 'value': False}, 'isPrivate': {'type': 'Property', 'value': False}}:
2025-10-01:17:30:29,428 INFO     [kafka_client.py:54] Protobuf message: b'\n&\n$urn:ngsi-ld:Service:complexmultimode'
```

KPI 1.1.2 Open-source components for aerOS to deploy and manage applications spanning the continuum (KVI-1.2)

Table 24: KPI 1.1.2 Open-source components for aerOS to deploy and manage applications spanning the continuum (KVI-1.2)

KPI ID number and partner resp.	**KPI 1.1.2**		
KPI Name	**Open-source components for aerOS to deploy and manage applications spanning the continuum**		
Description	The KPI determines the number of components that aerOS generates and are able to deploy and manage applications in the continuum that have been shared with external communities through open-source contributions.		
Motivation	This KPI is important to measure the impact aerOS has on the technological ecosystem, allowing the support of new technological business models and third-parties exploitation resulting from the project innovative work.		
Target value	3		
Prerequisites	The components are aerOS-created, or aerOS-enhanced, and are ***available for inspection, download, reuse and replication*** by the community outside of the project. A pre-requisite for final acceptance of this KPI is also the completion of the Open-Source Strategy that has been defined during the latest period of the project, and that will be put in place from M24 to M38 of aerOS.		
aerOS components (task)	Any aerOS component subject of being open-source licensed (all WP3-WP4, and T6.4)		
Evaluation means	The number of public repositories related aerOS will be used for the evaluation.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*N/A*	*0 (0%)*	*37 (1000%)*
Outcome elaboration (M38)	A total of 37 aerOS components have been selected for open-source release under the Eclipse Foundation, establishing a complete aerOS ecosystem and validating the KPI on open-source impact and technological exploitation. This open-source release confirms aerOS role as a catalyst for innovation across the continuum computing landscape, enabling cross-domain collaboration and ensuring long-term sustainability through transparent, community-driven development.		



WP	Task	IP item	Repo	Will this be part of ECLIPSE?
WP3	T3.1	Aeros-Orchestrator-Overlay	https://gitlab.aeros-project.eu/wp3/t3.1/aeros-orchestrator-overlay	YES
WP3	T3.3	Aeros K8s Shim	https://gitlab.aeros-project.eu/wp3/t3.3/aeros-k8s-shim	YES
WP3	T3.3	Autoscaler Monitor	https://gitlab.aeros-project.eu/wp3/t3.3/autoscaler-monitor	YES
WP3	T3.3	HLO Data Aggregator	https://gitlab.aeros-project.eu/wp3/t3.3/hlo-data-aggregator	YES
WP3	T3.3	HLO Fake Allocator	https://gitlab.aeros-project.eu/wp3/t3.3/hlo-fake-allocator	YES
WP3	T3.3	HLO Frontend	https://gitlab.aeros-project.eu/wp3/t3.3/hlo-frontend	YES
WP3	T3.3	HLO Local Allocation Manager	https://gitlab.aeros-project.eu/wp3/t3.3/hlo-local-allocation-manager	YES
WP3	T3.3	LLO CR API	https://gitlab.aeros-project.eu/wp3/t3.3/llo-cr-api	YES
WP3	T3.3	LLO Docker	https://gitlab.aeros-project.eu/wp3/t3.3/llo-docker	YES
WP3	T3.3	LLO Docker controller	https://gitlab.aeros-project.eu/wp3/t3.3/llo-docker-controller	YES
WP3	T3.3	LLO K8s Operator SDK	https://gitlab.aeros-project.eu/wp3/t3.3/llo-k8s-operator-sdk	YES YES
WP3	T3.4	API Gateway	https://gitlab.aeros-project.eu/wp3/t3.4/api-gateway	YES
WP3	T3.4	IdM	https://gitlab.aeros-project.eu/wp3/t3.4/idm	YES YES
WP3	T3.5	Self-awareness	https://gitlab.aeros-project.eu/wp3/t3.5/self-awareness	YES
WP3	T3.5	Self-healing	https://gitlab.aeros-project.eu/wp3/t3.5/self-healing	YES
WP3	T3.5	Self-orchestrator	https://gitlab.aeros-project.eu/wp3/t3.5/self-orchestrator	YES
WP3	T3.5	Self-scaling	https://gitlab.aeros-project.eu/wp3/t3.5/self-scaling	YES
WP3	T3.5	Self-security	https://gitlab.aeros-project.eu/wp3/t3.5/self-security	YES
WP3	T3.5	Self-API	https://gitlab.aeros-project.eu/wp3/t3.5/self-api	YES
WP3	T3.5	Self-optimization	https://gitlab.aeros-project.eu/wp3/t3.5/self-optimization	YES
WP4	T4.2	Context Broker	https://gitlab.aeros-project.eu/wp4/t4.2/context-broker	YES It is a dependency
WP4	T4.3	Explainability Service	https://gitlab.aeros-project.eu/wp4/t4.3/explainability-service/	YES
WP4	T4.4	Embedded Analytics Tool	https://gitlab.aeros-project.eu/wp4/t4.4/embedded-analytics-tool	YES
WP4	T4.5	IOTA	https://gitlab.aeros-project.eu/wp4/t4.5/iota/	YES YES
WP4	T4.5	Trust Management	https://gitlab.aeros-project.eu/wp4/t4.5/trust-management/	YES
WP4	T4.6	aerOS Federator	https://gitlab.aeros-project.eu/wp4/t4.6/aeros-federator	YES
WP4	T4.6	Benchmark	https://gitlab.aeros-project.eu/wp4/t4.6/benchmark	YES
WP4	T4.6	Entrypoint balancer	https://gitlab.aeros-project.eu/wp4/t4.6/entrypoint-balancer	YES
WP4	T4.6	Management portal	https://gitlab.aeros-project.eu/wp4/t4.6/management-portal	YES YES
WP4	T4.6	Management portal backend	https://gitlab.aeros-project.eu/wp4/t4.6/management-portal-backend	YES YES YES
aerOS Public		aerOS installation guide	https://docs.aeros-project.eu/en/latest/installation/index.html	YES

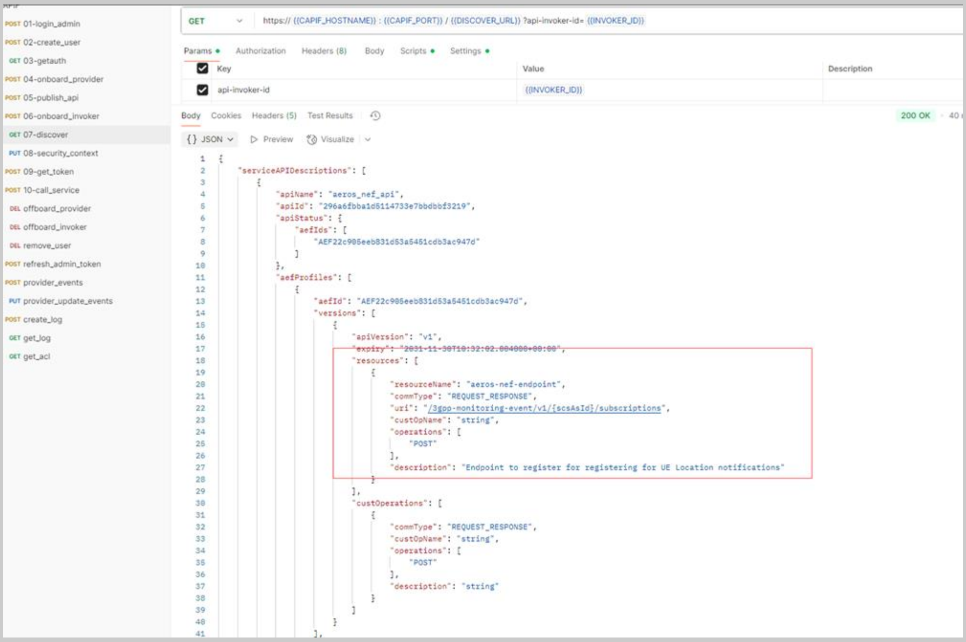
KPI 1.1.3 Usage of 5G native APIs (3GPP NEF and CAPIF) (KVI-1.3)

Table 25: KPI 1.1.3 Usage of 5G native APIs (3GPP NEF and SEAL) (KVI-1.3)

KPI ID number and partner resp.	**KPI 1.1.3**
KPI Name	**Usage of 5G native APIs (3GPP NEF and CAPIF)**
Description	5G Native APIs that have been specified by 3GPP allows the tight integration of services and applications in order to improve their performance and features, such as by retrieving information of the QoS level of the network and the location of the user.
Motivation	The use of 5G native APIs can significantly enhance the performance of a service/application, providing additional context information and network awareness. Therefore, assessing the use of native 5G APIs as a KPI is important because it denotes the disruptive innovation of the developed services and applications.
Target value	>50% aerOS scenarios using 5G network
Prerequisites	Functional aerOS domain and aerOS APIs exposed
aerOS components (task)	Ingress (T3.1), TLS (T3.1), OpenAPI (T3.2), HLO (T3.3), API Gateway (T3.4), Context Broker (T4.2)

<table>
<tr><td>Evaluation means</td><td colspan="3">Out of all the aerOS scenarios, the following two are going to use cellular network:
1. Pilot 4 – Predictive maintenance (with two scenarios)
2. Pilot 5 – Smart buildings (with two scenarios)
As long as any two of those four scenarios show that the OpenCAPIF is integrated in, the KPI will be considered as fulfilled.
To carry out the evaluation, there are three options: (i) reporting tools of OpenCAPIF will be used; (ii) exported report on discovered (aerOS) APIs; and (iii) POSTMAN endpoints with OpenCAPIF acting as consumer, getting all aerOS registered APIs.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td>N/A</td><td>1/4 scenarios deployed, but not integrated (50%)</td><td>2/4 scenarios deployed, integrated, and validated
(100% achieved KPI)</td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">The usage of 5G native APIs (3GPP NEF and CAPIF) within aerOS has been fully validated through the activities conducted primarily in Pilot 5, where aerOS capability to host and orchestrate 5G-related VNFs and to expose them securely through standardized 3GPP interfaces was demonstrated.
As already presented in Deliverable D5.4 – Use cases deployment and implementation (2), aerOS successfully integrated the Open5GS 5G core and deployed both the Network Exposure Function (NEF) and the User Plane Function (UPF) as VNFs over the continuum. Two key NEF functionalities were validated: Event Monitoring for UE location and AsSessionWithQoS. Both were onboarded as aerOS services using TOSCA descriptors and orchestrated via the aerOS HLO, ensuring full lifecycle management from deployment to exposure through Swagger APIs. These services operated in conjunction with UERANSIM, validating end-to-end connectivity and functionality. This action has also been described in P5-BP2-VA1.
To enhance interoperability and discoverability, an OpenCAPIF instance was deployed and integrated with aerOS. Through this integration, the NEF APIs exposed from aerOS were securely registered and became discoverable by external invokers via the CAPIF registry. The process included the registration of the aerOS user and API provider within OpenCAPIF, the creation of secure certificates and keys, and the publication of the NEF notification endpoints. Postman collections were used to verify these steps, confirming that the NEF APIs deployed on aerOS could be discovered and invoked securely via OpenCAPIF, in line with 3GPP TS 23.222 specifications.
All necessary validation tools were used — including k9s for service-level verification, Postman for API interactions, and OpenCAPIF reporting tools to verify discovery operations. The combined evidence, reported in D5.4 and extended in the current deliverable (D5.6), demonstrates that the KPI is fully achieved: aerOS manages and exposes 5G-native APIs (NEF) through standardized CAPIF interfaces, ensuring secure discoverability and controlled access to 5G network functions deployed at the edge. Given the extensive nature of the evidence (deployment screenshots, NEF and UPF logs, Postman traces, and CAPIF discovery results), only representative figures are included in this deliverable. The complete validation dataset is hosted in D5.4 and in this deliverable also (D5.6) in the sections connected with Pilot 5 “Setup, Development, Integration and Validation activities”. The following two indicative screenshots demonstrate the deployment and interaction of NEF functionalities (Event Monitoring and QoS) over aerOS, and their secure exposure via OpenCAPIF for</td></tr>
</table>

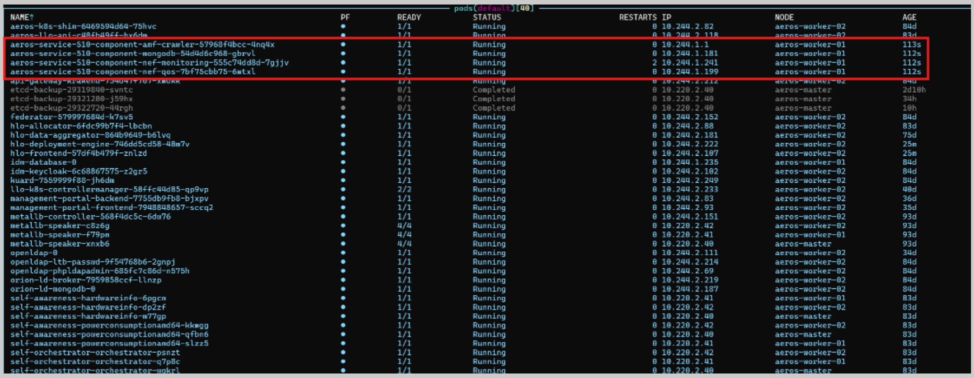
standardized discovery and controlled external access. First figure shows the NEF Location Event Monitoring endpoint discovery via OpenCAPIF and the second shows the 2 NEF APIs integrates running as aerOS hosted services.



GET https:// {{CAPIF_HOSTNAME}} : {{CAPIF_PORT}} / {{DISCOVER_URL}} ?api-invoker-id= {{INVOKER_ID}}

Key	Value	Description
☑ api-invoker-id	{{INVOKER_ID}}	

```
{
    "serviceAPIDescriptions": [
        {
            "apiName": "aeros_nef_api",
            "apiId": "296a6fbba1d5114733e7bbdbbf3219",
            "apiStatus": {
                "aefIds": [
                    "AEF22c985eeb831d53a5451cdb3ac947d"
                ]
            },
            "aefProfiles": [
                {
                    "aefId": "AEF22c985eeb831d53a5451cdb3ac947d",
                    "versions": [
                        {
                            "apiVersion": "v1",
                            "expiry": "2031-11-30T10:32:02.004000+00:00",
                            "resources": [
                                {
                                    "resourceName": "aeros-nef-endpoint",
                                    "commType": "REQUEST_RESPONSE",
                                    "uri": "/3gpp-monitoring-event/v1/{scsAsId}/subscriptions",
                                    "custOpName": "string",
                                    "operations": [
                                        "POST"
                                    ],
                                    "description": "Endpoint to register for registering for UE Location notifications"
                                }
                            ],
                            "custOperations": [
                                {
                                    "commType": "REQUEST_RESPONSE",
                                    "custOpName": "string",
                                    "operations": [
                                        "POST"
                                    ],
                                    "description": "string"
                                }
                            ]
                        }
                    ],
```



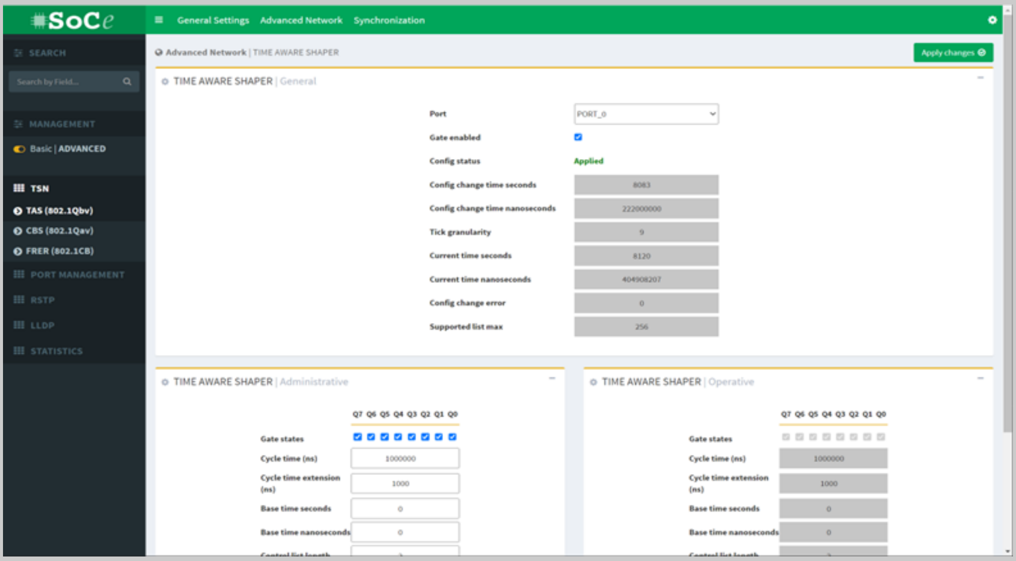
```
                                              pods(default)[40]
NAME↑                                           PF  READY  STATUS     RESTARTS IP            NODE             AGE
aeros-k8s-shim-6469594d64-75hvc                 ●   1/1    Running           0 10.244.2.82   aeros-worker-02  84d
aeros-llo-api-c48fb49ff-bx6dm                   ●   1/1    Running           0 10.244.2.118  aeros-worker-02  83d
aeros-service-510-component-amf-crawler-57968f4bcc-4nq4x  ●   1/1    Running           0 10.244.1.1    aeros-worker-01  113s
aeros-service-510-component-mongodb-54d4d6c968-gbrvl  ●   1/1    Running           0 10.244.1.181  aeros-worker-01  112s
aeros-service-510-component-nef-monitoring-555c74dd8d-7gjjv  ●   1/1    Running           2 10.244.1.241  aeros-worker-01  112s
aeros-service-510-component-nef-qos-7bf75cbb75-6wtxl  ●   1/1    Running           0 10.244.1.199  aeros-worker-01  112s
api-gateway-krakend-[illegible]                 ●   1/1    Running           0 10.244.2.212  aeros-worker-02  84d
etcd-backup-29319840-svntc                      ●   0/1    Completed         0 10.220.2.40   aeros-master     2d10h
etcd-backup-29321280-j59hx                      ●   0/1    Completed         0 10.220.2.40   aeros-master     34h
etcd-backup-29322720-44rgh                      ●   0/1    Completed         0 10.220.2.40   aeros-master     10h
federator-579997684d-k7sv5                      ●   1/1    Running           0 10.244.2.152  aeros-worker-02  84d
hlo-allocator-6fdc99b7f4-lbcbn                  ●   1/1    Running           0 10.244.2.88   aeros-worker-02  83d
hlo-data-aggregator-864b9649-b6lvq              ●   1/1    Running           0 10.244.2.181  aeros-worker-02  75d
hlo-deployment-engine-746dd5cd58-48m7v          ●   1/1    Running           0 10.244.2.222  aeros-worker-02  25m
hlo-frontend-57df4b479f-znlzd                   ●   1/1    Running           0 10.244.2.107  aeros-worker-02  25m
idm-database-0                                  ●   1/1    Running           0 10.244.1.235  aeros-worker-01  84d
idm-keycloak-6c68867575-z2gr5                   ●   1/1    Running           0 10.244.2.102  aeros-worker-02  84d
kuard-7559999f88-jh6dm                          ●   1/1    Running           0 10.244.2.249  aeros-worker-02  84d
llo-k8s-controllermanager-58ffc44d85-qp9vp      ●   2/2    Running           0 10.244.2.233  aeros-worker-02  40d
management-portal-backend-7755db9fb8-bjxpv      ●   1/1    Running           0 10.244.2.83   aeros-worker-02  36d
management-portal-frontend-7948848657-sccq2     ●   1/1    Running           0 10.244.2.93   aeros-worker-02  35d
metallb-controller-568f4dc5c-6dw76              ●   1/1    Running           0 10.244.2.151  aeros-worker-02  93d
metallb-speaker-c8z6g                           ●   4/4    Running           0 10.220.2.42   aeros-worker-02  93d
metallb-speaker-f79pm                           ●   4/4    Running           0 10.220.2.41   aeros-worker-01  93d
metallb-speaker-xnxb6                           ●   4/4    Running           0 10.220.2.40   aeros-master     93d
openldap-0                                      ●   1/1    Running           0 10.244.2.111  aeros-worker-02  34d
openldap-ltb-passwd-9f54768b6-2gnpj             ●   1/1    Running           0 10.244.2.214  aeros-worker-02  84d
openldap-phpldapadmin-685fc7c86d-n575h          ●   1/1    Running           0 10.244.2.69   aeros-worker-02  84d
orion-ld-broker-7959858ccf-llnzp                ●   1/1    Running           0 10.244.2.219  aeros-worker-02  84d
orion-ld-mongodb-0                              ●   1/1    Running           0 10.244.2.187  aeros-worker-02  84d
self-awareness-hardwareinfo-6pgcm               ●   1/1    Running           0 10.220.2.41   aeros-worker-01  83d
self-awareness-hardwareinfo-dp2zf               ●   1/1    Running           0 10.220.2.42   aeros-worker-02  83d
self-awareness-hardwareinfo-m77gp               ●   1/1    Running           0 10.220.2.40   aeros-master     83d
self-awareness-powerconsumptionamd64-kkwgg      ●   1/1    Running           0 10.220.2.42   aeros-worker-02  83d
self-awareness-powerconsumptionamd64-qfbn6      ●   1/1    Running           0 10.220.2.40   aeros-master     83d
self-awareness-powerconsumptionamd64-slzz5      ●   1/1    Running           0 10.220.2.41   aeros-worker-01  83d
self-orchestrator-orchestrator-psnzt            ●   1/1    Running           0 10.220.2.42   aeros-worker-02  83d
self-orchestrator-orchestrator-q7p8c            ●   1/1    Running           0 10.220.2.41   aeros-worker-01  83d
self-orchestrator-orchestrator-wgkrl            ●   1/1    Running           0 10.220.2.40   aeros-master     83d
```

KPI 1.1.4 Usage of TSN (KVI-1.4)

Table 26: KPI 1.1.4 Usage of TSN (KVI-1.4)

KPI ID number and partner resp.	**KPI 1.1.4**
KPI Name	**Usage of TSN**
Description	This KPI measures the adoption rate of Time-Sensitive Networking (TSN) in more than 50% of the applicable scenarios within the project (as per Amendment #2). TSN is a set of standards designed to improve the reliability, latency, and synchronization of standard Ethernet networks. The goal is to quantify the extent to which TSN is being utilized in scenarios where real-time, deterministic communication is critical.
Motivation	The integration of TSN is crucial for scenarios that demand high levels of network determinism and reliability, such as in industrial automation, real-time control systems,

	and applications requiring precise synchronization. By targeting a 20% adoption rate in relevant scenarios, this KPI encourages the advancement of network infrastructure towards more robust, latency-sensitive, and synchronized communication capabilities. This, in turn, supports the overall efficiency, safety, and performance of the systems relying on aerOS.		
Target value	>= 20% scenarios		
Prerequisites	To effectively evaluate the usage of TSN, it is essential to have a TSN-enabled infrastructure in place. This includes ensuring that the network infrastructure is equipped with TSN-compatible switches, routers, or other network devices. Additionally, it is crucial to have TSN-aware applications that can fully utilize TSN's capabilities.		
aerOS components (task)	Ingress (T3.1), TLS (T3.1)		
Evaluation means	The evaluation of this KPI aims to measure the number of scenarios where TSN is deployed. The process involves identifying and counting scenarios in which TSN-capable networking hardware such as switches or routers is utilized. Additionally, a pivot scenario that highlights successful TSN integration will be examined. This scenario will provide valuable insights into challenges faced, benefits gained, and lessons learned.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*N/A*	*1/5 pilots (100%)*	*20% - 1/5 pilots (100% accomplishment)*
Outcome elaboration (M38)	At the start of the project, we discussed amongst 5 pilot use cases and respective scenarios regarding usability of TSN. At that time the project consortium decided to discuss TSN integration within their pilots after they start with deployment. As the complexity in the individual pilots grew, only one key scenario in Pilot 1 lead by Siemens continued the integration of TSN with aerOS technologies and the other pilots saw no real need of doing so, and the efforts to achieve this TSN integration in their pilots had to be put elsewhere. Nevertheless, the replication or transfer of TSN integration from Siemens pilot scenario to others would be possible but cannot be integrated in those pilots due to their integration priorities during the project duration. Therefore, the project consortium requested to change the following wording in the GA amendment, which was approved by the Project Officer: *"KVI-2.2: Usage of TSN in at least 50% of the scenarios." to "KVI-2.2: Usage of TSN in at least 20% of the pilots."* There is no impact of not using TSN in other scenarios as the integration of TSN in other scenarios can be easily replicated. So, depending on the need of the other pilots for showcasing aerOS technologies, Siemens pilot scenario will show the pathway to achieve it. Following is the snapshot of TSN setup in Siemens scenario of Pilot 1.		



Advanced Network | TIME AWARE SHAPER

TIME AWARE SHAPER | General

Port: PORT_0
Gate enabled: ☑
Config status: Applied
Config change time seconds: 8083
Config change time nanoseconds: 222000000
Tick granularity: 9
Current time seconds: 8120
Current time nanoseconds: 404908207
Config change error: 0
Supported list max: 256

TIME AWARE SHAPER | Administrative

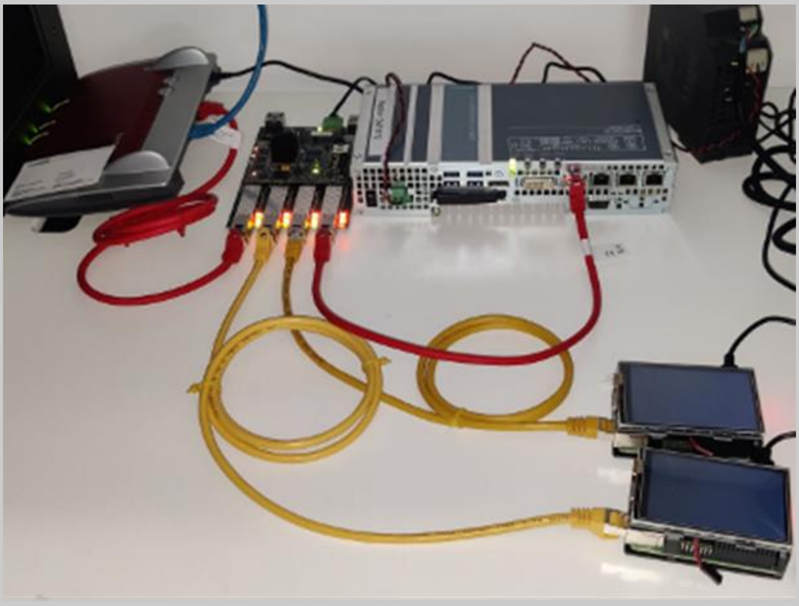
	Q7	Q6	Q5	Q4	Q3	Q2	Q1	Q0
Gate states	☑	☑	☑	☑	☑	☑	☑	☑

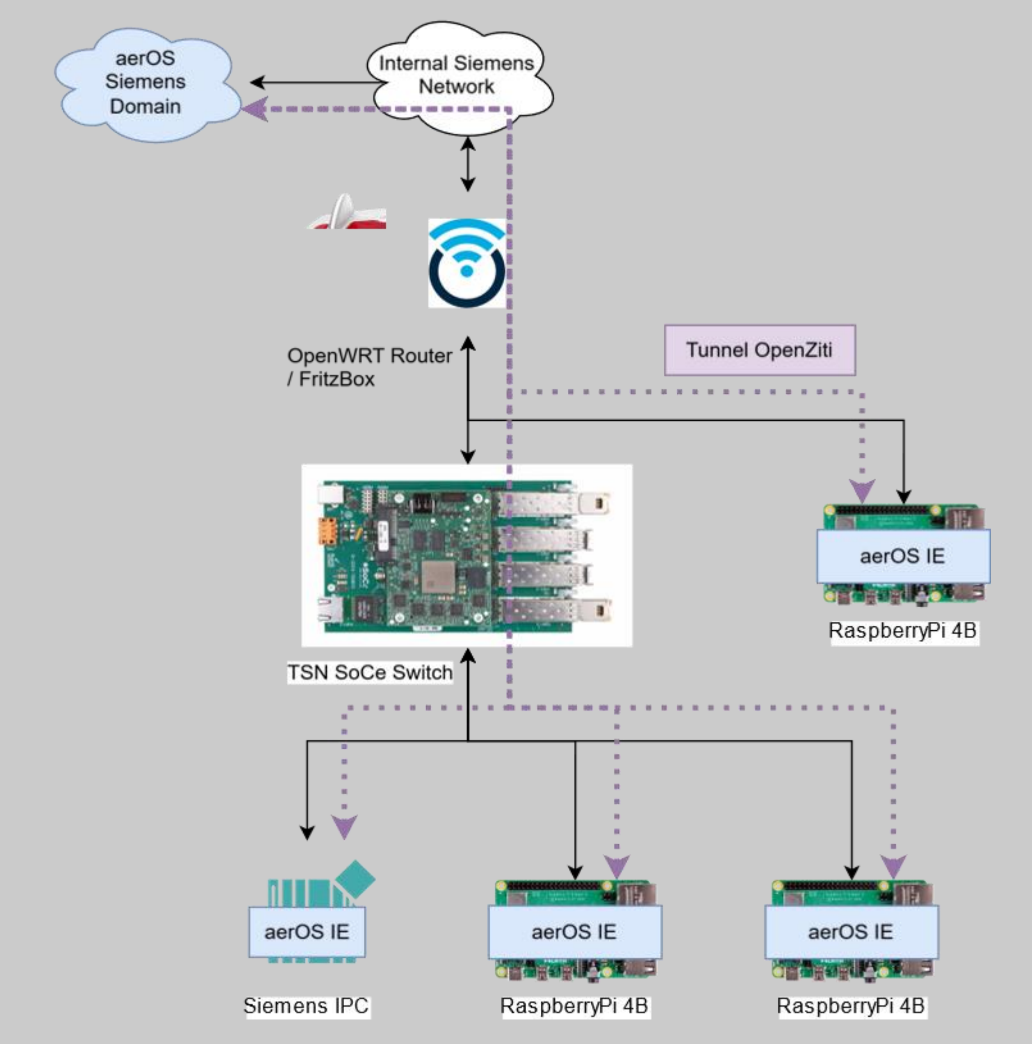
Cycle time (ns): 1000000
Cycle time extension (ns): 1000
Base time seconds: 0
Base time nanoseconds: 0

TIME AWARE SHAPER | Operative

	Q7	Q6	Q5	Q4	Q3	Q2	Q1	Q0
Gate states	☑	☑	☑	☑	☑	☑	☑	☑

Cycle time (ns): 1000000
Cycle time extension (ns): 1000
Base time seconds: 0
Base time nanoseconds: 0



KPI 1.1.5 Number of old equipment units turned on actionable aerOS nodes

Table 27:KPI 1.1.5 Number of old equipment units turned on actionable aerOS nodes

KPI ID number and partner resp.	**KPI 1.1.5**
KPI Name	**Number of old equipment units turned on actionable aerOS nodes**
Description	Devices that are incorporated into the aerOS continuum enabled by the Meta-OS
Motivation	The operation of high-performance algorithms and highly efficient data transaction mechanisms depends on edge, IoT and cloud devices to orchestrate effectively the different services of the hyper distributed application workflows.
Target value	20
Prerequisites	aerOS self-components installed in an IE, and the IE integrated in a domain providing information and being able to accept workloads.

aerOS components (task)	The minimum aerOS core services, namely, LLO (T3.3), Self-* (T3.5), Context Broker (T4.2)		
Evaluation means	KPI leader (INNO) addressed the different aerOS pilots about the old equipment units that have become in new aerOS nodes. An internal description per pilot, including technical specifications, will be included within this table. The evidence of their use will be presented in D5.4.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***46 identified. The 6 IEs from Pilot5 up-and-running (30%)***	***67 devices (335%)***
Outcome elaboration (M38)	The number of devices varies significantly between pilots: ***Pilot 1**: 19 devices* • *SIPBB (Scenario1) –* ***5 devices****:* Quality control, Manual workstation, Smart conveyor, AGV, PCB THT-Soldering, SMC AMS). This is the hardware needed to integrate into aerOS and whose data must be collected. • INNOVALIA (Scenario2) **- 0 devices**: There is no old equipment expected now to be turn on thanks to aerOS within the scope of the project • SIEMENS (Scenario3) **- 9 devices**: 4 existing AGVs and 1 industrial PC have been transformed into aerOS devices. Regarding the testbed, 3 existing Raspberry Pi's and 1 TSN switch were also converted into aerOS devices. • MADE&POLIMI (Scenario4) – **3 devices**: MADE relies on their pre-existing on premise server. POLIMI relies on an existing Industrial PC and onboard PC of the AGV (the AGV was recently refurbished – addition of an SSD, new batteries, and upgrade from ubuntu 18.04). Considering AGV as old equipment there are 3.  ***Pilot 2: 45 devices.*** Since the Cloud is public and out of context, is not considered. Within the Edge, decommissioned servers from another project are used. 1x router node with following specification: 2 processors 12 core 2.3 GHz, 128 GB RAM, 2 x SSD Boot Disk 1TB 1x control plane node with following specification: 2 processors 12 core 2.3 GHz, 256 GB RAM, 2 x SSD Boot Disk 1TB 37x compute nodes with following specification: 2 processors 12 core 2.3 GHz, 256 GB RAM, 2 x SSD Boot Disk 120GB		

	3x storage nodes with following specification: 2 processors 12 core 2.3 GHz, 256 GB RAM, 2 x SSD Boot Disk 120GB 1x network switch: Mellanox MSN3420 2x network switch: Cisco Nexus 3048TP ***Pilot 3: 1 device.*** During the project, both the edge node and cloud node were procured to support the system architecture. The master node was repurposed from a previous project: Lenovo ThinkPad T14s Gen 2i with the following specifications: Intel Core i7-1165G7 processor, 16 GB RAM, integrated Mesa Intel Xe Graphics (TGL GT2), 1 Gbit/s Ethernet connectivity, 512 GB SSD, running Debian GNU/Linux 12 (Bookworm)  ***Pilot 4 - 0 devices***: All the hardware equipment for Pilot 4 has been procured as new, so there is no equipment turned on thanks to aerOS. ***Pilot 5 – 4 devices:*** 3 AAEON UP-CHT01 boards (Up-boards) used as IOT GWs (9 years old based on the age of the *processor* - Intel Atom x5-Z8350 @ 1.44 GHz) and 1 HP ProLiant DL380 Gen10 server used as an ESXi host to spawn VMs (8 years old based on the age of the *processor* - Intel Xeon Gold 6152 @ 2.11 GHz) 

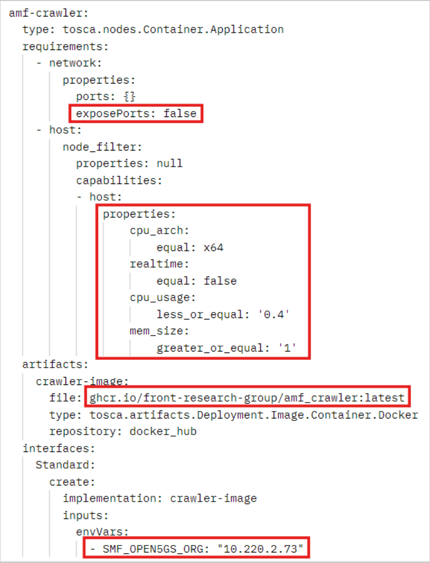
KPI 1.1.6 Consistency of deployment compared to app blueprints

Table 28: KPI 1.1.6 Consistency of deployment compared to app blueprints

KPI ID number and partner resp.	**KPI 1.1.6**
KPI Name	**Consistency of deployment compared to app blueprints**
Description	This KPI measures how consistent the consecutive deployments of the application compared its specified blueprint (TOSCA).
Motivation	The orchestration system is an autonomous system. It is important that this system keeps consistent its automatic deployments with respect to the blueprint and doesn't require manual oversight.

Target value	>95%		
Prerequisites	aerOS installation ready in the concerned domains		
aerOS components (task)	HLO (T3.3), LLO (T3.3), Management Portal (T4.6).		
Evaluation means	Manual test of the applications by pilots and observability tools, such as K9s, for deployments verification. For now, only manual tests have been done on the application mid-review demonstration. With continuous integration of the pilots, more data will be collected, and more observability tools will be integrated as part of the test.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***100% (105% of the target value – 95%)***	***100% (105% of the target value – 95%)***
Outcome elaboration (M38)	In all deployments, Infrastructure Element (IE) selection respected the mode defined in the blueprint: in manual cases, specific IEs were directly chosen; in semi-automated cases, user constraints guided the orchestrator's choice; and in automated cases, the orchestrator evaluated availability and host capabilities (CPU architecture, realtime support, memory, CPU usage limits, energy or domain constraints) before assigning the component. At runtime, each service component was instantiated from the declared container image and repository, with the specified environment variables correctly injected. Networking behavior matched blueprint requirements: ports were exposed or kept internal as defined, and where overlays were requested, isolated networks were created to link components while preserving external exposure rules. Multi-component and multi-domain deployments were handled consistently, with inter-component connectivity, overlays, and policies provisioned exactly as described. Indicatively, we include below screenshots from a selected deployment—showing the TOSCA specification, pod descriptions, and overlay runtime configurations—to demonstrate full compliance with the blueprint. We have a service consisting of three service components and we ask for an overlay to provide a dedicated isolated network for them. The first (mongodb) asks to be deployed in one of 3 IEs and have a network port but not exposed out of the overlay, the second (amf-crawler) set some IEs requirements and lets the continuum decide best placement (AI predictive process inlcuded) and also has an environment variable for configuring mongdb access and the third (NEF AP) sets IE requirements but also limits candidate IEs in one domain (OTE domain), additionally it requires to expose an access port for its API and sets a lot of environment variables.		

```yaml
tosca_definitions_version: tosca_simple_yaml_1_3
description: NEF Deployment (mongodb, amf-crawler, nef)
serviceOverlay: true

node_templates:
  mongodb:
    type: tosca.nodes.Container.Application
    requirements:
      - network:
          properties:
            ports:
              api:
                properties:
                  protocol: [tcp]
                  source: 27017
            exposePorts: false
      - host:
          node_filter:
            properties:
              id:
                - "urn:ngsi-ld:InfrastructureElement:ncsrd01:02000adc022a"
                - "urn:ngsi-ld:InfrastructureElement:ncsrd01:02000adc0228"
                - "urn:ngsi-ld:InfrastructureElement:DomainOTE:d23699581cdc"
    artifacts:
      mongodb-image:
        file: mongo:latest
        type: tosca.artifacts.Deployment.Image.Container.Docker
        repository: docker_hub
    interfaces:
      Standard:
        create:
          implementation: mongodb-image
          inputs:
            envVars: []
```

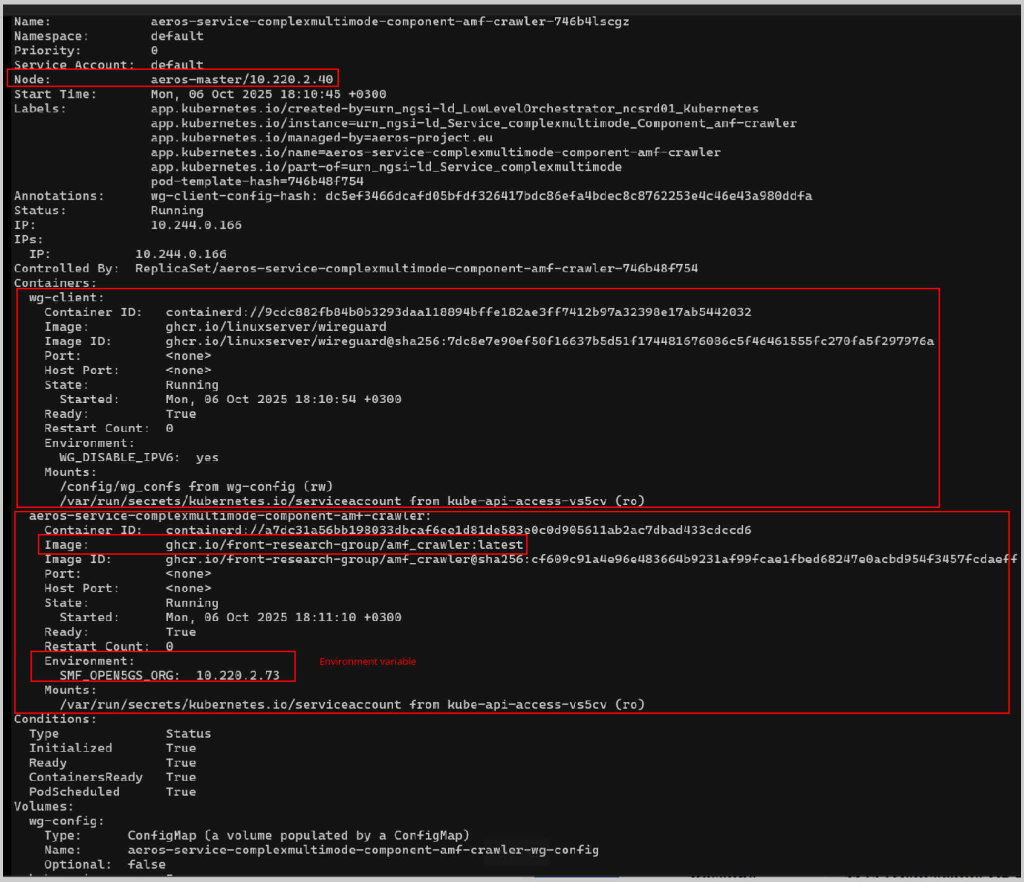


```yaml
amf-crawler:
  type: tosca.nodes.Container.Application
  requirements:
    - network:
        properties:
          ports: {}
          exposePorts: false
    - host:
        node_filter:
          properties: null
          capabilities:
          - host:
              properties:
                  cpu_arch:
                      equal: x64
                  realtime:
                      equal: false
                  cpu_usage:
                      less_or_equal: '0.4'
                  mem_size:
                      greater_or_equal: '1'
  artifacts:
    crawler-image:
      file: ghcr.io/front-research-group/amf_crawler:latest
      type: tosca.artifacts.Deployment.Image.Container.Docker
      repository: docker_hub
  interfaces:
    Standard:
      create:
        implementation: crawler-image
        inputs:
          envVars:
          - SMF_OPEN5GS_ORG: "10.220.2.73"
```

```yaml
nef:
  type: tosca.nodes.Container.Application
  requirements:
    - network:
        properties:
          ports:
            api:
              properties:
                protocol: [tcp]
                source: 8000
          exposePorts: true
    - host:
        node_filter:
          properties: null
          capabilities:
          - host:
              properties:
                  cpu_arch:
                      equal: x64
                  realtime:
                      equal: false
                  cpu_usage:
                      less_or_equal: '0.4'
                  mem_size:
                      greater_or_equal: '1'
                  domain_id:
                      equal: urn:ngsi-ld:Domain:DomainOTE
  artifacts:
    nef-image:
      file: ghcr.io/front-research-group/monitoringeventapi:latest
      type: tosca.artifacts.Deployment.Image.Container.Docker
      repository: docker_hub
  interfaces:
    Standard:
      create:
        implementation: nef-image
        inputs:
          envVars:
            - HOST: "0.0.0.0"
            - PORT: "8000"
            - LOG_DIRECTORY_PATH: "./app/log1/"
            - LOG_FILENAME_PATH: "./app/log1/app_logger"
            - MONGO_DB_URI: "mongodb://mongodb:27017/"
            - MONGO_DB_IP: "mongodb"
            - MONGO_DB_PORT: "27017"
            - MONGO_DB_NAME: "amf_logs"
            - MONGO_LOCATION_COLLECTION_NAME: "location_info"
            - MONGO_SUBSCRIPTION_COLLECTION_NAME: "subscriptions"
            - CACHE_IN_MONGO: "true"
            - CACHE_COLLECTION_NAME: "cache_reports"
            - MAP_MSISDN_IMSI_COLLECTION_NAME: "imsi_to_phone_number"
            - MAP_CELLID_TO_POLYGON_COLLECTION_NAME: "cell_to_polygons"
```

So now we expect to validate these requirements and do so by displaying the services deployment to be accurate (k9s) and respecting all TOSCA descriptions (pod description, and Orion-LD hosting IE capabilities) and have a network overlay for dedicated and isolated connectivity. The following screenshots provide evidence of these.

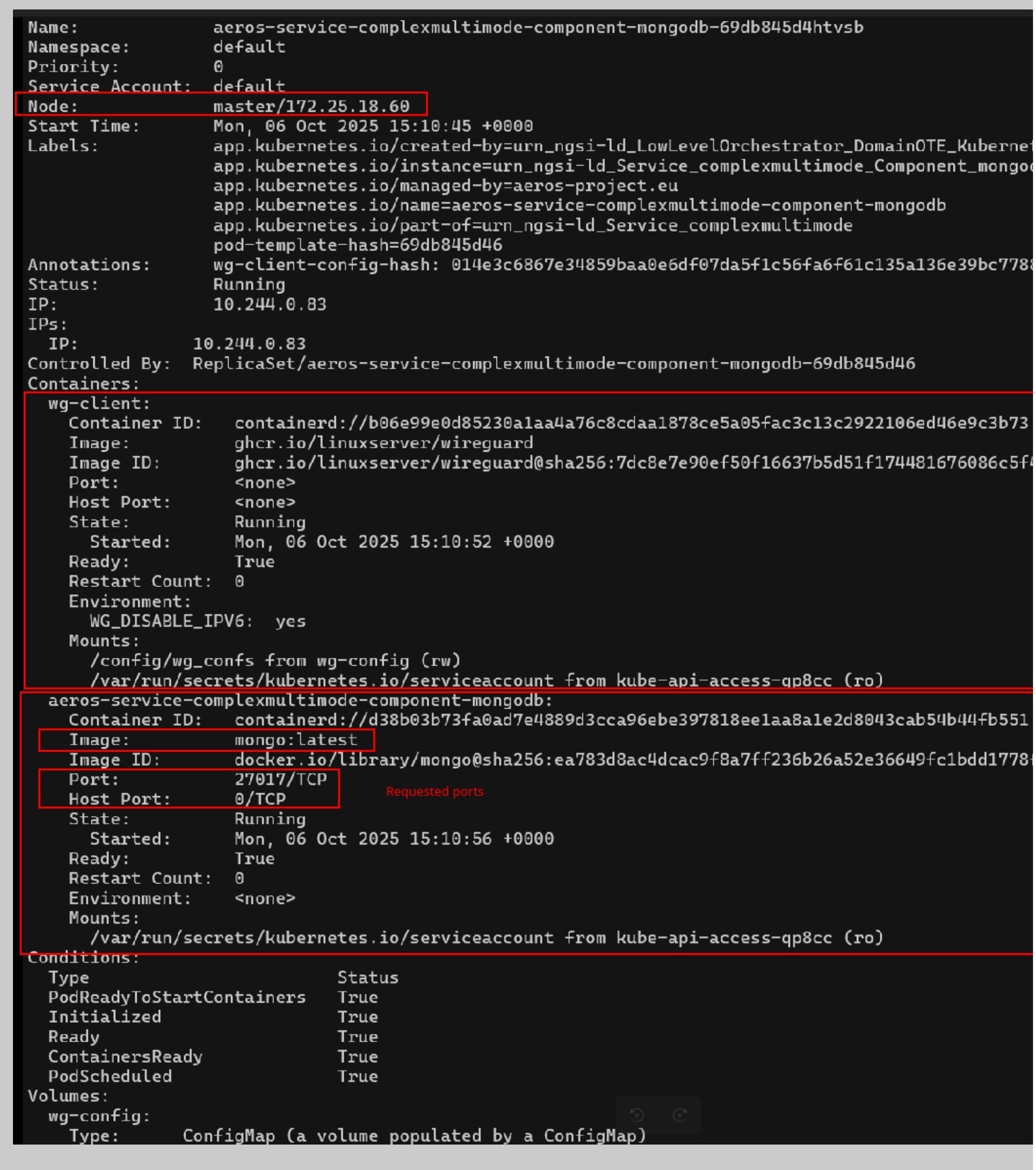
- *amf_crawler*, service component placement (port created, environment variable created, image selected, wg client sidecar created) and compliance of IE selected (one out of 3 indicated)



```
Name:               aeros-service-complexmultimode-component-amf-crawler-746b4lscgz
Namespace:          default
Priority:           0
Service Account:    default
Node:               aeros-master/10.220.2.40
Start Time:         Mon, 06 Oct 2025 18:10:45 +0300
Labels:             app.kubernetes.io/created-by=urn_ngsi-ld_LowLevelOrchestrator_ncsrd01_Kubernetes
                    app.kubernetes.io/instance=urn_ngsi-ld_Service_complexmultimode_Component_amf-crawler
                    app.kubernetes.io/managed-by=aeros-project.eu
                    app.kubernetes.io/name=aeros-service-complexmultimode-component-amf-crawler
                    app.kubernetes.io/part-of=urn_ngsi-ld_Service_complexmultimode
                    pod-template-hash=746b48f754
Annotations:        wg-client-config-hash: dc5ef3466dcafd05bfdf326417bdc86efa4bdec8c8762253e4c46e43a980ddfa
Status:             Running
IP:                 10.244.0.166
IPs:
  IP:             10.244.0.166
Controlled By:  ReplicaSet/aeros-service-complexmultimode-component-amf-crawler-746b48f754
Containers:
  wg-client:
    Container ID:   containerd://9cdc882fb84b0b3293daa118894bffe182ae3ff7412b97a32398e17ab5442032
    Image:          ghcr.io/linuxserver/wireguard
    Image ID:       ghcr.io/linuxserver/wireguard@sha256:7dc8e7e90ef50f16637b5d51f174481676086c5f46461555fc270fa5f297976a
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Mon, 06 Oct 2025 18:10:54 +0300
    Ready:          True
    Restart Count:  0
    Environment:
      WG_DISABLE_IPV6:  yes
    Mounts:
      /config/wg_confs from wg-config (rw)
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-vs5cv (ro)
  aeros-service-complexmultimode-component-amf-crawler:
    Container ID:   containerd://a7dc31a56bb198033dbcaf6ee1d81de583e0c0d905611ab2ac7dbad433cdccd6
    Image:          ghcr.io/front-research-group/amf_crawler:latest
    Image ID:       ghcr.io/front-research-group/amf_crawler@sha256:cf609c91a4e96e483664b9231af99fcae1fbed68247e0acbd954f3457fcdaeff
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Mon, 06 Oct 2025 18:11:10 +0300
    Ready:          True
    Restart Count:  0
    Environment:
      SMF_OPEN5GS_ORG:  10.220.2.73
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-vs5cv (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  wg-config:
    Type:       ConfigMap (a volume populated by a ConfigMap)
    Name:       aeros-service-complexmultimode-component-amf-crawler-wg-config
    Optional:   false
```

```
{
    "id": "urn:ngsi-ld:InfrastructureElement:ncsrd01:02000adc0228",
    "type": "InfrastructureElement",
    "domain": "urn:ngsi-ld:Domain:ncsrd01",
    "hostname": "aeros-master",
    "containerTechnology": "Kubernetes",
    "internalIpAddress": "10.220.2.40",
    "macAddress": "02:00:0a:dc:02:28",
    "lowLevelOrchestrator": "urn:ngsi-ld:LowLevelOrchestrator:ncsrd01:Kubernetes",
    "cpuCores": 2,
    "cpuFreqMax": 2200,
    "currentCpuUsage": 13,
    "gpu": "false",
    "gpuMemory": -1,
    "ramCapacity": 12544,
    "availableRam": 10624,
    "currentRamUsage": 1920,
    "currentRamUsagePct": 16,
    "diskType": "HDD",
    "diskCapacity": 50885,
    "availableDisk": 34052,
    "currentDiskUsage": 16816,
    "currentDiskUsagePct": 34,
    "netSpeedUp": -1,
    "netSpeedDown": -1,
    "netTrafficUp": 0,
    "netTrafficDown": 0,
    "netLostPackages": 0,
    "avgPowerConsumption": 5,
    "currentPowerConsumption": 3,
    "powerSource": "urn:ngsi-ld:none",
    "energyEfficiencyRatio": 0,
    "realTimeCapable": false,
    "trustScore": -1,
    "cpuArchitecture": "urn:ngsi-ld:CpuArchitecture:x64",
    "operatingSystem": "urn:ngsi-ld:OperatingSystem:Linux",
    "infrastructureElementTier": "urn:ngsi-ld:InfrastructureElementTier:Cloud",
```

- *mongodb*, service component placement (port created, image selected, wg client sidecar created) and compliance of IE selected (respecting restraints)

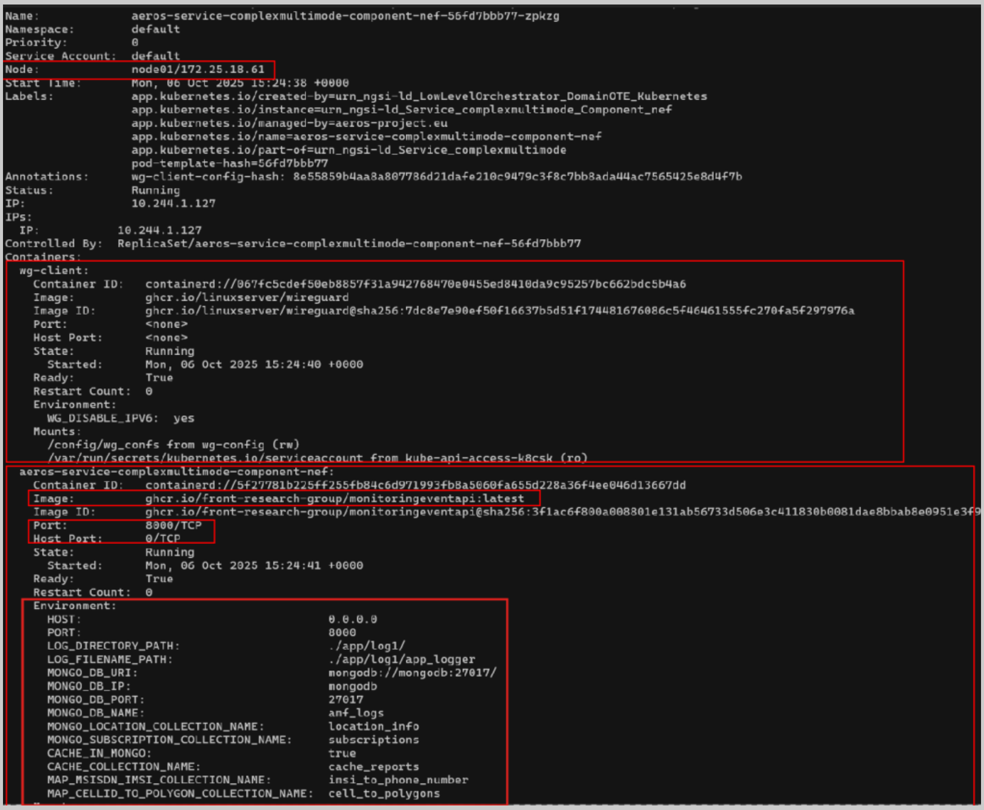


```
Name:             aeros-service-complexmultimode-component-mongodb-69db845d4htvsb
Namespace:        default
Priority:         0
Service Account:  default
Node:             master/172.25.18.60
Start Time:       Mon, 06 Oct 2025 15:10:45 +0000
Labels:           app.kubernetes.io/created-by=urn_ngsi-ld_LowLevelOrchestrator_DomainOTE_Kuberne
                  app.kubernetes.io/instance=urn_ngsi-ld_Service_complexmultimode_Component_mongo
                  app.kubernetes.io/managed-by=aeros-project.eu
                  app.kubernetes.io/name=aeros-service-complexmultimode-component-mongodb
                  app.kubernetes.io/part-of=urn_ngsi-ld_Service_complexmultimode
                  pod-template-hash=69db845d46
Annotations:      wg-client-config-hash: 014e3c6867e34859baa0e6df07da5f1c56fa6f61c135a136e39bc778
Status:           Running
IP:               10.244.0.83
IPs:
  IP:           10.244.0.83
Controlled By:  ReplicaSet/aeros-service-complexmultimode-component-mongodb-69db845d46
Containers:
  wg-client:
    Container ID:   containerd://b06e99e0d85230a1aa4a76c8cdaa1878ce5a05fac3c13c2922106ed46e9c3b73
    Image:          ghcr.io/linuxserver/wireguard
    Image ID:       ghcr.io/linuxserver/wireguard@sha256:7dc8e7e90ef50f16637b5d51f174481676086c5f
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Mon, 06 Oct 2025 15:10:52 +0000
    Ready:          True
    Restart Count:  0
    Environment:
      WG_DISABLE_IPV6:  yes
    Mounts:
      /config/wg_confs from wg-config (rw)
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-qp8cc (ro)
  aeros-service-complexmultimode-component-mongodb:
    Container ID:   containerd://d38b03b73fa0ad7e4889d3cca96ebe397818ee1aa8a1e2d8043cab54b44fb551
    Image:          mongo:latest
    Image ID:       docker.io/library/mongo@sha256:ea783d8ac4dcac9f8a7ff236b26a52e36649fc1bdd1778
    Port:           27017/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Mon, 06 Oct 2025 15:10:56 +0000
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-qp8cc (ro)
Conditions:
  Type                        Status
  PodReadyToStartContainers   True
  Initialized                 True
  Ready                       True
  ContainersReady             True
  PodScheduled                True
Volumes:
  wg-config:
    Type:       ConfigMap (a volume populated by a ConfigMap)
```

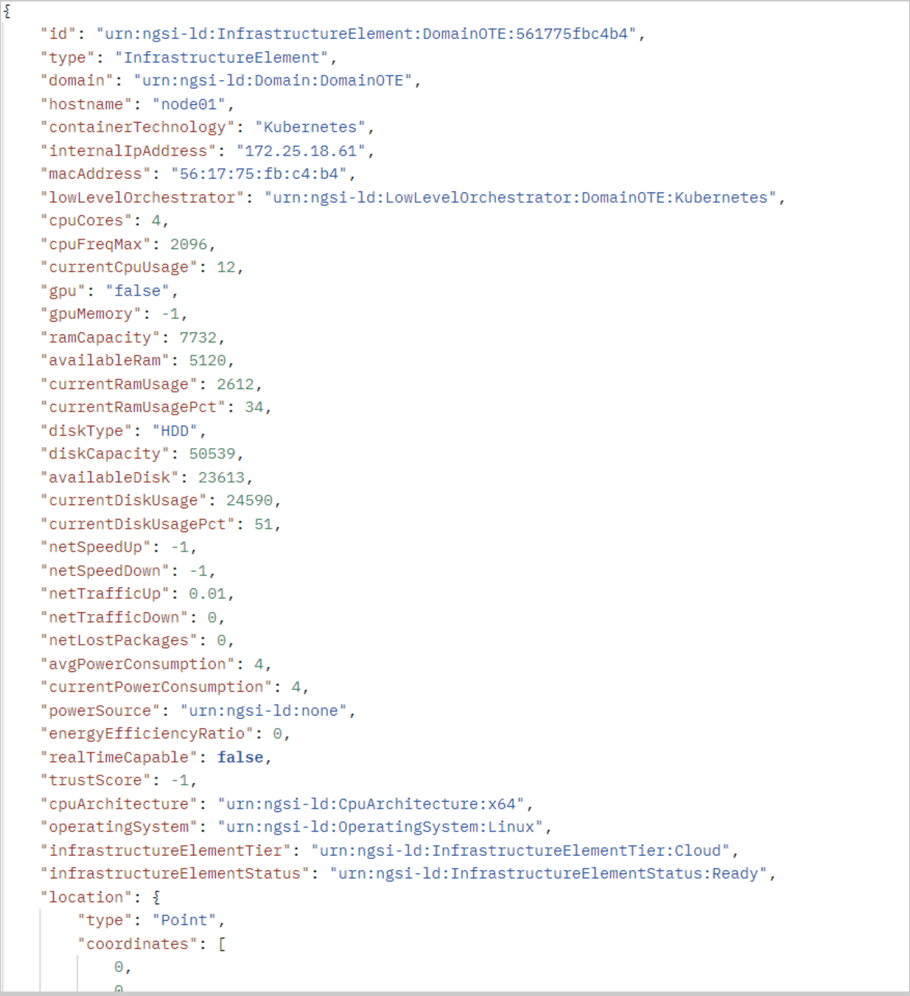
Requested ports

```
{
    "id": "urn:ngsi-ld:InfrastructureElement:DomainOTE:d23699581cdc",
    "type": "InfrastructureElement",
    "domain": "urn:ngsi-ld:Domain:DomainOTE",
    "hostname": "master",
    "containerTechnology": "Kubernetes",
    "internalIpAddress": "172.25.18.60",
    "macAddress": "d2:36:99:58:1c:dc",
    "lowLevelOrchestrator": "urn:ngsi-ld:LowLevelOrchestrator:DomainOTE:Kubernetes",
    "cpuCores": 4,
    "cpuFreqMax": 2096,
    "currentCpuUsage": 9,
    "gpu": "false",
    "gpuMemory": -1,
    "ramCapacity": 7138,
    "availableRam": 16886,
    "currentRamUsage": 4087,
    "currentRamUsagePct": 20,
    "diskType": "HDD",
    "diskCapacity": 103375,
    "availableDisk": 56438,
    "currentDiskUsage": 42454,
    "currentDiskUsagePct": 43,
    "netSpeedUp": -1,
    "netSpeedDown": -1,
    "netTrafficUp": 0.01,
    "netTrafficDown": 0.01,
    "netLostPackages": 0,
    "avgPowerConsumption": 4,
    "currentPowerConsumption": 4,
    "powerSource": "urn:ngsi-ld:none",
    "energyEfficiencyRatio": 0,
    "realTimeCapable": false,
    "trustScore": -1,
    "cpuArchitecture": "urn:ngsi-ld:CpuArchitecture:x64",
    "operatingSystem": "urn:ngsi-ld:OperatingSystem:Linux",
    "infrastructureElementTier": "urn:ngsi-ld:InfrastructureElementTier:Cloud",
    "infrastructureElementStatus": "urn:ngsi-ld:InfrastructureElementStatus:Ready",
    "location": {
```

- *nef*, service component placement (port exposed, evn variables set, image selected, wg client sidecar created) and compliance of IE selected (respecting restraints and also located in OTE domain)

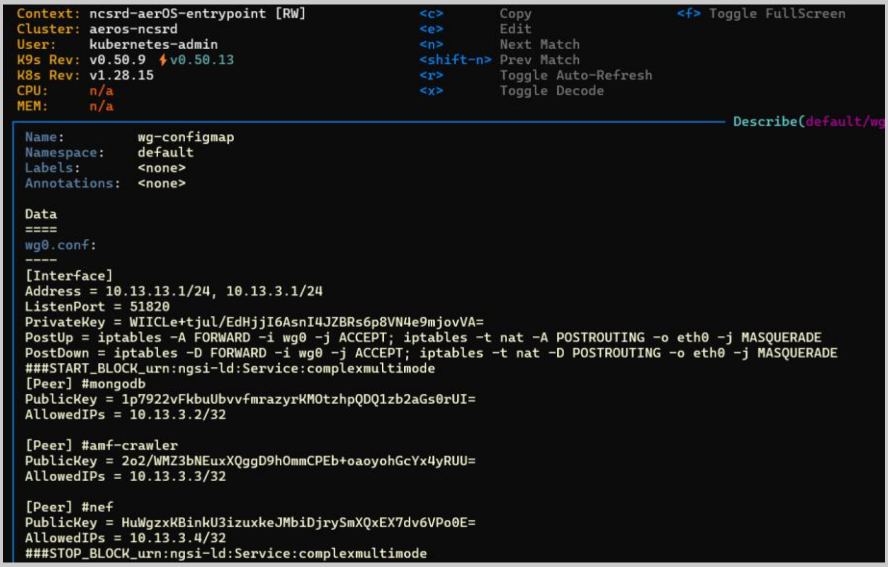


```
Name:             aeros-service-complexmultimode-component-nef-56fd7bbb77-zpkzg
Namespace:        default
Priority:         0
Service Account:  default
Node:             node01/172.25.18.61
Start Time:       Mon, 06 Oct 2025 15:24:38 +0000
Labels:           app.kubernetes.io/created-by=urn_ngsi-ld_LowLevelOrchestrator_DomainOTE_Kubernetes
                  app.kubernetes.io/instance=urn_ngsi-ld_Service_complexmultimode_Component_nef
                  app.kubernetes.io/managed-by=aeros-project.eu
                  app.kubernetes.io/name=aeros-service-complexmultimode-component-nef
                  app.kubernetes.io/part-of=urn_ngsi-ld_Service_complexmultimode
                  pod-template-hash=56fd7bbb77
Annotations:      wg-client-config-hash: 8e55859b4aa8a807786d21dafe210c9479c3f8c7bb8ada44ac7565425e8d4f7b
Status:           Running
IP:               10.244.1.127
IPs:
  IP:           10.244.1.127
Controlled By:  ReplicaSet/aeros-service-complexmultimode-component-nef-56fd7bbb77
Containers:
  wg-client:
    Container ID:   containerd://067fc5cdef50eb8857f31a942768470e0455ed8410da9c95257bc662bdc5b4a6
    Image:          ghcr.io/linuxserver/wireguard
    Image ID:       ghcr.io/linuxserver/wireguard@sha256:7dc8e7e90ef50f16637b5d51f174481676086c5f46461555fc270fa5f297976a
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Mon, 06 Oct 2025 15:24:40 +0000
    Ready:          True
    Restart Count:  0
    Environment:
      WG_DISABLE_IPV6:  yes
    Mounts:
      /config/wg_confs from wg-config (rw)
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-k8csk (ro)
  aeros-service-complexmultimode-component-nef:
    Container ID:   containerd://5f27781b225ff255fb84c6d971993fb8a5060fa655d228a36f4ee046d13667dd
    Image:          ghcr.io/front-research-group/monitoringeventapi:latest
    Image ID:       ghcr.io/front-research-group/monitoringeventapi@sha256:3f1ac6f800a008801e131ab56733d506e3c411830b0081dae8bbab8e0951e3f9
    Port:           8000/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Mon, 06 Oct 2025 15:24:41 +0000
    Ready:          True
    Restart Count:  0
    Environment:
      HOST:                                     0.0.0.0
      PORT:                                     8000
      LOG_DIRECTORY_PATH:                       ./app/log1/
      LOG_FILENAME_PATH:                        ./app/log1/app_logger
      MONGO_DB_URI:                             mongodb://mongodb:27017/
      MONGO_DB_IP:                              mongodb
      MONGO_DB_PORT:                            27017
      MONGO_DB_NAME:                            anf_logs
      MONGO_LOCATION_COLLECTION_NAME:           location_info
      MONGO_SUBSCRIPTION_COLLECTION_NAME:       subscriptions
      CACHE_IN_MONGO:                           true
      CACHE_COLLECTION_NAME:                    cache_reports
      MAP_MSISDN_IMSI_COLLECTION_NAME:          imsi_to_phone_number
      MAP_CELLID_TO_POLYGON_COLLECTION_NAME:    cell_to_polygons
```

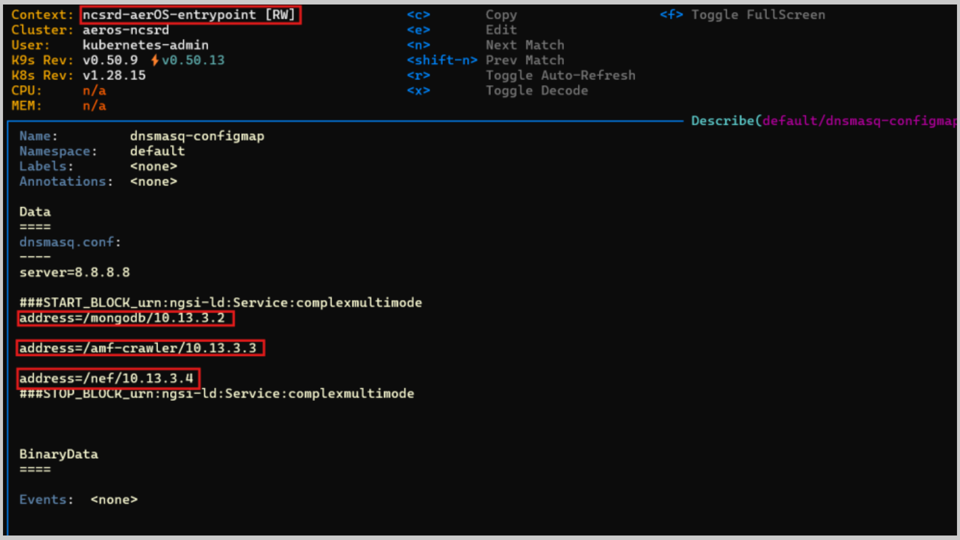


```json
{
    "id": "urn:ngsi-ld:InfrastructureElement:DomainOTE:561775fbc4b4",
    "type": "InfrastructureElement",
    "domain": "urn:ngsi-ld:Domain:DomainOTE",
    "hostname": "node01",
    "containerTechnology": "Kubernetes",
    "internalIpAddress": "172.25.18.61",
    "macAddress": "56:17:75:fb:c4:b4",
    "lowLevelOrchestrator": "urn:ngsi-ld:LowLevelOrchestrator:DomainOTE:Kubernetes",
    "cpuCores": 4,
    "cpuFreqMax": 2096,
    "currentCpuUsage": 12,
    "gpu": "false",
    "gpuMemory": -1,
    "ramCapacity": 7732,
    "availableRam": 5120,
    "currentRamUsage": 2612,
    "currentRamUsagePct": 34,
    "diskType": "HDD",
    "diskCapacity": 50539,
    "availableDisk": 23613,
    "currentDiskUsage": 24590,
    "currentDiskUsagePct": 51,
    "netSpeedUp": -1,
    "netSpeedDown": -1,
    "netTrafficUp": 0.01,
    "netTrafficDown": 0,
    "netLostPackages": 0,
    "avgPowerConsumption": 4,
    "currentPowerConsumption": 4,
    "powerSource": "urn:ngsi-ld:none",
    "energyEfficiencyRatio": 0,
    "realTimeCapable": false,
    "trustScore": -1,
    "cpuArchitecture": "urn:ngsi-ld:CpuArchitecture:x64",
    "operatingSystem": "urn:ngsi-ld:OperatingSystem:Linux",
    "infrastructureElementTier": "urn:ngsi-ld:InfrastructureElementTier:Cloud",
    "infrastructureElementStatus": "urn:ngsi-ld:InfrastructureElementStatus:Ready",
    "location": {
        "type": "Point",
        "coordinates": [
            0,
            0
```

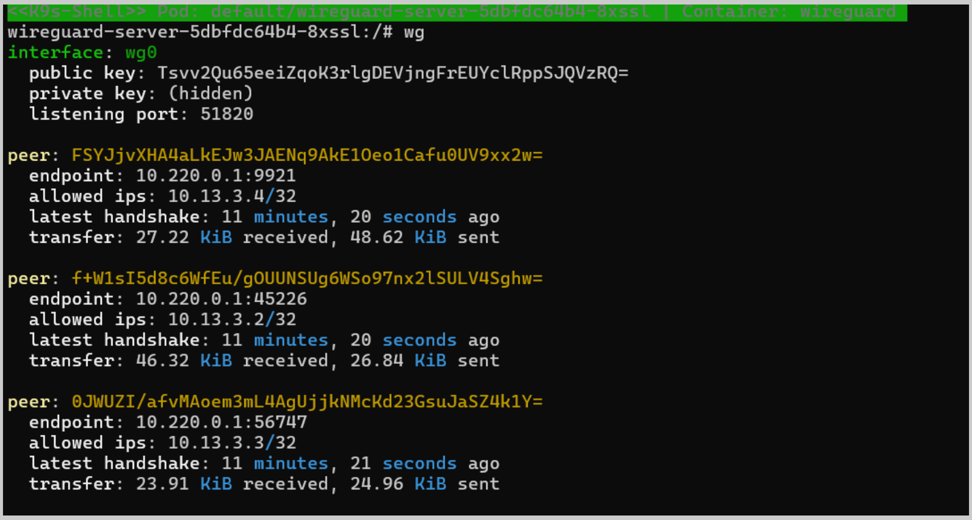
- Overlay configuration (server conf and registered peers in both wg and dnsmasq) and wireguard handshakes.



```
Context: ncsrd-aerOS-entrypoint [RW]             <c>        Copy                  <f> Toggle FullScreen
Cluster: aeros-ncsrd                             <e>        Edit
User:    kubernetes-admin                        <n>        Next Match
K9s Rev: v0.50.9 ⚡v0.50.13                       <shift-n>  Prev Match
K8s Rev: v1.28.15                                <r>        Toggle Auto-Refresh
CPU:     n/a                                     <x>        Toggle Decode
MEM:     n/a
 ──────────────────────────────────────────────────────────────────────────── Describe(default/wg
  Name:         wg-configmap
  Namespace:    default
  Labels:       <none>
  Annotations:  <none>

  Data
  ====
  wg0.conf:
  ----
  [Interface]
  Address = 10.13.13.1/24, 10.13.3.1/24
  ListenPort = 51820
  PrivateKey = WIICLe+tjul/EdHjjI6AsnI4JZBRs6p8VN4e9mjovVA=
  PostUp = iptables -A FORWARD -i wg0 -j ACCEPT; iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
  PostDown = iptables -D FORWARD -i wg0 -j ACCEPT; iptables -t nat -D POSTROUTING -o eth0 -j MASQUERADE
  ###START_BLOCK_urn:ngsi-ld:Service:complexmultimode
  [Peer] #mongodb
  PublicKey = 1p7922vFkbuUbvvfmrazyrKMOtzhpQDQ1zb2aGs0rUI=
  AllowedIPs = 10.13.3.2/32

  [Peer] #amf-crawler
  PublicKey = 2o2/WMZ3bNEuxXQggD9hOmmCPEb+oaoyohGcYx4yRUU=
  AllowedIPs = 10.13.3.3/32

  [Peer] #nef
  PublicKey = HuWgzxKBinkU3izuxkeJMbiDjrySmXQxEX7dv6VPo0E=
  AllowedIPs = 10.13.3.4/32
  ###STOP_BLOCK_urn:ngsi-ld:Service:complexmultimode
```

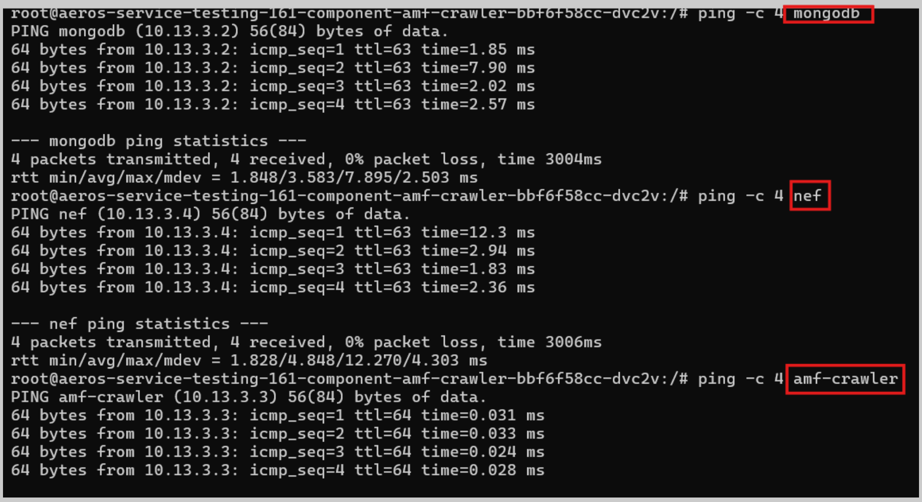


```
Context: ncsrd-aerOS-entrypoint [RW]             <c>        Copy                  <f> Toggle FullScreen
Cluster: aeros-ncsrd                             <e>        Edit
User:    kubernetes-admin                        <n>        Next Match
K9s Rev: v0.50.9 ⚡v0.50.13                       <shift-n>  Prev Match
K8s Rev: v1.28.15                                <r>        Toggle Auto-Refresh
CPU:     n/a                                     <x>        Toggle Decode
MEM:     n/a
 ──────────────────────────────────────────────────────────────────── Describe(default/dnsmasq-configmap
  Name:         dnsmasq-configmap
  Namespace:    default
  Labels:       <none>
  Annotations:  <none>

  Data
  ====
  dnsmasq.conf:
  ----
  server=8.8.8.8

  ###START_BLOCK_urn:ngsi-ld:Service:complexmultimode
  address=/mongodb/10.13.3.2

  address=/amf-crawler/10.13.3.3

  address=/nef/10.13.3.4
  ###STOP_BLOCK_urn:ngsi-ld:Service:complexmultimode



  BinaryData
  ====

  Events:  <none>
```



```
<<K9s-Shell>> Pod: default/wireguard-server-5dbfdc64b4-8xssl | Container: wireguard
wireguard-server-5dbfdc64b4-8xssl:/# wg
interface: wg0
  public key: Tsvv2Qu65eeiZqoK3rlgDEVjngFrEUYclRppSJQVzRQ=
  private key: (hidden)
  listening port: 51820

peer: FSYJjvXHA4aLkEJw3JAENq9AkE1Oeo1Cafu0UV9xx2w=
  endpoint: 10.220.0.1:9921
  allowed ips: 10.13.3.4/32
  latest handshake: 11 minutes, 20 seconds ago
  transfer: 27.22 KiB received, 48.62 KiB sent

peer: f+W1sI5d8c6WfEu/gOUUNSUg6WSo97nx2lSULV4Sghw=
  endpoint: 10.220.0.1:45226
  allowed ips: 10.13.3.2/32
  latest handshake: 11 minutes, 20 seconds ago
  transfer: 46.32 KiB received, 26.84 KiB sent

peer: 0JWUZI/afvMAoem3mL4AgUjjkNMcKd23GsuJaSZ4k1Y=
  endpoint: 10.220.0.1:56747
  allowed ips: 10.13.3.3/32
  latest handshake: 11 minutes, 21 seconds ago
  transfer: 23.91 KiB received, 24.96 KiB sent
```

- Ping from component to component over network overlay resolving service components names



```
root@aeros-service-testing-161-component-amf-crawler-bbf6f58cc-dvc2v:/# ping -c 4 mongodb
PING mongodb (10.13.3.2) 56(84) bytes of data.
64 bytes from 10.13.3.2: icmp_seq=1 ttl=63 time=1.85 ms
64 bytes from 10.13.3.2: icmp_seq=2 ttl=63 time=7.90 ms
64 bytes from 10.13.3.2: icmp_seq=3 ttl=63 time=2.02 ms
64 bytes from 10.13.3.2: icmp_seq=4 ttl=63 time=2.57 ms

--- mongodb ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.848/3.583/7.895/2.503 ms
root@aeros-service-testing-161-component-amf-crawler-bbf6f58cc-dvc2v:/# ping -c 4 nef
PING nef (10.13.3.4) 56(84) bytes of data.
64 bytes from 10.13.3.4: icmp_seq=1 ttl=63 time=12.3 ms
64 bytes from 10.13.3.4: icmp_seq=2 ttl=63 time=2.94 ms
64 bytes from 10.13.3.4: icmp_seq=3 ttl=63 time=1.83 ms
64 bytes from 10.13.3.4: icmp_seq=4 ttl=63 time=2.36 ms

--- nef ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 1.828/4.848/12.270/4.303 ms
root@aeros-service-testing-161-component-amf-crawler-bbf6f58cc-dvc2v:/# ping -c 4 amf-crawler
PING amf-crawler (10.13.3.3) 56(84) bytes of data.
64 bytes from 10.13.3.3: icmp_seq=1 ttl=64 time=0.031 ms
64 bytes from 10.13.3.3: icmp_seq=2 ttl=64 time=0.033 ms
64 bytes from 10.13.3.3: icmp_seq=3 ttl=64 time=0.024 ms
64 bytes from 10.13.3.3: icmp_seq=4 ttl=64 time=0.028 ms
```

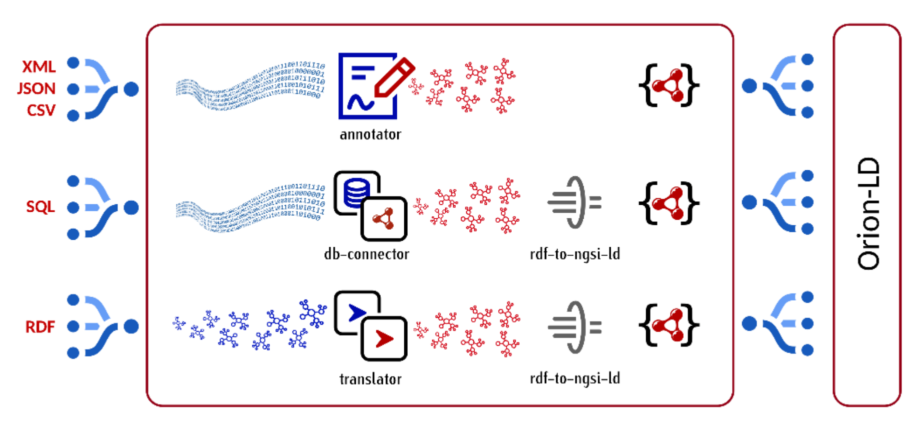
The same results were observed in all pilot deployments; in particular, the complex service deployments presented in **KPI 1.1.1** fully validated this, as they spanned manual, semi-automated, and automated placements, local and cross-domain deployments, NodePort exposure, and overlay creation.

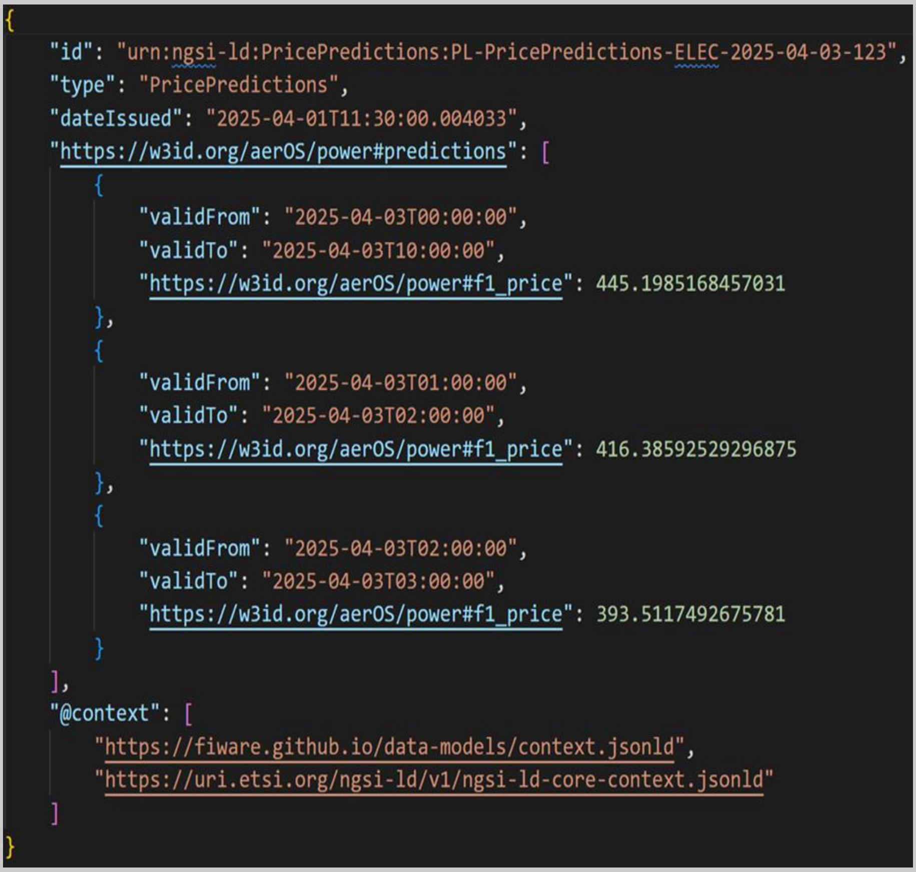
aerOS Data Fabric

KPI 1.2.1 Full support for data pipelines in all use cases (incl. open calls), identified through requirements elicitation (KVI-5.1)

Table 29: KPI 1.2.1 Full support for data pipelines in all use cases, identified through requirements elicitation (KVI-5.1)

KPI ID number and partner resp.	**KPI 1.2.1**
KPI Name	**Full support for data pipelines in all use cases (incl. open calls), identified through requirements elicitation**
Description	aerOS Data Fabric exposes configurable tools that are used to build data transit and transformation workflows (data pipelines). This KPI measures the coverage of pipelines, that are prepared within the project, and required by the use-cases.
Motivation	Verification of the Data Fabric tools, that support the creation of data pipelines in practice.
Target value	>50% scenarios
Prerequisites	Data Product Manager and Data Product Pipeline components have been implemented and deployed.
aerOS components (task)	Semantic Annotator, Semantic Translator (T4.1), Data Fabric (T4.2)

<table>
<tr><td>Evaluation means</td><td colspan="3">One of the aerOS basic goals is to provide flexibility and adequacy in handling the data pipelines that may be encountered in its applications. To verify that this is achievable, data pipelines required by the use-cases (and open-callers) have been specified, configured and created using aerOS Data Fabric and associated tools.
Components such as the DB Connector, Semantic Annotator, the Semantic Translator, and the RDF to NGSI-LD serializer are used to construct data processing pipelines within the Data Fabric. These components allow for the convenient definition of data processing steps that lead from potentially unstructured input data to semantically annotated output data offered to Data Fabric users/customers.

By taking advantage of its modular architecture and the flexibility of the data handling mechanisms it offers, as well as the comprehensive support already available for the most commonly used data formats, Data Fabric have met these requirements. The evaluation process, while somewhat “binary” in nature, has been conducted at all stages of pilots (and open calls) development. It has been applied to all scenarios where there was a need to create and handle data pipelines.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>0</i></td><td><i>7/12 scenarios (58%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">All 12 use cases by employing the fundamental aerOS infrastructure leverage the Data Fabric mechanisms within the context of LLO or HLO, for instance. However, some, such as three of the four use cases of Pilot 1 and both use cases of Pilot 4, found no application for Data Fabric due to the specific nature of the problems they solve.
In Pilot 1.1, the semantically annotated data includes energy-related information such as the power consumption and CO_2 footprint of the production line's various components. It also offers parameters measured by the machines themselves, including temperature and humidity. The JSON-LD data is produced directly via Node-RED components and sent to the JSON-LD context broker. The two Pilot 2 use cases offer semantically annotated data coming from a dedicated weather station and energy sensors connected to a variable power source (Photovoltaic panels), using directly the Orion-LD context broker.</td></tr>
</table>



```
{
    "id": "urn:ngsi-ld:PricePredictions:PL-PricePredictions-ELEC-2025-04-03-123",
    "type": "PricePredictions",
    "dateIssued": "2025-04-01T11:30:00.004033",
    "https://w3id.org/aerOS/power#predictions": [
        {
            "validFrom": "2025-04-03T00:00:00",
            "validTo": "2025-04-03T10:00:00",
            "https://w3id.org/aerOS/power#f1_price": 445.1985168457031
        },
        {
            "validFrom": "2025-04-03T01:00:00",
            "validTo": "2025-04-03T02:00:00",
            "https://w3id.org/aerOS/power#f1_price": 416.38592529296875
        },
        {
            "validFrom": "2025-04-03T02:00:00",
            "validTo": "2025-04-03T03:00:00",
            "https://w3id.org/aerOS/power#f1_price": 393.5117492675781
        }
    ],
    "@context": [
        "https://fiware.github.io/data-models/context.jsonld",
        "https://uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context.jsonld"
    ]
}
```

Rysunek 1: Pilot 2 power prediction NGSI-LD message

Pilot 3 use cases explicitly employ Data Fabric for their internal needs, but do not offer semantically annotated data sources externally. Among the semantically annotated data sources are: vehicle location data, field condition data, and vehicle operation data,

The use cases of Pilot 5 demonstrate the full potential of Data Fabric; they both offer and consume a wide range of semantically annotated data sources and define and utilize semantic data pipelines.

Name: Energy-Power-Consumption
Owner: Cosmote-dev
Measurements from sensors that can be activated/deactivated and measure also energy metrics. It can be plugs (with remote on/off capabilities) or HVAC machines. The tag that seperates them is the device_type which describes all the different devices.
Context Broker
@ Domain: Domain01
Tags: energy power
Glossary terms:

Name: Environmental-Air-Quality
Owner: Cosmote-dev
Measurements from sensor that outputs an air quality score for the room.
Context Broker
@ Domain: Domain01
Tags: air quality
Glossary terms:

Name: Environmental-Dust
Owner: Cosmote-dev
Environmental measurements for various dust particles and gasses by a specific sensor type\n
Context Broker
@ Domain: Domain01
Tags: environmental dust
Glossary terms:

Name: Environmental-Gas
Owner: Cosmote-dev
Environmental measurements for various gasses by a specific sensor type.
Context Broker
@ Domain: Domain01
Tags: environmental gas
Glossary terms:

Name: Environmental-TH
Owner: Cosmote-dev
Environmental measurements: Temperature, Humidity taken from env sensors that are placed somewhere in the room.
Context Broker
@ Domain: Domain01
Tags: environmental temperature humidity
Glossary terms:

Name: Environmental-THP
Owner: Cosmote-dev
Environmental measurements: Temperature, Humidity, Pressure taken from env sensors that are placed somewhere in the room.
Context Broker
@ Domain: Domain01
Tags: environmental humidity temperature pressure
Glossary terms:

Name: Meter-Consumption
Owner: Cosmote-dev
Measurements from sensors that are placed in the panel of a room and measure energy metrics for the whole room.
Context Broker
@ Domain: Domain01
Tags: power meter consumption
Glossary terms:

Name: States
Owner: Cosmote-dev
Measurements from sensors that can be activated/deactivated. It can be plugs (with remote on/off capabilities).The tag that seperates them is the device_type which describes all the different devices.
Context Broker
@ Domain: Domain01
Tags: state device
Glossary terms:

Rysunek 2: Pilot 5 Orion-LD data products

In total, Pilot 5 creates eleven (11) data pipelines through the Data Fabric. All of these pipelines play an important role in the exchange of messages between various Pilot 5 application components. Based on the exposed data the management system for the "energy efficient, health-safe and sustainable smart building" can monitor, control, and predict different health parameters for its rooms.

KPI 1.2.2 Semantic and syntactic interoperability between all data producers and consumers in all use cases (KVI-5.2)

Table 30: KPI 1.2.2 Semantic and syntactic interoperability between all data producers and consumers in all use cases (KVI-5.2)

KPI ID number and partner resp.	**KPI 1.2.2**
KPI Name	**Semantic and syntactic interoperability between all data producers and consumers in all use cases**
Description	To achieve interoperability when exchanging data all participants, data producers and data consumers must understand the data. This KPI ensures, that the data is useful for producers and consumers, through either usage of a common syntax and semantics from the get-go, or by applying data transformations.
Motivation	Semantic and syntactic interoperability of data for aerOS Data Fabric.
Target value	>50% scenarios
Prerequisites	Data Product Manager and Data Product Pipeline components have been implemented and deployed.

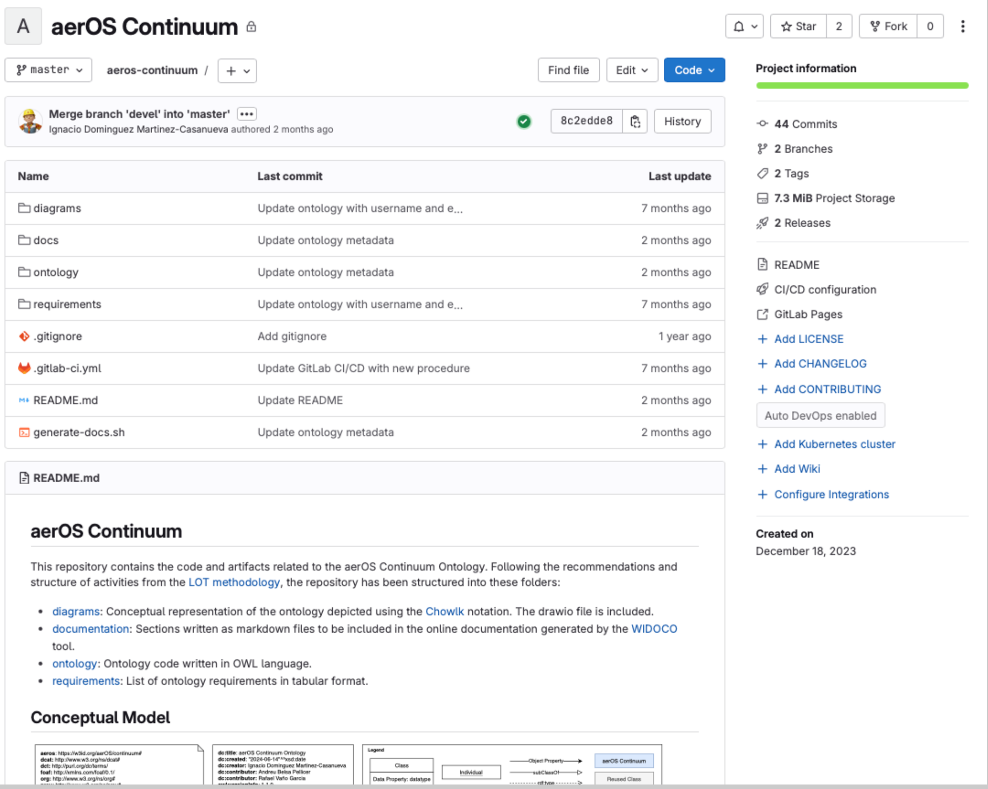
aerOS components (task)	Semantic Annotator, Semantic Translator (T4.1), Data Fabric (T4.2)		
Evaluation means	The scenario/use case is considered applicable for semantic and/or syntactic interoperability as long as the Data Fabric and the necessary tools (such as the DB Connector, Semantic Annotator, Semantic Translator, and/or the RDF to NGSI-LD serializer) for defining specific data pipelines are deployed, and a data producer and consumer are connected.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***N/A***	***7/12 scenarios (58%)***
Outcome elaboration (M38)	All 12 use cases employ the fundamental aerOS infrastructure and leverage Data Fabric-based mechanisms within the context of LLO or HLO. Seven of the use cases offer semantically annotated data and use the Orion-LD context broker, including the Pilot 1.1 and two Pilot 2 use cases. The Pilot 3 use cases explicitly employ the Data Fabric for internal purposes, but they do not offer externally accessible, semantically annotated data sources. The Pilot 5 use cases demonstrate the full potential of aerOS's syntactic and semantic interoperability mechanisms by offering and consuming semantically annotated data sources and by defining and utilizing eleven (11) semantic data pipelines through the aerOS Data Fabric.		

KPI 1.2.3 Reference implementation for a data infrastructure supporting full user-control in the definition of data sources, consumers and flows (KVI-5.3)

Table 31: KPI 1.2.3 Reference implementation for a data infrastructure supporting full user-control in the definition of data sources, consumers and flows (KVI-5.3)

KPI ID number and partner resp.	**KPI 1.2.3**
KPI Name	**Reference implementation for a data infrastructure supporting full user-control in the definition of data sources, consumers and flows of deployment compared to app blueprints**
Description	Number of aerOS use cases that have followed the standard Linked Open Terms (LOT) methodology for ontology development.
Motivation	Ontologies enable integrating data in the knowledge graph that implements the Data Fabric (i.e., the data infrastructure).
Target value	>=3
Prerequisites	Ontology has been developed following the guidelines defined by the LOT methodology.

<table>
<tr><td>aerOS components (task)</td><td colspan="3">Semantic Annotator (T4.1), Data Fabric (T4.2)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">The resulting ontology must be available in a GitLab repository, where the following ontology artifacts as recommended by the LOT methodology:
• Ontology requirements (in CSV format)
• Ontology diagram (based on Chowlk notation, created using draw.io tool)
• Ontology code (OWL code programmed and validated with Protégé tool)</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>0</i></td><td><i>2 (66%)</i></td><td><i>3 (100%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">Three use cases have been identified, namely, aerOS Continuum, aerOS Building, and the aerOS Data Catalog. The respective ontologies for these use cases have been development according to LOT methodology guidelines and recommended tools. The resulting ontologies are available on separate repositories in aerOS GitLab, including the ontology artifacts mentioned above.

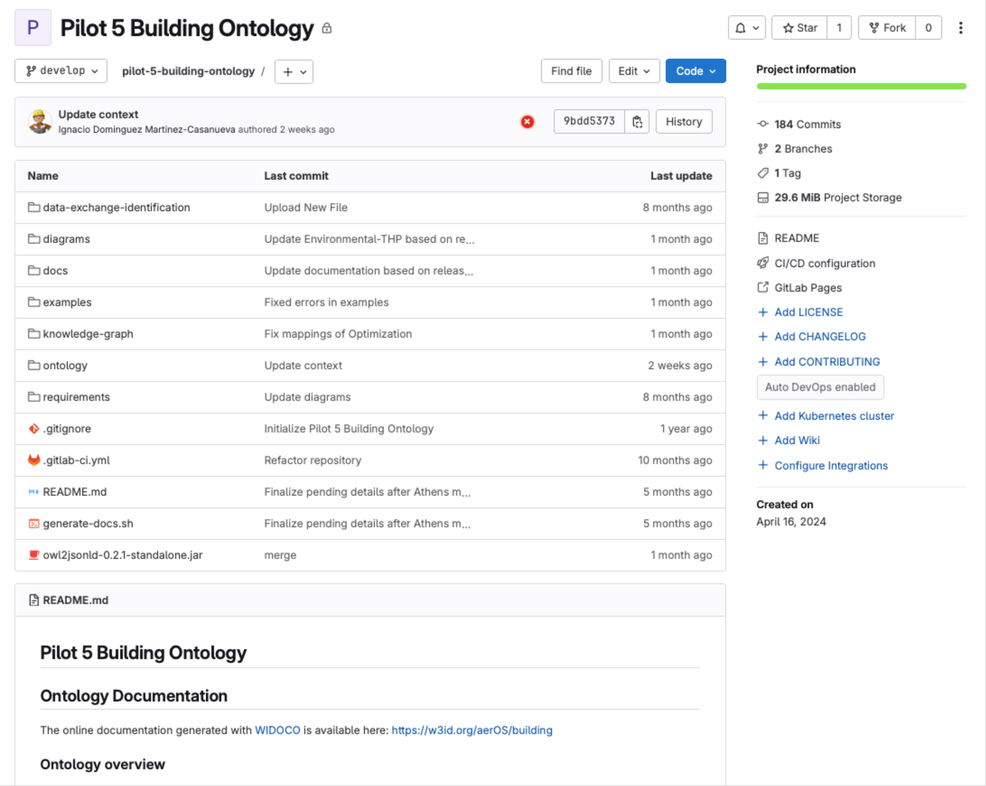
<u>Continuum Ontology</u>
GitLab repository: https://gitlab.aeros-project.eu/wp4/t4.1/aeros-continuum
The following snapshot depicts the structure of the GitLab repository for the aerOS Continuum.

aerOS Continuum
master · aeros-continuum /
Merge branch 'devel' into 'master'
Ignacio Dominguez Martinez-Casanueva authored 2 months ago
8c2edde8
<table><tr><th>Name</th><th>Last commit</th><th>Last update</th></tr><tr><td>diagrams</td><td>Update ontology with username and e...</td><td>7 months ago</td></tr><tr><td>docs</td><td>Update ontology metadata</td><td>2 months ago</td></tr><tr><td>ontology</td><td>Update ontology metadata</td><td>2 months ago</td></tr><tr><td>requirements</td><td>Update ontology with username and e...</td><td>7 months ago</td></tr><tr><td>.gitignore</td><td>Add gitignore</td><td>1 year ago</td></tr><tr><td>.gitlab-ci.yml</td><td>Update GitLab CI/CD with new procedure</td><td>7 months ago</td></tr><tr><td>README.md</td><td>Update README</td><td>2 months ago</td></tr><tr><td>generate-docs.sh</td><td>Update ontology metadata</td><td>2 months ago</td></tr></table>README.md
aerOS Continuum
This repository contains the code and artifacts related to the aerOS Continuum Ontology. Following the recommendations and structure of activities from the LOT methodology, the repository has been structured into these folders:
• diagrams: Conceptual representation of the ontology depicted using the Chowlk notation. The drawio file is included.
• documentation: Sections written as markdown files to be included in the online documentation generated by the WIDOCO tool.
• ontology: Ontology code written in OWL language.
• requirements: List of ontology requirements in tabular format.
Conceptual Model

Project information
44 Commits
2 Branches
2 Tags
7.3 MiB Project Storage
2 Releases
README
CI/CD configuration
GitLab Pages
Created on
December 18, 2023

<u>Building Ontology</u></td></tr>
</table>

GitLab repository: https://gitlab.aeros-project.eu/wp4/t4.1/pilot-5-building-ontology

The following snapshot depicts the structure of the GitLab repository for the aerOS Building Ontology.



Update context
Ignacio Dominguez Martinez-Casanueva authored 2 weeks ago

Name	Last commit	Last update
data-exchange-identification	Upload New File	8 months ago
diagrams	Update Environmental-THP based on re...	1 month ago
docs	Update documentation based on releas...	1 month ago
examples	Fixed errors in examples	1 month ago
knowledge-graph	Fix mappings of Optimization	1 month ago
ontology	Update context	2 weeks ago
requirements	Update diagrams	8 months ago
.gitignore	Initialize Pilot 5 Building Ontology	1 year ago
.gitlab-ci.yml	Refactor repository	10 months ago
README.md	Finalize pending details after Athens m...	5 months ago
generate-docs.sh	Finalize pending details after Athens m...	5 months ago
owl2jsonld-0.2.1-standalone.jar	merge	1 month ago

README.md

Pilot 5 Building Ontology

Ontology Documentation

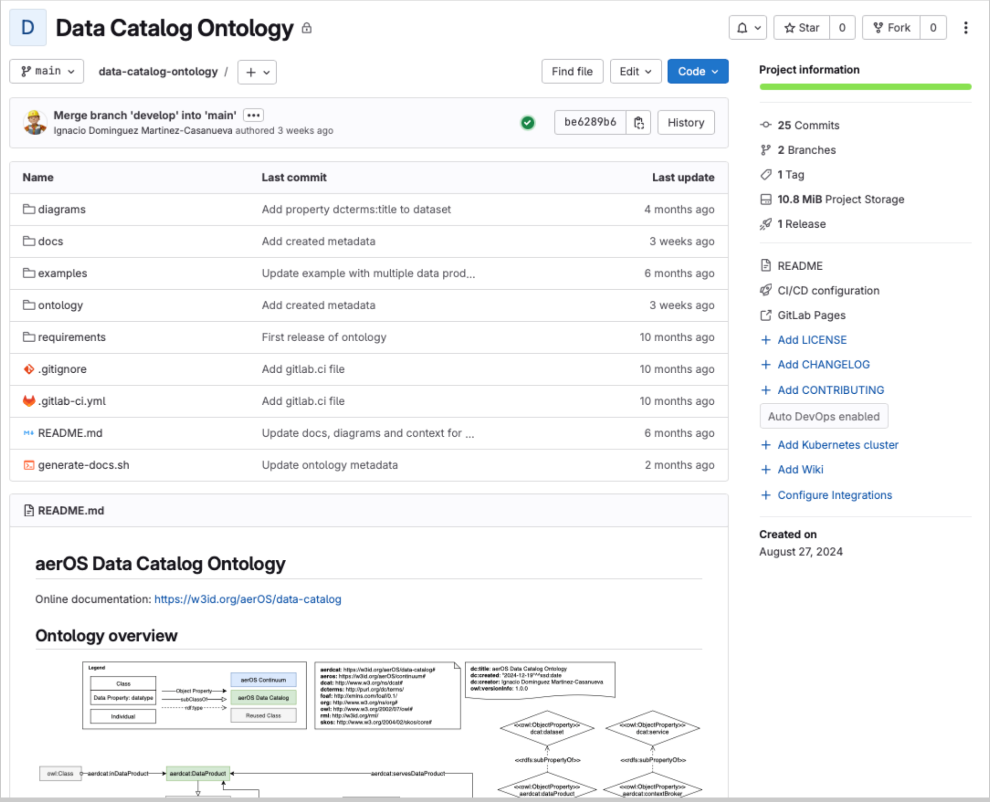
The online documentation generated with WIDOCO is available here: https://w3id.org/aerOS/building

Ontology overview

Data Catalog Ontology

GitLab repository: https://gitlab.aeros-project.eu/wp4/t4.1/data-catalog-ontology

The following snapshot depicts the structure of the GitLab repository for the aerOS Data Catalog Ontology.



Data Catalog Ontology

main / data-catalog-ontology /

Merge branch 'develop' into 'main'
Ignacio Dominguez Martinez-Casanueva authored 3 weeks ago
be6289b6

Name	Last commit	Last update
diagrams	Add property dcterms:title to dataset	4 months ago
docs	Add created metadata	3 weeks ago
examples	Update example with multiple data prod...	6 months ago
ontology	Add created metadata	3 weeks ago
requirements	First release of ontology	10 months ago
.gitignore	Add gitlab.ci file	10 months ago
.gitlab-ci.yml	Add gitlab.ci file	10 months ago
README.md	Update docs, diagrams and context for ...	6 months ago
generate-docs.sh	Update ontology metadata	2 months ago

README.md

aerOS Data Catalog Ontology

Online documentation: https://w3id.org/aerOS/data-catalog

Ontology overview

aerdcat: https://w3id.org/aerOS/data-catalog#
aeros: https://w3id.org/aerOS/continuum#
dcat: http://www.w3.org/ns/dcat#
dcterms: http://purl.org/dc/terms/
foaf: http://xmlns.com/foaf/0.1/
org: http://www.w3.org/ns/org#
owl: http://www.w3.org/2002/07/owl#
rml: http://w3id.org/rml/
skos: http://www.w3.org/2004/02/skos/core#

dc:title: aerOS Data Catalog Ontology
dc:created: "2024-12-19"^^xsd:date
dc:creator: Ignacio Dominguez Martinez-Casanueva
owl:versionInfo: 1.0.0

Project information

25 Commits
2 Branches
1 Tag
10.8 MiB Project Storage
1 Release

README
CI/CD configuration
GitLab Pages
Add LICENSE
Add CHANGELOG
Add CONTRIBUTING
Auto DevOps enabled
Add Kubernetes cluster
Add Wiki
Configure Integrations

Created on
August 27, 2024

KPI 1.2.4 # of data sovereignty initiatives

Table 32: KPI 1.2.4 # of data sovereignty initiatives

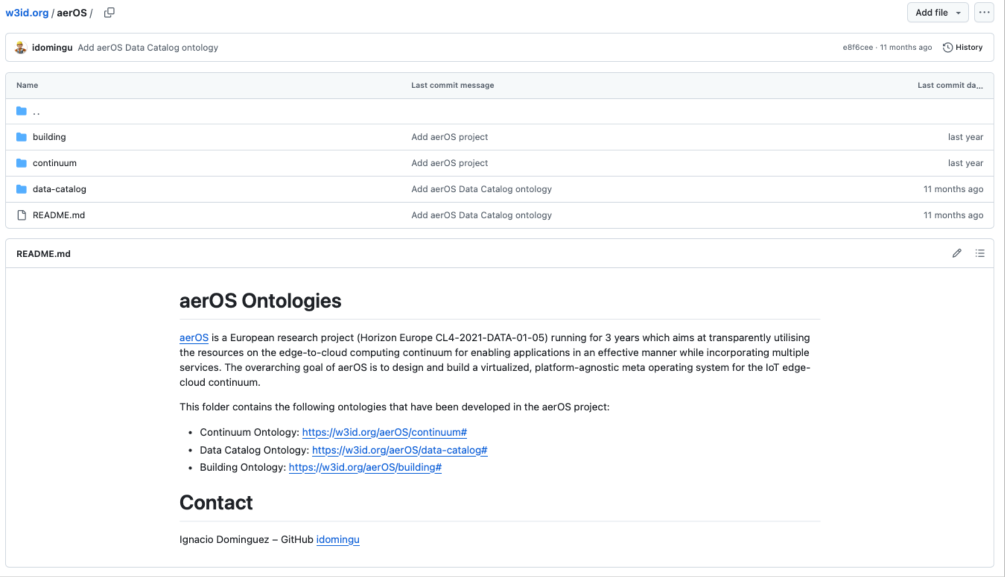
KPI ID number and partner resp.	**KPI 1.2.4**
KPI Name	**# of data sovereignty initiatives**
Description	Data sovereignty initiatives refer to efforts, policies, and frameworks with the main goal of ensuring data is subject to the laws and governance structure of where it is collected or processed. This KPI quantifies the number of data sovereignty initiatives that influence aerOS components.
Motivation	Data sovereignty is crucial to facilitate data sharing and trusted data transaction ensuring effective data usage control in distributed environments. The number of data sovereignty activities enhances the fidelity of AI models and effectiveness of autonomous control loops.
Target value	5
Prerequisites	*N/A*
aerOS components (task)	*N/A*
Evaluation means	By tracking and reporting every data sovereignty in which aerOS partners have actively contributed. A valid initiative is considered each action that directly helps handling

	complies with regulations or legal issues, increasing control over the data and protecting it from unauthorized access		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0***	***9 (180%)***	***9 (180%)***
Outcome elaboration (M38)	Software/components/applications that enable sovereignty: - Blueprint: 1 - DSSC v1.0: - Data Sovereignty Policy Management: 1 – ODLR - Data Space Communication Protocol: 1 - IDSCP 2.0 - Data Space Connector Supported: 2 – DSC, EDC - Data Catalogue: 1 – DCAT-AP - ID Management system: 1 – DAPS - Certificate Authority: 1 – X.509 - Digital Twin & Data Models: 1 – OPC-UA Work in progress - Data Space Interoperability: DSP (ISO/IEC JTC 1/SC 38) - Data Space Compatibility: 1 Eclipse TCK Further information on the enablers (software/components/applications) of sovereign data space related/due to aerOS can be found - https://gaia-x.eu/ - https://bdva.eu/ - https://www.fiware.org/ - https://dssc.eu/ https://internationaldataspaces.org/		

KPI 1.2.5 aerOS data models in open markets

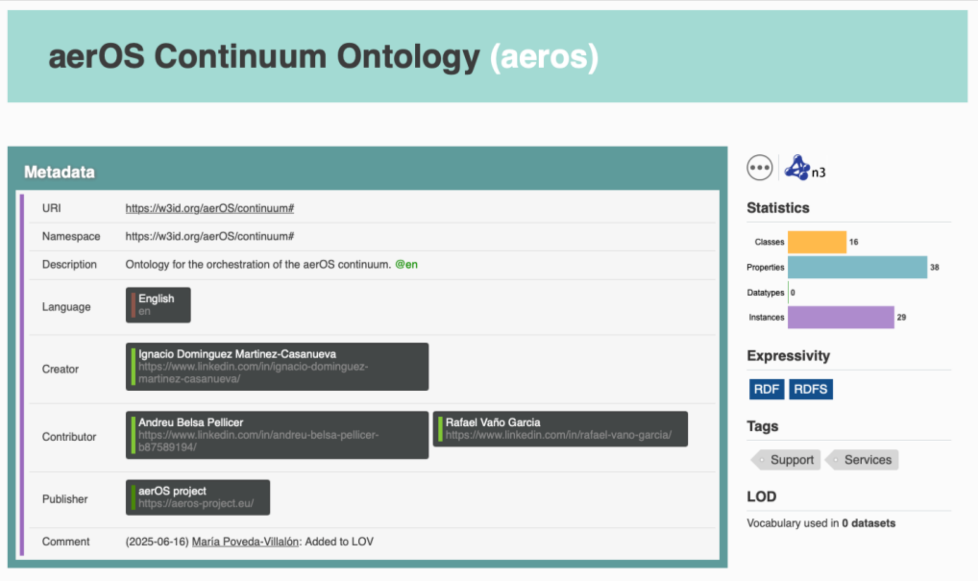
Table 33: KPI 1.2.5 aerOS data models in open markets

KPI ID number and partner resp.	**KPI 1.2.5**
KPI Name	**aerOS data models in open markets**
Description	Number of data models used in aerOS publicly available to the open-source community.
Motivation	Promotion of open data models targeting the IoT-Edge-Cloud continuum for improving interoperability.
Target value	5
Prerequisites	Ontology artifacts, namely, requirements list, diagram, code, and documentation, available in the respective GitLab repository.
aerOS components (task)	Semantic Annotator (T4.1), Data Fabric (T4.2)

<table>
<tr><td>Evaluation means</td><td colspan="3">The ontology must be publicly available, exposing an online documentation based on the WIDOCO tool[4]. Additionally, following LOT methodology best practices, the namespace URI of the ontology must be registered under the open w3id.org domain[5] and an entry of the ontology must be created in the Linked Open Vocabularies (LOV)[6] service for improved discoverability by the open-source community.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>2 (40%)</i></td><td><i>4 (80%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">All the ontologies developed in aerOS, namely, the aerOS Continuum Ontology, the aerOS Building Ontology, and the aerOS Data Catalog Ontology have been published publicly online, using WIDOCO tool and registered under the w3id.org domain.

w3id.org / aerOS /
idomingu Add aerOS Data Catalog ontology
e8f6cee · 11 months ago History
<table><tr><th>Name</th><th>Last commit message</th><th>Last commit da...</th></tr><tr><td>..</td><td></td><td></td></tr><tr><td>building</td><td>Add aerOS project</td><td>last year</td></tr><tr><td>continuum</td><td>Add aerOS project</td><td>last year</td></tr><tr><td>data-catalog</td><td>Add aerOS Data Catalog ontology</td><td>11 months ago</td></tr><tr><td>README.md</td><td>Add aerOS Data Catalog ontology</td><td>11 months ago</td></tr></table>README.md
aerOS Ontologies
aerOS is a European research project (Horizon Europe CL4-2021-DATA-01-05) running for 3 years which aims at transparently utilising the resources on the edge-to-cloud computing continuum for enabling applications in an effective manner while incorporating multiple services. The overarching goal of aerOS is to design and build a virtualized, platform-agnostic meta operating system for the IoT edge-cloud continuum.
This folder contains the following ontologies that have been developed in the aerOS project:
• Continuum Ontology: https://w3id.org/aerOS/continuum#
• Data Catalog Ontology: https://w3id.org/aerOS/data-catalog#
• Building Ontology: https://w3id.org/aerOS/building#
Contact
Ignacio Dominguez – GitHub idomingu
Additionally, the ontologies have been registered in the LOV service, as shown in the following figures:</td></tr>
</table>

---

[4] https://github.com/dgarijo/Widoco
[5] https://w3id.org
[6] https://lov.linkeddata.es/dataset/lov



aerOS Continuum Ontology (aeros)

Metadata

URI	https://w3id.org/aerOS/continuum#
Namespace	https://w3id.org/aerOS/continuum#
Description	Ontology for the orchestration of the aerOS continuum. @en
Language	English en
Creator	Ignacio Dominguez Martinez-Casanueva https://www.linkedin.com/in/ignacio-dominguez-martinez-casanueva/
Contributor	Andreu Belsa Pellicer https://www.linkedin.com/in/andreu-belsa-pellicer-b87589194/ Rafael Vaño Garcia https://www.linkedin.com/in/rafael-vano-garcia/
Publisher	aerOS project https://aeros-project.eu/
Comment	(2025-06-16) María Poveda-Villalón: Added to LOV

Statistics

Classes	16
Properties	38
Datatypes	0
Instances	29

Expressivity

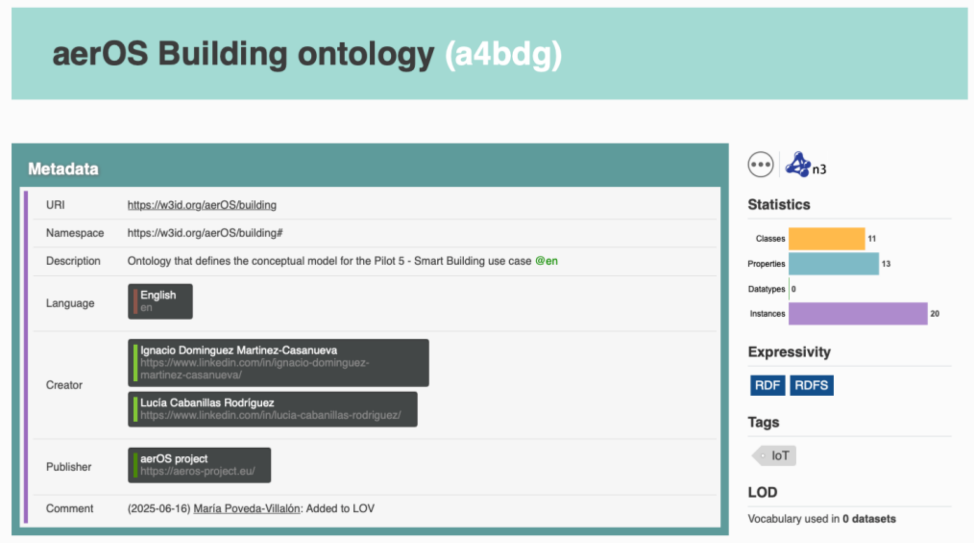
RDF RDFS

Tags

Support Services

LOD

Vocabulary used in **0 datasets**



aerOS Building ontology (a4bdg)

Metadata

URI	https://w3id.org/aerOS/building
Namespace	https://w3id.org/aerOS/building#
Description	Ontology that defines the conceptual model for the Pilot 5 - Smart Building use case @en
Language	English en
Creator	Ignacio Dominguez Martinez-Casanueva https://www.linkedin.com/in/ignacio-dominguez-martinez-casanueva/ Lucía Cabanillas Rodríguez https://www.linkedin.com/in/lucia-cabanillas-rodriguez/
Publisher	aerOS project https://aeros-project.eu/
Comment	(2025-06-16) María Poveda-Villalón: Added to LOV

Statistics

Classes	11
Properties	13
Datatypes	0
Instances	20

Expressivity

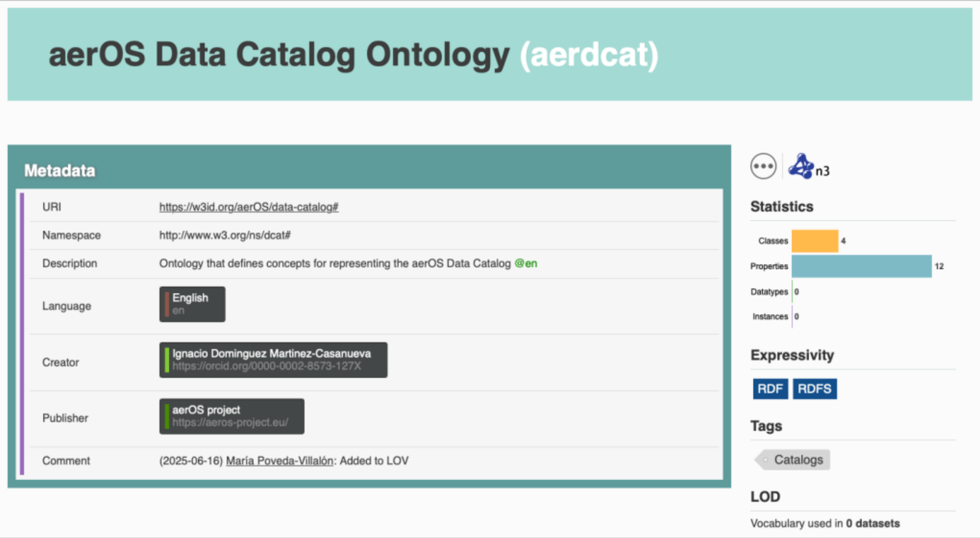
RDF RDFS

Tags

IoT

LOD

Vocabulary used in **0 datasets**



aerOS Data Catalog Ontology (aerdcat)

Metadata

URI	https://w3id.org/aerOS/data-catalog#
Namespace	http://www.w3.org/ns/dcat#
Description	Ontology that defines concepts for representing the aerOS Data Catalog @en
Language	English en
Creator	Ignacio Dominguez Martinez-Casanueva https://orcid.org/0000-0002-8573-127X
Publisher	aerOS project https://aeros-project.eu/
Comment	(2025-06-16) María Poveda-Villalón: Added to LOV

Statistics

Classes	4
Properties	12
Datatypes	0
Instances	0

Expressivity

RDF RDFS

Tags

Catalogs

LOD

Vocabulary used in **0 datasets**

Finally, the online documentation for each ontology can be found in the following links:

- **Continuum Ontology:** https://w3id.org/aerOS/continuum#

	- **Building Ontology:** https://w3id.org/aerOS/building# - **Data Catalog Ontology:** https://w3id.org/aerOS/data-catalog# The value achieved in this KPI (3) reflects a methodological decision rather than a technical limitation. While the original target was to publish 5 ontologies, during the project execution priority was given to the quality and relevance of the outputs rather than their quantity. Specifically, ontologies were developed for those pilots where data were modeled through the Data Fabric approach, and in those scenarios where semantic representation provided clear added value. This ensured that the ontologies complied with the required standards following the LOT methodology and best practices for publication (documentation with WIDOCO, namespace registration under w3id.org, and inclusion in LOV for discoverability). Therefore, the final outcome (3 ontologies) represents a set of high-value, contextually relevant ontologies aligned with the project's objectives, instead of meeting the quantitative target at the expense of applicability or relevance.

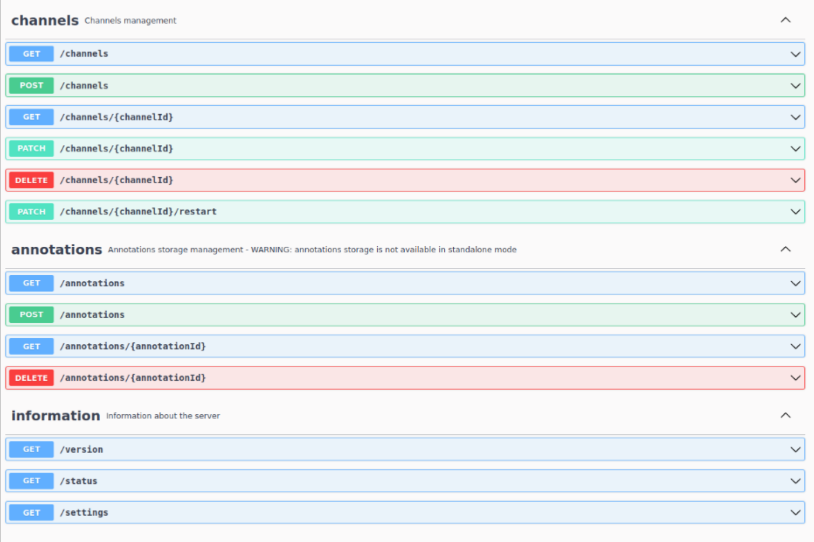
KPI 1.2.6 Semantic annotation support for commonly used data format

Table 34: KPI 1.2.6 Semantic annotation support for commonly used data format

KPI ID number and partner resp.	**KPI 1.2.6**		
KPI Name	**Semantic annotation support for commonly used data formats**		
Description	Semantic annotation component transforms "raw data" into NGSI-LD based on specific annotation rules. This KPI will measure the number of data formats that aerOS can transform into NGSI-LD.		
Motivation	Data-level semantic interoperability and support for the unified data handling within aerOS Data Fabric.		
Target value	>=3		
Prerequisites	Data Product Manager and Data Product Pipeline components have been implemented and deployed.		
aerOS components (task)	Semantic Annotator (T4.1), Data Fabric (T4.2)		
Evaluation means	The evaluation of the KPI is based on the number of "raw" data formats supported by the Semantic Annotator.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*N/A*	*3 (100%)*	*3 (100%)*

Outcome elaboration (M38)

Current implementation of the Semantic Annotator supports 3 widely used “raw data” formats: XML, CSV, and JSON. The REST configuration interface of the Semantic Annotator, based on Swagger, is shown in the following figure.



channels Channels management

- GET /channels
- POST /channels
- GET /channels/{channelId}
- PATCH /channels/{channelId}
- DELETE /channels/{channelId}
- PATCH /channels/{channelId}/restart

annotations Annotations storage management - WARNING: annotations storage is not available in standalone mode

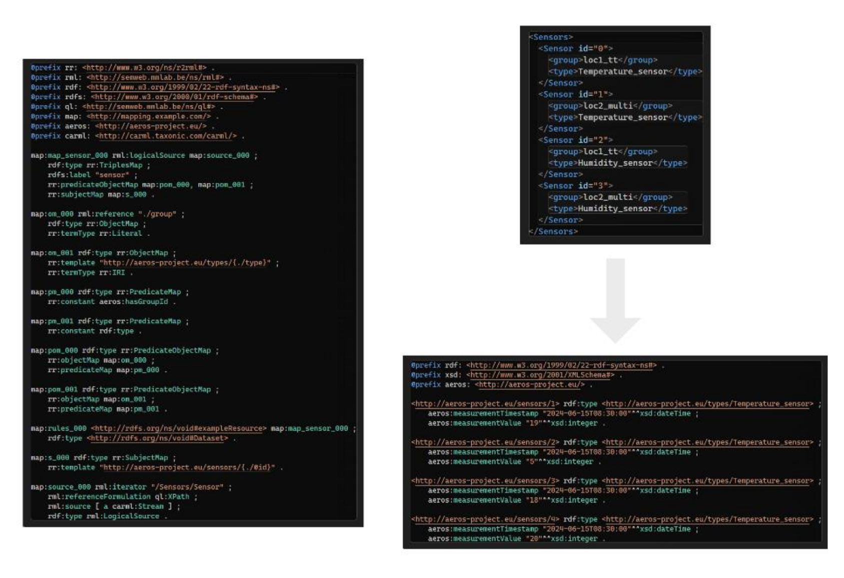
- GET /annotations
- POST /annotations
- GET /annotations/{annotationId}
- DELETE /annotations/{annotationId}

information Information about the server

- GET /version
- GET /status
- GET /settings

The first example demonstrates annotation of JSON-based “personal data”. In the process, the Semantic Annotator utilizes annotation rules (depicted on the left and expressed in CARML format) telling how to transform the “raw data” into its semantic counterpart.

In the second example, the tool is used to semantically annotate a series of CSV-encoded “measurement data” coming from a temperature sensor.



```
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix ql: <http://semweb.mmlab.be/ns/ql#> .
@prefix map: <http://mapping.example.com/> .
@prefix aeros: <http://aeros-project.eu/> .
@prefix carml: <http://carml.taxonic.com/carml/> .

map:map_sensor_000 rml:logicalSource map:source_000 ;
    rdf:type rr:TriplesMap ;
    rdfs:label "sensor" ;
    rr:predicateObjectMap map:pom_000, map:pom_001 ;
    rr:subjectMap map:s_000 .

map:om_000 rml:reference "./group" ;
    rdf:type rr:ObjectMap ;
    rr:termType rr:Literal .

map:om_001 rdf:type rr:ObjectMap ;
    rr:template "http://aeros-project.eu/types/{./type}" ;
    rr:termType rr:IRI .

map:pm_000 rdf:type rr:PredicateMap ;
    rr:constant aeros:hasGroupId .

map:pm_001 rdf:type rr:PredicateMap ;
    rr:constant rdf:type .

map:pom_000 rdf:type rr:PredicateObjectMap ;
    rr:objectMap map:om_000 ;
    rr:predicateMap map:pm_000 .

map:pom_001 rdf:type rr:PredicateObjectMap ;
    rr:objectMap map:om_001 ;
    rr:predicateMap map:pm_001 .

map:rules_000 <http://rdfs.org/ns/void#exampleResource> map:map_sensor_000 ;
    rdf:type <http://rdfs.org/ns/void#Dataset> .

map:s_000 rdf:type rr:SubjectMap ;
    rr:template "http://aeros-project.eu/sensors/{./@id}" .

map:source_000 rml:iterator "/Sensors/Sensor" ;
    rml:referenceFormulation ql:XPath ;
    rml:source [ a carml:Stream ] ;
    rdf:type rml:LogicalSource .
```

```
<Sensors>
  <Sensor id="0">
    <group>loc1_tt</group>
    <type>Temperature_sensor</type>
  </Sensor>
  <Sensor id="1">
    <group>loc2_multi</group>
    <type>Temperature_sensor</type>
  </Sensor>
  <Sensor id="2">
    <group>loc1_tt</group>
    <type>Humidity_sensor</type>
  </Sensor>
  <Sensor id="3">
    <group>loc2_multi</group>
    <type>Humidity_sensor</type>
  </Sensor>
</Sensors>
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix aeros: <http://aeros-project.eu/> .

<http://aeros-project.eu/sensors/1> rdf:type <http://aeros-project.eu/types/Temperature_sensor> ;
    aeros:measurementTimestamp "2024-06-15T08:30:00"^^xsd:dateTime ;
    aeros:measurementValue "19"^^xsd:integer .

<http://aeros-project.eu/sensors/2> rdf:type <http://aeros-project.eu/types/Temperature_sensor> ;
    aeros:measurementTimestamp "2024-06-15T08:30:00"^^xsd:dateTime ;
    aeros:measurementValue "5"^^xsd:integer .

<http://aeros-project.eu/sensors/3> rdf:type <http://aeros-project.eu/types/Temperature_sensor> ;
    aeros:measurementTimestamp "2024-06-15T08:30:00"^^xsd:dateTime ;
    aeros:measurementValue "18"^^xsd:integer .

<http://aeros-project.eu/sensors/4> rdf:type <http://aeros-project.eu/types/Temperature_sensor> ;
    aeros:measurementTimestamp "2024-06-15T08:30:00"^^xsd:dateTime ;
    aeros:measurementValue "20"^^xsd:integer .
```

```
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix ql: <http://semweb.mmlab.be/ns/ql#> .
@prefix map: <http://mapping.example.com/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix ex: <http://example.com/> .
@prefix carml: <http://carml.taxonic.com/carml/> .

map:map_person_000 rml:logicalSource map:source_000 ;
    rdf:type rr:TriplesMap ;
    rdfs:label "person" ;
    rr:predicateObjectMap map:pom_000, map:pom_001, map:pom_002 ;
    rr:subjectMap map:s_000 .

map:om_000 rdf:type rr:ObjectMap ;
    rr:constant foaf:Person ;
    rr:termType rr:IRI .

map:om_001 rml:reference "firstname" ;
    rdf:type rr:ObjectMap ;
    rr:termType rr:Literal .

map:om_002 rml:reference "lastname" ;
    rdf:type rr:ObjectMap ;
    rr:termType rr:Literal .

map:pm_000 rdf:type rr:PredicateMap ;
    rr:constant rdf:type .

map:pm_001 rdf:type rr:PredicateMap ;
    rr:constant ex:name .

map:pm_002 rdf:type rr:PredicateMap ;
    rr:constant ex:lastname .

map:pom_000 rdf:type rr:PredicateObjectMap ;
    rr:objectMap map:om_000 ;
    rr:predicateMap map:pm_000 .

map:pom_001 rdf:type rr:PredicateObjectMap ;
    rr:objectMap map:om_001 ;
    rr:predicateMap map:pm_001 .

map:pom_002 rdf:type rr:PredicateObjectMap ;
    rr:objectMap map:om_002 ;
    rr:predicateMap map:pm_002 .

map:rules_000 <http://rdfs.org/ns/void#exampleResource> map:map_person_000 ;
    rdf:type <http://rdfs.org/ns/void#Dataset> .

map:s_000 rdf:type rr:SubjectMap ;
    rr:template "http://example.com/person/{id}" .

map:source_000 rml:iterator "$" ;
    rml:referenceFormulation ql:JSONPath ;
    rml:source [ a carml:Stream ] ;
    rdf:type rml:LogicalSource .
```

```
{
    "id": 1,
    "firstname": "John",
    "lastname": "Doe"
}
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix ex: <http://example.com/> .

<http://example.com/person/1> rdf:type foaf:Person ;
    ex:name "John" ;
    ex:lastname "Doe" .
```

The last example presents annotation of XML-based data coming from the same temperature sensor.

```
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix rml: <http://semweb.mmlab.be/ns/rml#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix ql: <http://semweb.mmlab.be/ns/ql#> .
@prefix map: <http://mapping.example.com/> .
@prefix aeros: <http://aeros-project.eu/> .
@prefix carml: <http://carml.taxonic.com/carml/> .

map:map_sensor_000 rml:logicalSource map:source_000 ;
    rdf:type rr:TriplesMap ;
    rdfs:label "sensor" ;
    rr:predicateObjectMap map:pom_000, map:pom_001 ;
    rr:subjectMap map:s_000 .

map:om_000 rml:reference "./group" ;
    rdf:type rr:ObjectMap ;
    rr:termType rr:Literal .

map:om_001 rdf:type rr:ObjectMap ;
    rr:template "http://aeros-project.eu/types/{./type}" ;
    rr:termType rr:IRI .

map:pm_000 rdf:type rr:PredicateMap ;
    rr:constant aeros:hasGroupId .

map:pm_001 rdf:type rr:PredicateMap ;
    rr:constant rdf:type .

map:pom_000 rdf:type rr:PredicateObjectMap ;
    rr:objectMap map:om_000 ;
    rr:predicateMap map:pm_000 .

map:pom_001 rdf:type rr:PredicateObjectMap ;
    rr:objectMap map:om_001 ;
    rr:predicateMap map:pm_001 .

map:rules_000 <http://rdfs.org/ns/void#exampleResource> map:map_sensor_000 ;
    rdf:type <http://rdfs.org/ns/void#Dataset> .

map:s_000 rdf:type rr:SubjectMap ;
    rr:template "http://aeros-project.eu/sensors/{./@id}" .

map:source_000 rml:iterator "/Sensors/Sensor" ;
    rml:referenceFormulation ql:XPath ;
    rml:source [ a carml:Stream ] ;
    rdf:type rml:LogicalSource .
```

```
<Sensors>
  <Sensor id="0">
    <group>loc1_tt</group>
    <type>Temperature_sensor</type>
  </Sensor>
  <Sensor id="1">
    <group>loc2_multi</group>
    <type>Temperature_sensor</type>
  </Sensor>
  <Sensor id="2">
    <group>loc1_tt</group>
    <type>Humidity_sensor</type>
  </Sensor>
  <Sensor id="3">
    <group>loc2_multi</group>
    <type>Humidity_sensor</type>
  </Sensor>
</Sensors>
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix aeros: <http://aeros-project.eu/> .

<http://aeros-project.eu/sensors/1> rdf:type <http://aeros-project.eu/types/Temperature_sensor> ;
    aeros:measurementTimestamp "2024-06-15T08:30:00"^^xsd:dateTime ;
    aeros:measurementValue "19"^^xsd:integer .

<http://aeros-project.eu/sensors/2> rdf:type <http://aeros-project.eu/types/Temperature_sensor> ;
    aeros:measurementTimestamp "2024-06-15T08:30:00"^^xsd:dateTime ;
    aeros:measurementValue "5"^^xsd:integer .

<http://aeros-project.eu/sensors/3> rdf:type <http://aeros-project.eu/types/Temperature_sensor> ;
    aeros:measurementTimestamp "2024-06-15T08:30:00"^^xsd:dateTime ;
    aeros:measurementValue "18"^^xsd:integer .

<http://aeros-project.eu/sensors/4> rdf:type <http://aeros-project.eu/types/Temperature_sensor> ;
    aeros:measurementTimestamp "2024-06-15T08:30:00"^^xsd:dateTime ;
    aeros:measurementValue "20"^^xsd:integer .
```

KPI 1.2.7 % data sources from aerOS scenarios to be semantically annotated and exposed via Data Fabric

Table 35: KPI 1.2.7 % data sources from aerOS scenarios to be semantically annotated and exposed via Data Fabric

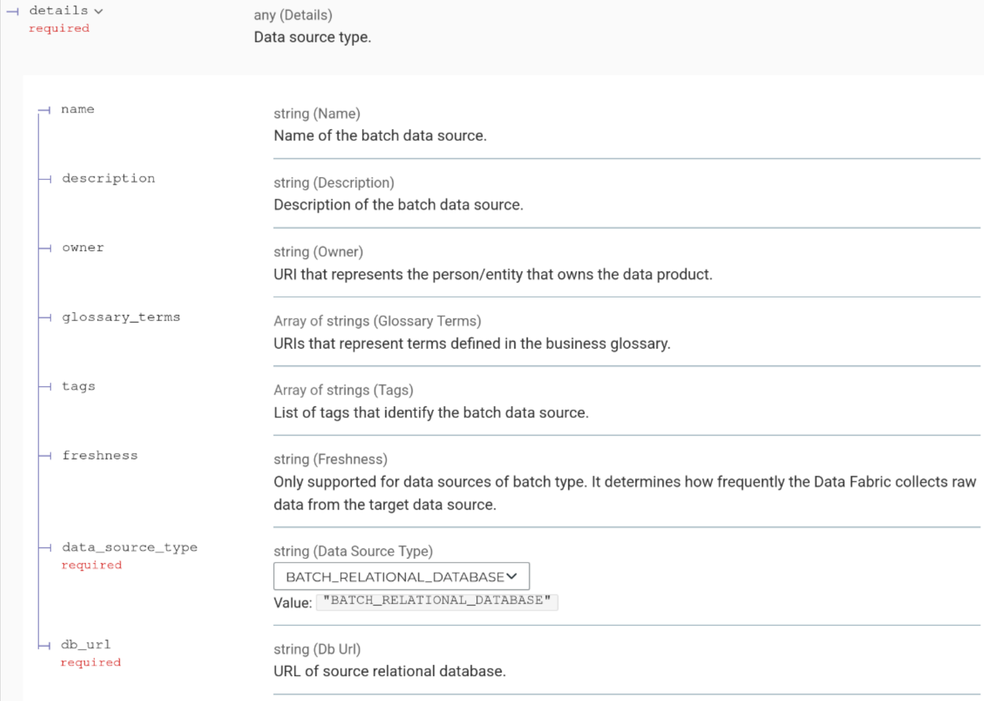
KPI ID number and partner resp.	**KPI 1.2.7**
KPI Name	**% data sources from aerOS scenarios to be semantically annotated and exposed via Data Fabric**

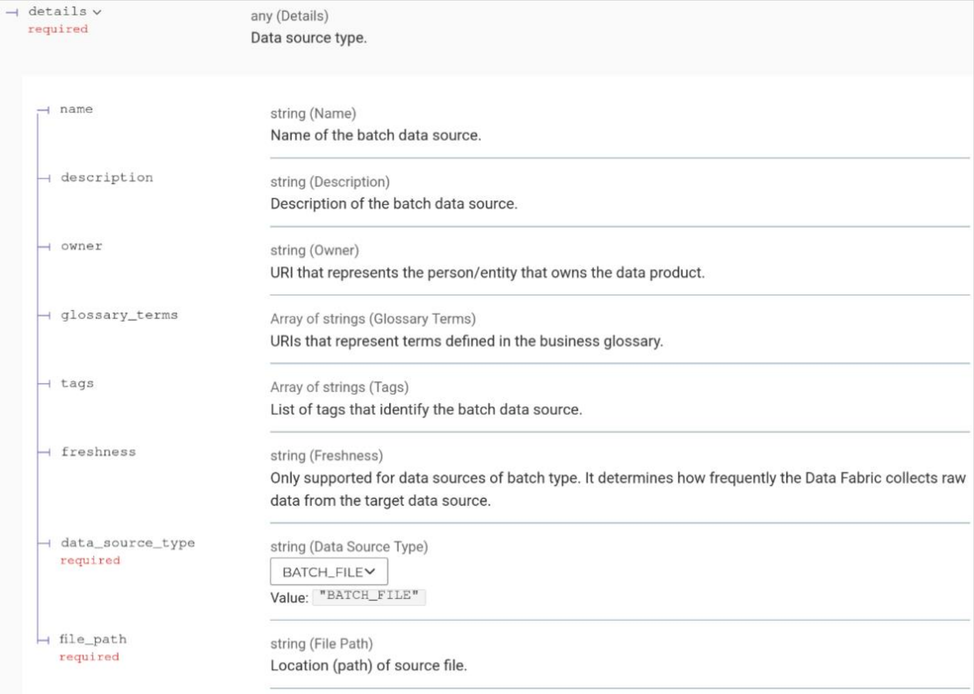
Description	Data ingested/manipulated by aerOS and expressed in NGSI-LD needs to use formal semantic models. This will enable semantic harmonization for heterogeneous data sources.		
Motivation	To achieve data shareability in aerOS.		
Target value	>50% scenarios		
Prerequisites	Data Product Manager and Data Product Pipeline components have been deployed.		
aerOS components (task)	Semantic Annotator, Semantic Translator (T4.1), Data Fabric (T4.2)		
Evaluation means	The evaluation takes into account the use cases which create/offer data and verifies if the data is properly served as NGSI-LD through aerOS Data Fabric.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***>20% scenarios***	***>45% scenarios (90%)***	***>50% scenarios (100%)***
Outcome elaboration (M38)	In aerOS, data shareability is achieved through semantically annotated data pipelines offered via the Data Fabric infrastructure, and exposed through the Context Broker. Methods and tools for defining these pipelines are available to aerOS users, and seven of the twelve aerOS use case scenarios directly incorporate their use. Pilot 1.1 directly exports production line monitoring data as JSON-LD through the Orion-LD Context Broker. Also Pilot 2 provides its data (based on Kepler metrics, describing power consumption, resource utilization, and predicted availability of green energy) through the Context Broker. Pilot 5 involves two kinds of data sources – with data that originates either from MySQL database or MQTT broker. It offers environmental parameters taken from sensors located in a smart building (such as temperature, humidity, pressure, or air quality). Additionally, based on the data coming from sensors, various monitoring, forecasting, recommendation/optimization data is made available. All the data sources are semantically annotated and exposed via Data Fabric.		

KPI 1.2.8 Support for multiple types of data sources

Table 36: KPI 1.2.8 Support for multiple types of data sources

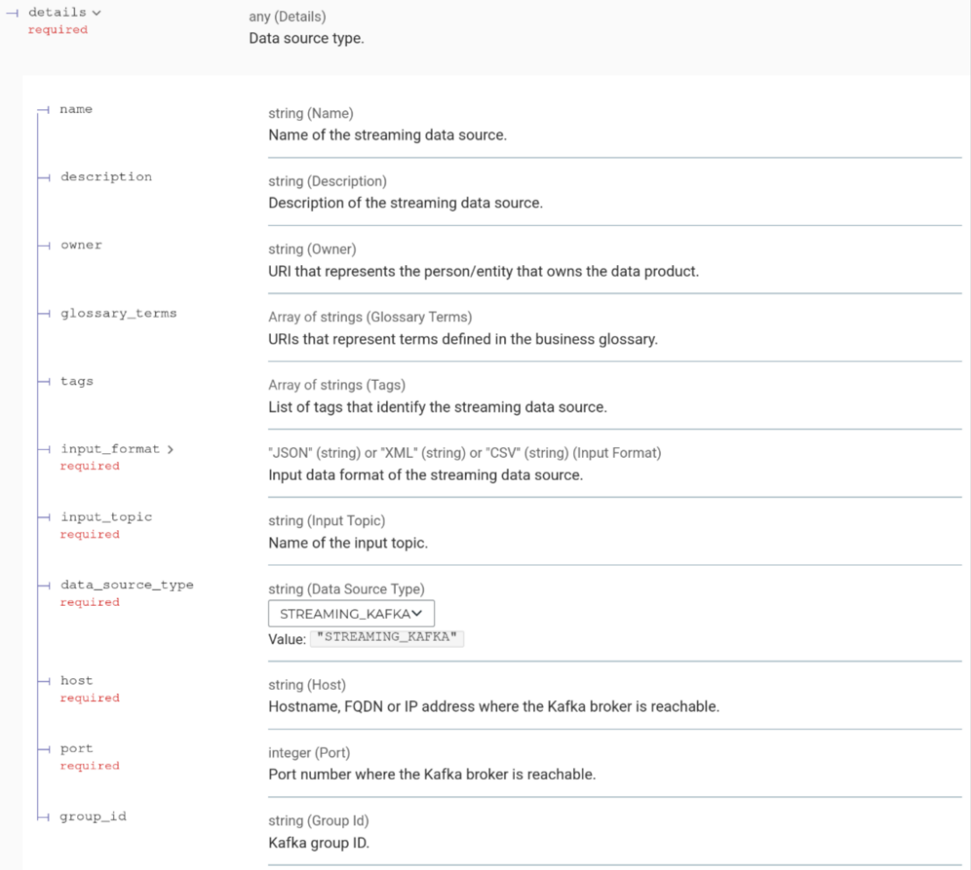
KPI ID number and partner resp.	**KPI 1.2.8**
KPI Name	**Support for multiple types of data sources**
Description	The Data Fabric can support the ingestion of data from data sources based on different protocols and data formats such Files, RDBMS, Kafka or MQTT
Motivation	Demonstrates how the Data Fabric can cope with the heterogeneity of the continuum
Target value	>=3

Prerequisites	The Data Product Manager and Data Product Pipeline components have been implemented and deployed in a scenario with multiple heterogenous data sources.		
aerOS components (task)	Semantic Annotator (T4.1), Data Fabric (T4.2)		
Evaluation means	Data Fabric's support for a specific type of data source is validated when the data product owner can onboard a data product for such data source type and the Data Fabric builds a pipeline that retrieves raw data and eventually stores it in the NGSI-LD Context Broker.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0***	***2 (66%)***	***4 (133,33 %)***
Outcome elaboration (M38)	There is a total of four (4) types of data sources supported in the aerOS Data Fabric: batch (remote files and relational databases); and streaming (Kafka and MQTT). The integration and extension of the Morph-KGC component has provided support for batch data sources. In the same way, the integration of the Semantic Annotator has provided support for streaming data sources. Hence, the Data Product Manager REST API has been extended to enable the onboarding of data products based on these types of data sources. The following snapshot depicts the documentation of the REST API to onboard batch data products from relational databases data sources:  details ⌄ required — any (Details) Data source type. name — string (Name) Name of the batch data source. description — string (Description) Description of the batch data source. owner — string (Owner) URI that represents the person/entity that owns the data product. glossary_terms — Array of strings (Glossary Terms) URIs that represent terms defined in the business glossary. tags — Array of strings (Tags) List of tags that identify the batch data source. freshness — string (Freshness) Only supported for data sources of batch type. It determines how frequently the Data Fabric collects raw data from the target data source. data_source_type required — string (Data Source Type) BATCH_RELATIONAL_DATABASE ⌄ Value: `"BATCH_RELATIONAL_DATABASE"` db_url required — string (Db Url) URL of source relational database. The following snapshot depicts the documentation of the REST API to onboard batch data products from remote files data sources:		



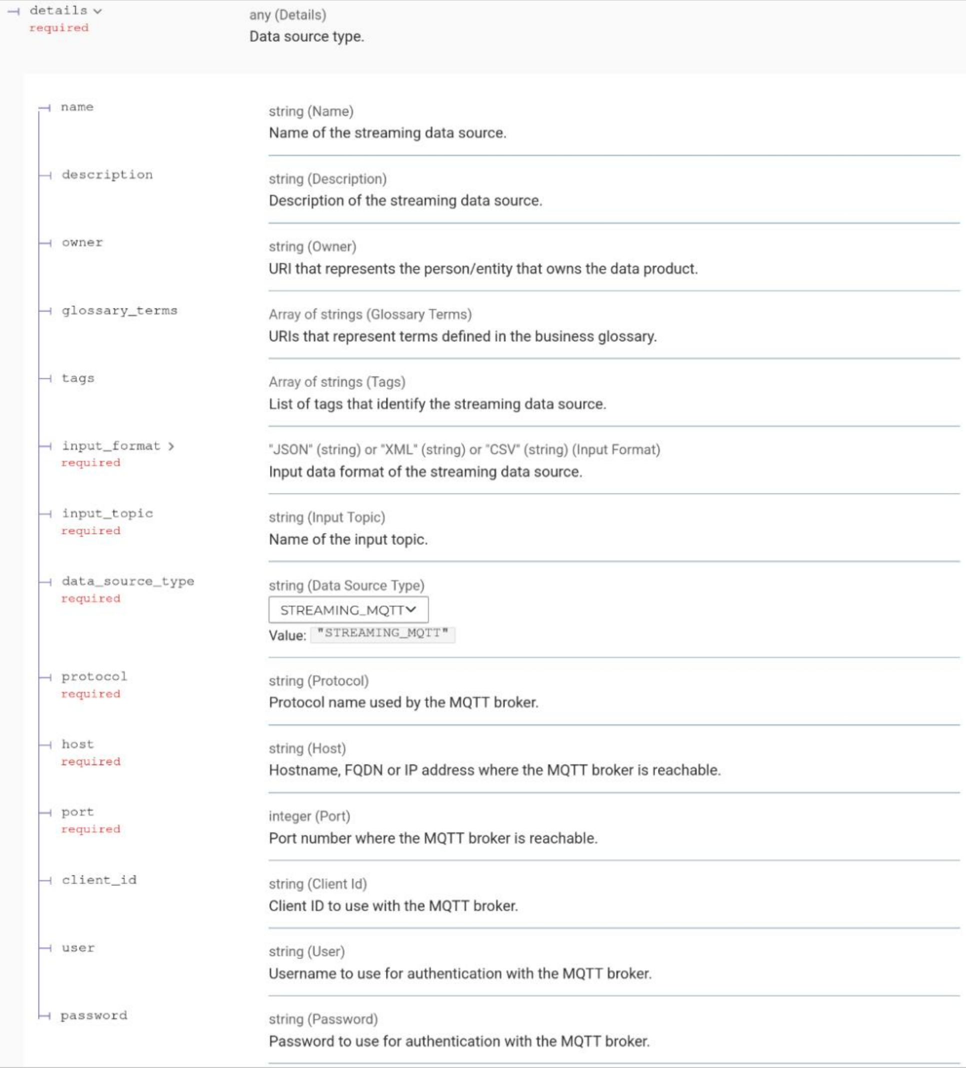
details required	any (Details) Data source type.
name	string (Name) Name of the batch data source.
description	string (Description) Description of the batch data source.
owner	string (Owner) URI that represents the person/entity that owns the data product.
glossary_terms	Array of strings (Glossary Terms) URIs that represent terms defined in the business glossary.
tags	Array of strings (Tags) List of tags that identify the batch data source.
freshness	string (Freshness) Only supported for data sources of batch type. It determines how frequently the Data Fabric collects raw data from the target data source.
data_source_type required	string (Data Source Type) BATCH_FILE Value: "BATCH_FILE"
file_path required	string (File Path) Location (path) of source file.

The following snapshot depicts the documentation of the REST API to onboard streaming data products from Kafka data sources:



details required	any (Details) Data source type.
name	string (Name) Name of the streaming data source.
description	string (Description) Description of the streaming data source.
owner	string (Owner) URI that represents the person/entity that owns the data product.
glossary_terms	Array of strings (Glossary Terms) URIs that represent terms defined in the business glossary.
tags	Array of strings (Tags) List of tags that identify the streaming data source.
input_format required	"JSON" (string) or "XML" (string) or "CSV" (string) (Input Format) Input data format of the streaming data source.
input_topic required	string (Input Topic) Name of the input topic.
data_source_type required	string (Data Source Type) STREAMING_KAFKA Value: "STREAMING_KAFKA"
host required	string (Host) Hostname, FQDN or IP address where the Kafka broker is reachable.
port required	integer (Port) Port number where the Kafka broker is reachable.
group_id	string (Group Id) Kafka group ID.

The following snapshot depicts the documentation of the REST API to onboard streaming data products from MQTT data sources:



details ⌄ required	any (Details) Data source type.
name	string (Name) Name of the streaming data source.
description	string (Description) Description of the streaming data source.
owner	string (Owner) URI that represents the person/entity that owns the data product.
glossary_terms	Array of strings (Glossary Terms) URIs that represent terms defined in the business glossary.
tags	Array of strings (Tags) List of tags that identify the streaming data source.
input_format › required	"JSON" (string) or "XML" (string) or "CSV" (string) (Input Format) Input data format of the streaming data source.
input_topic required	string (Input Topic) Name of the input topic.
data_source_type required	string (Data Source Type) STREAMING_MQTT⌄ Value: `"STREAMING_MQTT"`
protocol required	string (Protocol) Protocol name used by the MQTT broker.
host required	string (Host) Hostname, FQDN or IP address where the MQTT broker is reachable.
port required	integer (Port) Port number where the MQTT broker is reachable.
client_id	string (Client Id) Client ID to use with the MQTT broker.
user	string (User) Username to use for authentication with the MQTT broker.
password	string (Password) Password to use for authentication with the MQTT broker.

KPI 1.2.9 Data pipeline latency for data integration

Table 37: KPI 1.2.9 Data pipeline latency for data integration

KPI ID number and partner resp.	**KPI 1.2.9**
KPI Name	**Data pipeline latency for data integration**
Description	The latency added by the Data Fabric when integrating from raw data into the knowledge graph
Motivation	High latency would limit the adoption in near real-time use cases
Target value	<1 s
Prerequisites	Data Product Pipeline components implemented and deployed in a scenario with a data source like MySQL.

<table>
<tr><td>aerOS components (task)</td><td colspan="3">Semantic Annotator (T4.1), Data Fabric (T4.2)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">Using a custom developed tool, the end-to-end latency of a data product pipeline executed in the Data Fabric is measured. The total latency comprises the latencies introduced by the following steps:
Semantic annotation (<i>t1</i>)
RDF to NGSI-LD translation (<i>t2</i>)
Therefore, the latency formula looks as follows: <i>t=t1+t2</i></td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>N/A</i></td><td><i>0,40 + 0,04 = 0,44 (144%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">Semantic annotation (t1)
The experiments were designed to isolate latency of the Annotator itself, i.e. measure it without the added broker latency. Messages were generated with random size and grouped in batches of 20 messages sent at a time.

Two configurations were tested - a single channel (20 messages per batch), and 2 parallel channels (40 messages in a batch, but split in 2, thus, 20 per channel). The following table captures the results obtained for each configuration:
<table>
<tr><th>Metric</th><th>1 channel</th><th>2 channels</th></tr>
<tr><td>count</td><td>20</td><td>40</td></tr>
<tr><td>average</td><td>5.95</td><td>4</td></tr>
<tr><td>Max</td><td>15</td><td>10</td></tr>
<tr><td>Min</td><td>4</td><td>2</td></tr>
<tr><td>Median</td><td>5</td><td>3</td></tr>
<tr><td>Mode</td><td>4</td><td>3</td></tr>
</table>
In summary, the measured latency was 5,95 and 4 on average, and the median was 5 and 3. All measurements are represented in milliseconds.

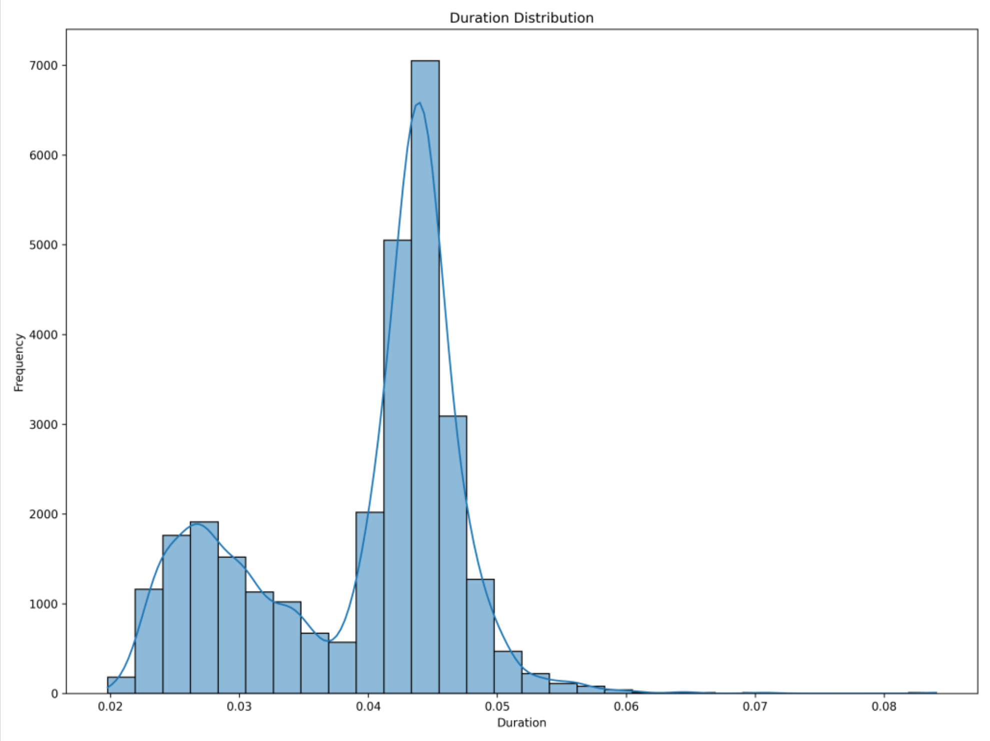
RDF to NGSI-LD translation (<i>t2</i>)
The latencies present during the ingestion of raw data in the mapping engine (e.g., Morph-KGC) as well as the materialization of the resulting NGSI-LD data in the NGSI-LD Context Broker.

Regarding the RDF to NGSI-LD translation, the testing environment has been i7 with 20 cores and 64Gb or RAM, each test consist of reading 30000 RDF triples in a Red Panda queue (the data is previously stored in Red Panda queues, so the writing does not interfere with the processing of the messages) and measuring the time that each message takes from its read the queue until it is confirmed by the Orion-LD – Latency per message.

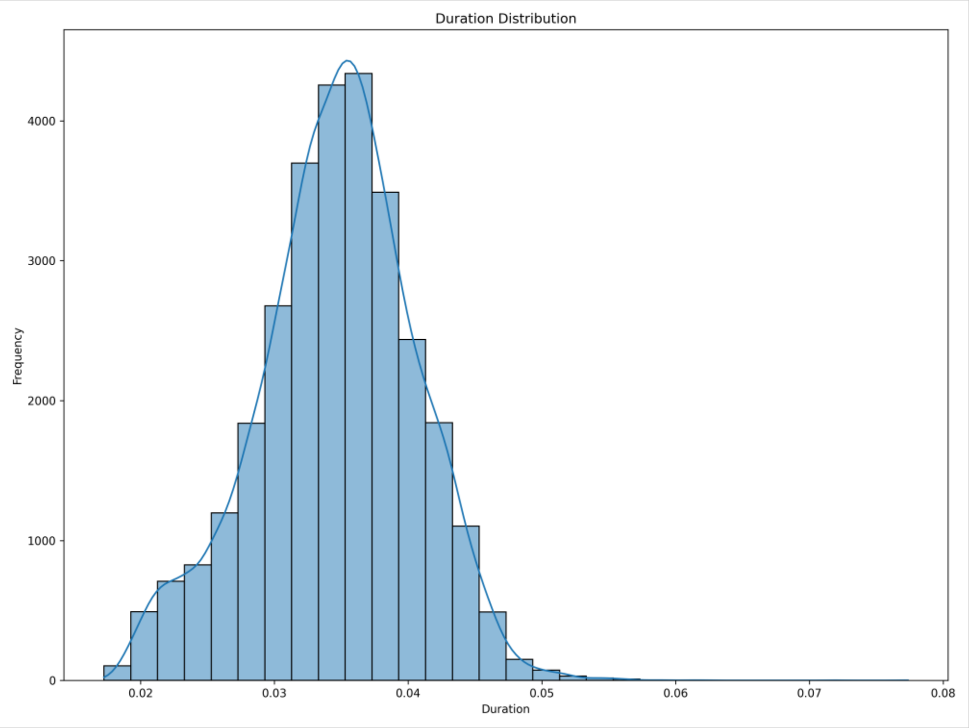
Regarding the latency in RDF to NGSI-LD translation, including communications with NGSI-LD Broker (bear in mind that more than one request to the broker can be done</td></tr>
</table>

per RDF file), these are the observed measurements in seconds:

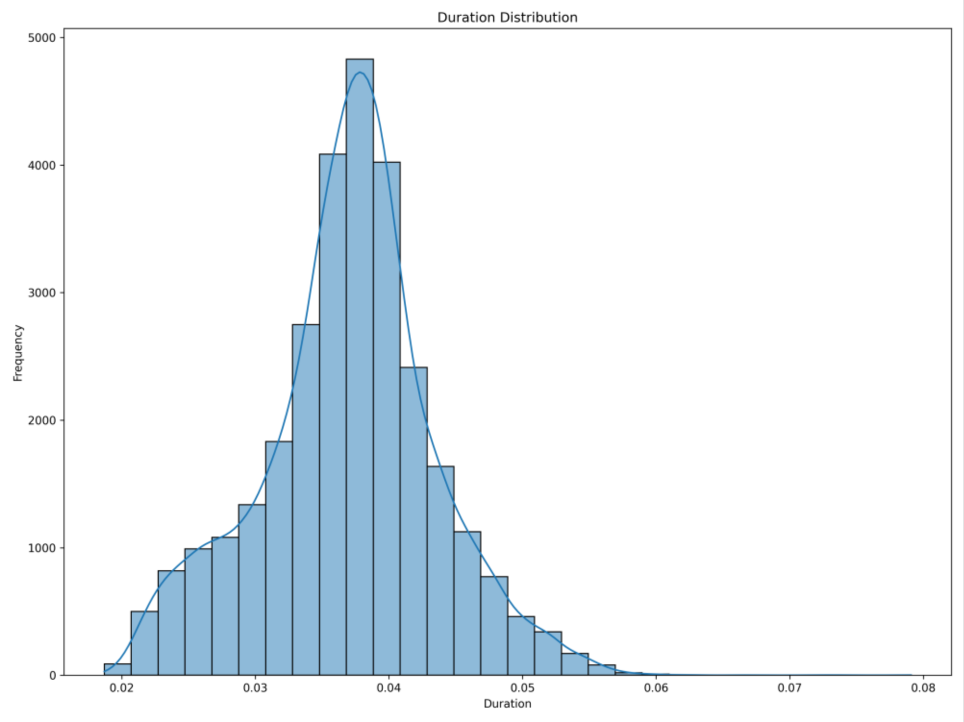
1 Core
Mean:0.04
Median: 0.04
25th percentile: 0.03
75th percentile: 0.04
90th percentile: 0.05
95th percentile: 0.05
99th percentile: 0.05
Total TIME: 1256.58 => 23.87 RDFs/second



2 Cores:
Mean: 0.03
Median: 0.03
25th percentile: 0.03
75th percentile: 0.04
90th percentile: 0.04
95th percentile: 0.04
99th percentile: 0.05
Total TIME: 521.29 => 57.5 RDFs/second

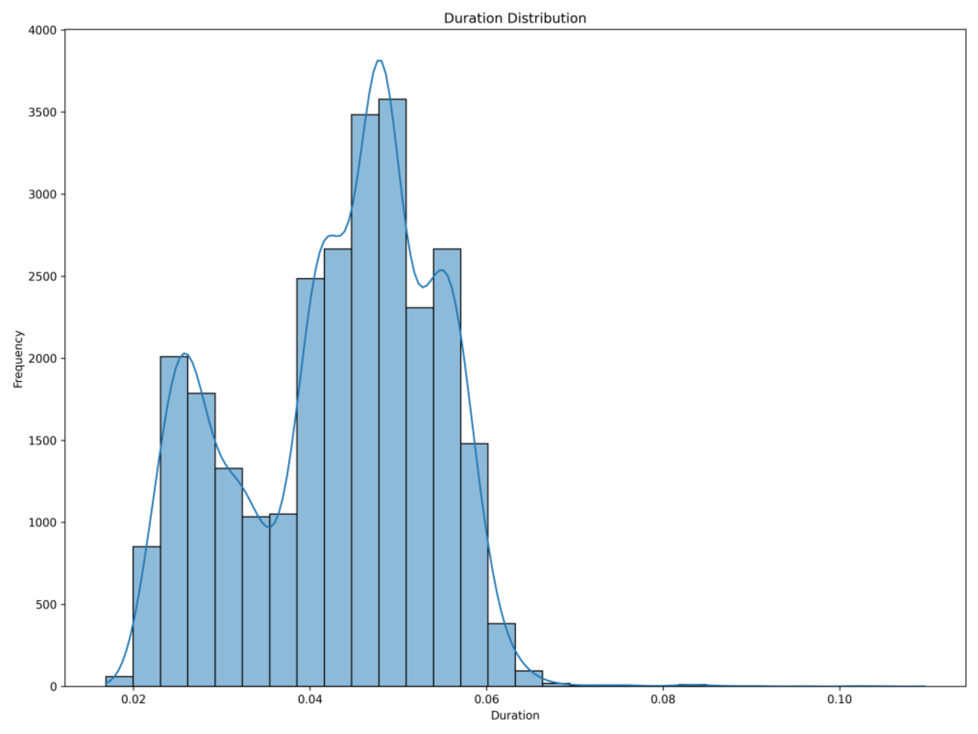


5 Cores:
Mean: 0.04
Median: 0.04
25th percentile: 0.03
75th percentile: 0.04
90th percentile: 0.04
95th percentile: 0.05
99th percentile: 0.05
Total TIME: 223.09 => 134.47 RDFs/second

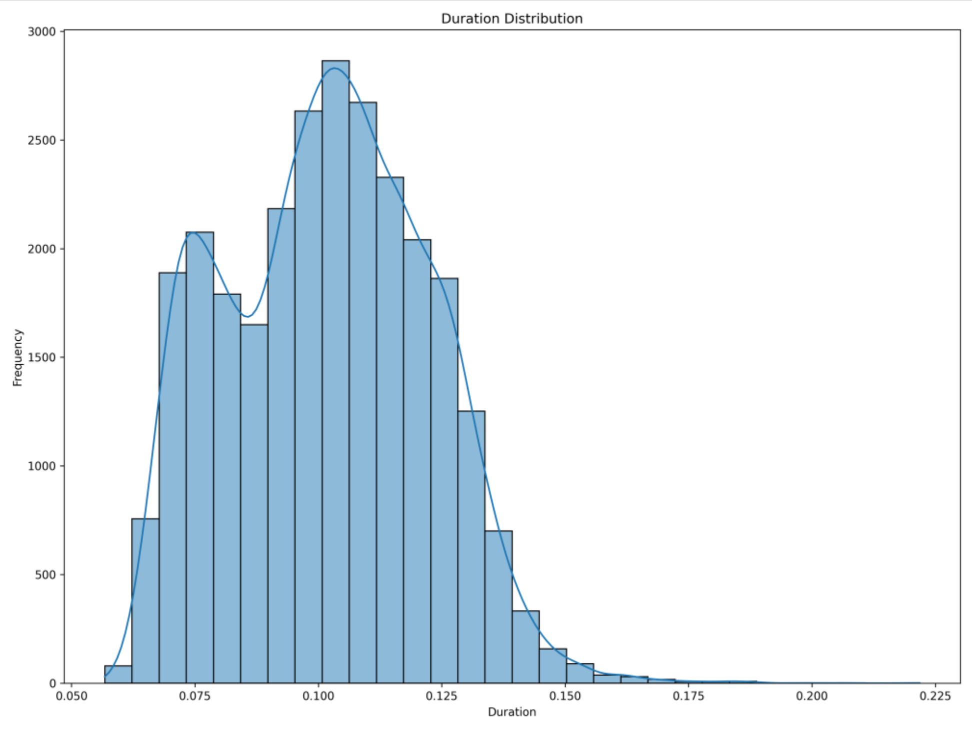


15 Cores:

Mean: 0.04
Median: 0.05
25th percentile: 0.03
75th percentile: 0.05
90th percentile: 0.06
95th percentile: 0.06
99th percentile: 0.06
TIME: 87.84 =>   => 341.53 RDFs/second



20 Cores
Mean: 0.10
Median: 0.10
25th percentile: 0.09
75th percentile: 0.12
90th percentile: 0.13
95th percentile: 0.13
99th percentile: 0.15
TIME: 154.59   => 194.06 RDFs/second

	 As we can see, the latency per message is quite stable if the number of cores devoted to the translator is equal or less than 75% of the physical cores in the testing server. --- It is empirically observed that if we configure the translator to work with more than 75% of cores, the performance degrades. The best times has been observed using 15 cores of 20 available.

KPI 1.2.10 Simultaneous data pipeline execution

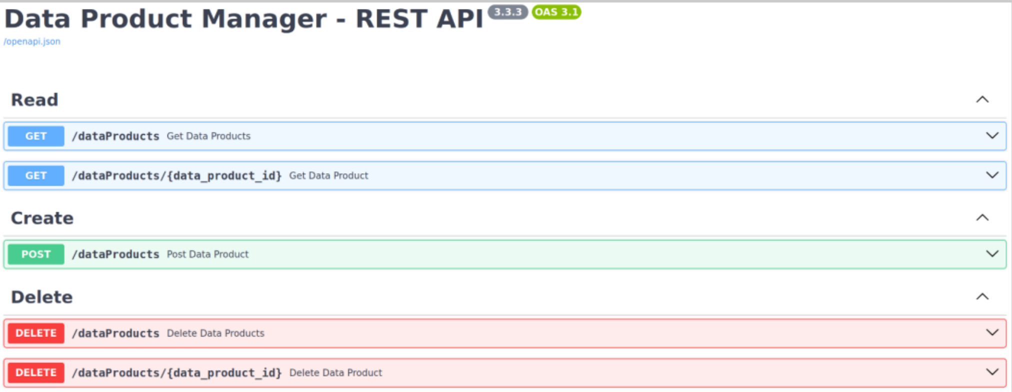
Table 38: KPI 1.2.10 Simultaneous data pipeline execution

KPI ID number and partner resp.	**KPI 1.2.10**
KPI Name	**Simultaneous data pipeline execution**
Description	Maximum number of concurrent data pipelines running in the Data Fabric with guaranteed performance.
Motivation	The Data Fabric is expected to simultaneously handle multiple data flows.
Target value	5
Prerequisites	Data Product Manager and Data Product Pipeline components implemented and deployed in a scenario with multiple batch data sources like MySQL or streaming data sources like Kafka.
aerOS components (task)	Semantic Annotator (T4.1), Data Fabric (T4.2)
Evaluation means	The Data Product Manager must handle the lifecycle of data product pipeline that ingest and integrate data from multiple data sources. To do so, the management of data

	products by data owners is enabled via a REST API implemented in the Data Product Manager. A preliminary version of this REST API is documented in the official aerOS documentation: https://docs.aeros-project.eu/en/latest/data/fabric/data_product_manager.html#interacting-with-the-data-product-manager		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***N/A***	***15 (300 %)***

Outcome elaboration (M38)

The data product lifecycle management feature in the Data Product Manager enables the creation, retrieval and deletion of data products, as depicted in the following figure, taken from Swagger UI (REST API documentation):



Data Product Manager - REST API 3.3.3 OAS 3.1

/openapi.json

Read

GET `/dataProducts` Get Data Products

GET `/dataProducts/{data_product_id}` Get Data Product

Create

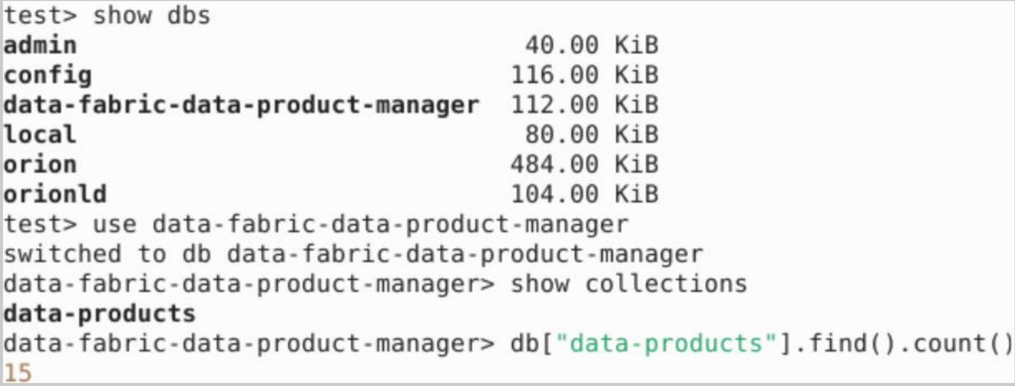
POST `/dataProducts` Post Data Product

Delete

DELETE `/dataProducts` Delete Data Products

DELETE `/dataProducts/{data_product_id}` Delete Data Product

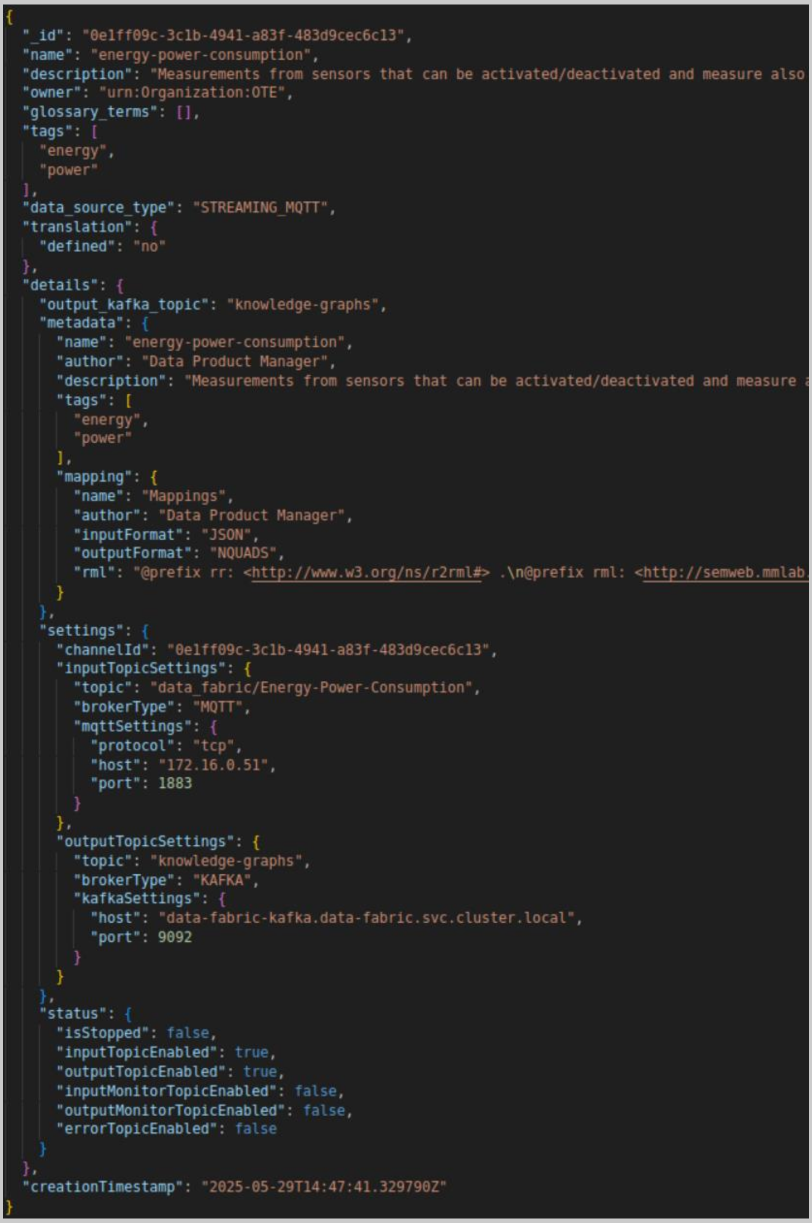
The Data Product Manager stores metadata of all onboarded data products in a MongoDB collection. In the following picture, a query to count the number of onboarded data products in Pilot 5's application domain is depicted:



```
test> show dbs
admin                              40.00 KiB
config                            116.00 KiB
data-fabric-data-product-manager  112.00 KiB
local                              80.00 KiB
orion                             484.00 KiB
orionld                           104.00 KiB
test> use data-fabric-data-product-manager
switched to db data-fabric-data-product-manager
data-fabric-data-product-manager> show collections
data-products
data-fabric-data-product-manager> db["data-products"].find().count()
15
```

As it can be seen, 15 data products were onboarded, with their corresponding pipelines working in parallel.

Finally, in the following snapshot, metadata of one data product is depicted as retrieved from the REST API (cropped to fill):

```
{
  "_id": "0e1ff09c-3c1b-4941-a83f-483d9cec6c13",
  "name": "energy-power-consumption",
  "description": "Measurements from sensors that can be activated/deactivated and measure also
  "owner": "urn:Organization:OTE",
  "glossary_terms": [],
  "tags": [
    "energy",
    "power"
  ],
  "data_source_type": "STREAMING_MQTT",
  "translation": {
    "defined": "no"
  },
  "details": {
    "output_kafka_topic": "knowledge-graphs",
    "metadata": {
      "name": "energy-power-consumption",
      "author": "Data Product Manager",
      "description": "Measurements from sensors that can be activated/deactivated and measure
      "tags": [
        "energy",
        "power"
      ],
      "mapping": {
        "name": "Mappings",
        "author": "Data Product Manager",
        "inputFormat": "JSON",
        "outputFormat": "NQUADS",
        "rml": "@prefix rr: <http://www.w3.org/ns/r2rml#> .\n@prefix rml: <http://semweb.mmlab
      }
    },
    "settings": {
      "channelId": "0e1ff09c-3c1b-4941-a83f-483d9cec6c13",
      "inputTopicSettings": {
        "topic": "data_fabric/Energy-Power-Consumption",
        "brokerType": "MQTT",
        "mqttSettings": {
          "protocol": "tcp",
          "host": "172.16.0.51",
          "port": 1883
        }
      },
      "outputTopicSettings": {
        "topic": "knowledge-graphs",
        "brokerType": "KAFKA",
        "kafkaSettings": {
          "host": "data-fabric-kafka.data-fabric.svc.cluster.local",
          "port": 9092
        }
      }
    },
    "status": {
      "isStopped": false,
      "inputTopicEnabled": true,
      "outputTopicEnabled": true,
      "inputMonitorTopicEnabled": false,
      "outputMonitorTopicEnabled": false,
      "errorTopicEnabled": false
    }
  },
  "creationTimestamp": "2025-05-29T14:47:41.329790Z"
}
```

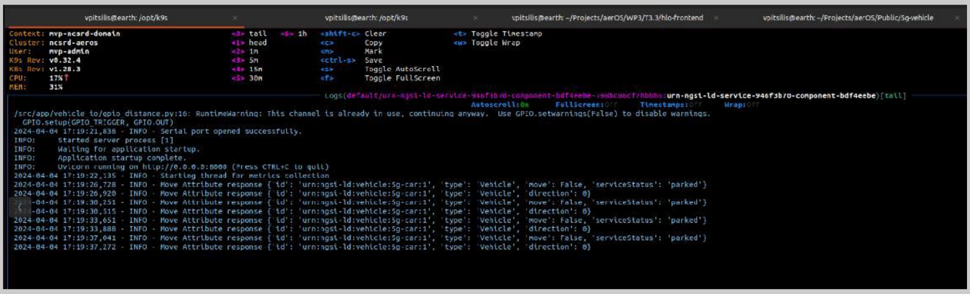
aerOS service fabric

KPI 1.3.1 Number of VNF/NetApps to improve performance and self-* network reconfiguration (KVI-2.3)

Table 39: KPI 1.3.1 Number of VNF/NetApps to improve performance and self- network reconfiguration (KVI-2.3)*

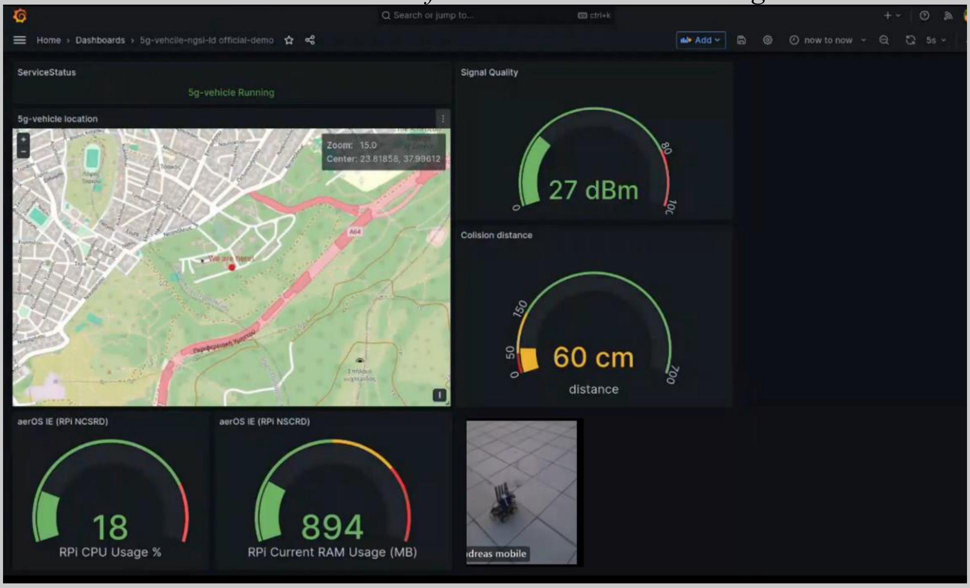
KPI ID number and partner resp.	**KPI 1.3.1**
KPI Name	**Number of VNF/NetApps to improve performance and self-* network reconfiguration**

Description	The use of VNFs (or Network Apps as defined in 3GPP) aims at improving the performance of the self-configuration of the network. This is achieved by interfacing each Network App on the one hand with the native APIs of the 5G network, and, on the other hand with application itself through business APIs, enhancing either the performance of the service provision or the network configuration itself. This KPI follows the paradigm of the 3GPP SA6 standardisation activities, where specific Network Apps are realized as Vertical Application Enablers (VAEs), improving the performance for supporting services of vertical industries, or the network configuration.		
Motivation	The higher the number that a specific application or service is interfacing with NetApps, the more innovative the specific application/service is becoming, because it integrates features that are not currently available or possible with a simple OTT provision approach.		
Target value	> 6 Services/Applications that are interfacing with at least one NetApp		
Prerequisites	aerOS domain set up complete with at least one IE connected over 5G network.		
aerOS components (task)	Ingress (T3.1), TLS (T3.1), HLO (T3.3), LLO (T3.3), API Gateway (T3.4), Data Fabric (T4.2)		
Evaluation means	Logs showing 5G metrics (QoS and GPS location) exposed in aerOS Data Fabric will be monitored and presented.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***3 (50%)***	***6 (100%)***
Outcome elaboration (M38)	In **D5.5**, three Network Applications (NetApps) were validated within the MVP phase, interfacing with aerOS services on top of the 5G vehicle in the NCSRD domain. These included the provision of QoS and GPS location data to the Command and Control (C2) and Visualization applications, as well as integration with Orion-LD for data dissemination across the continuum. Building on this foundation, the current deliverable (**D5.6**) extends the validation to **six active NetApps** deployed across the pilots. Two additional VNFs — **NEF Event Monitoring** and **AsSessionWithQoS** — were deployed on aerOS and integrated with the 5G core through native NEF APIs, also exposed via **OpenCAPIF** for API discoverability. These components enable QoS management, mobility and event monitoring, and context-aware data exposure to aerOS and vertical applications. Furthermore, in **Pilot 4**, industrial straddle carriers connected to aerOS through **4G LTE-A routers (Teltonika RUT series)** provide continuous telemetry and network visibility. This setup, integrated into aerOS's monitoring framework, constitutes an additional **monitoring-oriented Network Application** supporting **self-diagnosis and self-optimization** in network communication. Together, these developments confirm that aerOS now hosts **six validated Network Applications**, enhancing performance and enabling self-* network reconfiguration across heterogeneous connectivity environments. *MVP C2 over 5G*		

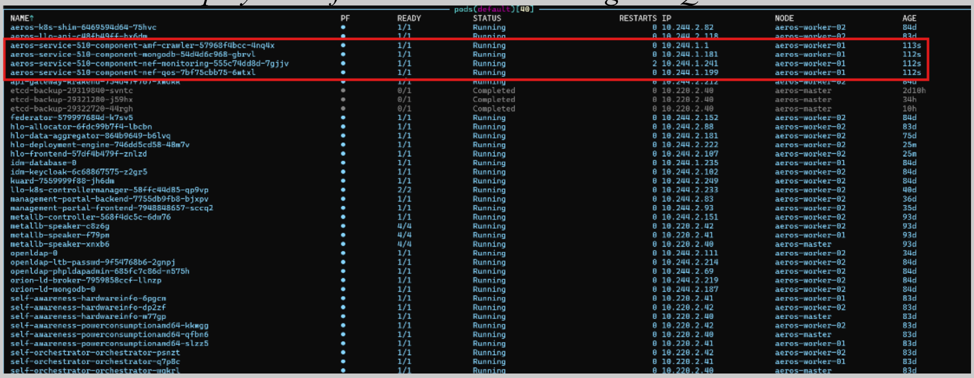


```
Context: mvp-ncsrd-domain          <0> tail   <6> 1h   <shift-c> Clear              <t> Toggle Timestamp
Cluster: ncsrd-aeros               <1> head            <c>       Copy               <w> Toggle Wrap
User:    mvp-admin                 <2> 1m              <m>       Mark
K9s Rev: v0.32.4                   <3> 5m              <ctrl-s>  Save
K8s Rev: v1.28.3                   <4> 15m             <s>       Toggle AutoScroll
CPU:     17%↑                      <5> 30m             <f>       Toggle FullScreen
MEM:     31%
 Logs(default/urn-ngsi-ld-service-946f3b70-component-bdf4eebe-798bc86cf7-bbbhs:urn-ngsi-ld-service-946f3b70-component-bdf4eebe)[tail]
                                     Autoscroll:On      FullScreen:Off      Timestamps:Off      Wrap:Off
/src/app/vehicle_io/gpio_distance.py:16: RuntimeWarning: This channel is already in use, continuing anyway.  Use GPIO.setwarnings(False) to disable warnings.
  GPIO.setup(GPIO_TRIGGER, GPIO.OUT)
2024-04-04 17:19:21,838 - INFO - Serial port opened successfully.
INFO:     Started server process [1]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO:     Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
2024-04-04 17:19:22,135 - INFO - Starting thread for metrics collection
2024-04-04 17:19:26,728 - INFO - Move Attribute response {'id': 'urn:ngsi-ld:vehicle:5g-car:1', 'type': 'Vehicle', 'move': False, 'serviceStatus': 'parked'}
2024-04-04 17:19:26,920 - INFO - Move Attribute response {'id': 'urn:ngsi-ld:vehicle:5g-car:1', 'type': 'Vehicle', 'direction': 0}
2024-04-04 17:19:30,251 - INFO - Move Attribute response {'id': 'urn:ngsi-ld:vehicle:5g-car:1', 'type': 'Vehicle', 'move': False, 'serviceStatus': 'parked'}
2024-04-04 17:19:30,515 - INFO - Move Attribute response {'id': 'urn:ngsi-ld:vehicle:5g-car:1', 'type': 'Vehicle', 'direction': 0}
2024-04-04 17:19:33,651 - INFO - Move Attribute response {'id': 'urn:ngsi-ld:vehicle:5g-car:1', 'type': 'Vehicle', 'move': False, 'serviceStatus': 'parked'}
2024-04-04 17:19:33,888 - INFO - Move Attribute response {'id': 'urn:ngsi-ld:vehicle:5g-car:1', 'type': 'Vehicle', 'direction': 0}
2024-04-04 17:19:37,041 - INFO - Move Attribute response {'id': 'urn:ngsi-ld:vehicle:5g-car:1', 'type': 'Vehicle', 'move': False, 'serviceStatus': 'parked'}
2024-04-04 17:19:37,272 - INFO - Move Attribute response {'id': 'urn:ngsi-ld:vehicle:5g-car:1', 'type': 'Vehicle', 'direction': 0}
```

MVP visualization for vehicle status monitoring

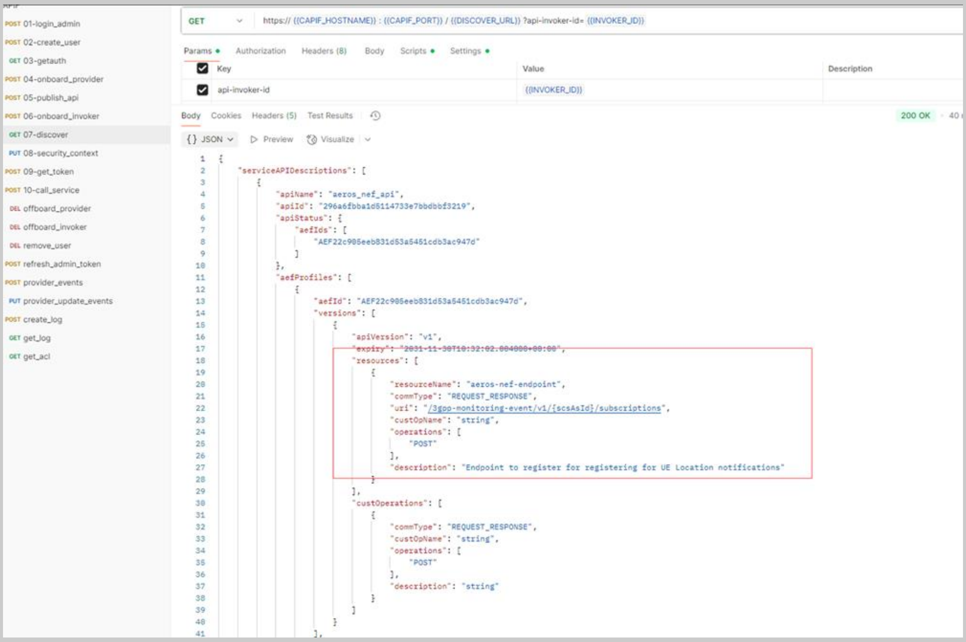


Deployment of NEF Event Monitoring and QoS VNFs



```
                                                          pods(default)[40]
NAME↑                                                PF    READY   STATUS          RESTARTS IP              NODE              AGE
aeros-k8s-shim-6469594d64-75hvc                      ●     1/1     Running                0 10.244.2.82     aeros-worker-02   84d
aeros-llo-api-c48fb49ff-bx6dm                        ●     1/1     Running                0 10.244.2.118    aeros-worker-02   83d
aeros-service-510-component-amf-crawler-57968f4bcc-4nq4x ● 1/1     Running                0 10.244.1.1      aeros-worker-01   113s
aeros-service-510-component-mongodb-54d4d6c968-gbrvl ●     1/1     Running                0 10.244.1.181    aeros-worker-01   112s
aeros-service-510-component-nef-monitoring-555c74dd8d-7gjjv ● 1/1  Running                2 10.244.1.241    aeros-worker-01   112s
aeros-service-510-component-nef-qos-7bf75cbb75-6mtxl ●     1/1     Running                0 10.244.1.199    aeros-worker-01   112s
api-gateway-krakend-734d47f767-xm6kk                 ●     1/1     Running                0 10.244.2.212    aeros-worker-02   84d
etcd-backup-29319840-svntc                           ●     0/1     Completed              0 10.220.2.40     aeros-master      2d10h
etcd-backup-29321280-j59hx                           ●     0/1     Completed              0 10.220.2.40     aeros-master      34h
etcd-backup-29322720-44rgh                           ●     0/1     Completed              0 10.220.2.40     aeros-master      10h
federator-579997684d-k7sv5                           ●     1/1     Running                0 10.244.2.152    aeros-worker-02   84d
hlo-allocator-6fdc99b7f4-lbcbn                       ●     1/1     Running                0 10.244.2.88     aeros-worker-02   83d
hlo-data-aggregator-864b9649-b6lvq                   ●     1/1     Running                0 10.244.2.181    aeros-worker-02   75d
hlo-deployment-engine-746dd5cd58-48m7v               ●     1/1     Running                0 10.244.2.222    aeros-worker-02   25m
hlo-frontend-57df4b479f-znlzd                        ●     1/1     Running                0 10.244.2.107    aeros-worker-02   25m
idm-database-0                                       ●     1/1     Running                0 10.244.1.235    aeros-worker-01   84d
idm-keycloak-6c68867575-z2gr5                        ●     1/1     Running                0 10.244.2.102    aeros-worker-02   84d
kuard-7559999f88-jh6dm                               ●     1/1     Running                0 10.244.2.249    aeros-worker-02   84d
llo-k8s-controllermanager-58ffc44d85-qp9vp           ●     2/2     Running                0 10.244.2.233    aeros-worker-02   40d
management-portal-backend-7755db9fb8-bjxpv           ●     1/1     Running                0 10.244.2.83     aeros-worker-02   36d
management-portal-frontend-7948848657-sccq2          ●     1/1     Running                0 10.244.2.93     aeros-worker-02   35d
metallb-controller-568f4dc5c-6dw76                   ●     1/1     Running                0 10.244.2.151    aeros-worker-02   93d
metallb-speaker-c8z6g                                ●     4/4     Running                0 10.220.2.42     aeros-worker-02   93d
metallb-speaker-f79pm                                ●     4/4     Running                0 10.220.2.41     aeros-worker-01   93d
metallb-speaker-xnxb6                                ●     4/4     Running                0 10.220.2.40     aeros-master      93d
openldap-0                                           ●     1/1     Running                0 10.244.2.111    aeros-worker-02   34d
openldap-ltb-passwd-9f54768b6-2gnpj                  ●     1/1     Running                0 10.244.2.214    aeros-worker-02   84d
openldap-phpldapadmin-685fc7c86d-n575h               ●     1/1     Running                0 10.244.2.69     aeros-worker-02   84d
orion-ld-broker-7959858ccf-llnzp                     ●     1/1     Running                0 10.244.2.219    aeros-worker-02   84d
orion-ld-mongodb-0                                   ●     1/1     Running                0 10.244.2.187    aeros-worker-02   84d
self-awareness-hardwareinfo-6pgcm                    ●     1/1     Running                0 10.220.2.41     aeros-worker-01   83d
self-awareness-hardwareinfo-dp2zf                    ●     1/1     Running                0 10.220.2.42     aeros-worker-02   83d
self-awareness-hardwareinfo-m77gp                    ●     1/1     Running                0 10.220.2.40     aeros-master      83d
self-awareness-powerconsumptionamd64-kkwgg           ●     1/1     Running                0 10.220.2.42     aeros-worker-02   83d
self-awareness-powerconsumptionamd64-qfbn6           ●     1/1     Running                0 10.220.2.40     aeros-master      83d
self-awareness-powerconsumptionamd64-slzz5           ●     1/1     Running                0 10.220.2.41     aeros-worker-01   83d
self-orchestrator-orchestrator-psnzt                 ●     1/1     Running                0 10.220.2.42     aeros-worker-02   83d
self-orchestrator-orchestrator-q7p8c                 ●     1/1     Running                0 10.220.2.41     aeros-worker-01   83d
self-orchestrator-orchestrator-wqkrl                 ●     1/1     Running                0 10.220.2.40     aeros-master      83d
```

CAPIF discovery of NEF APIs:

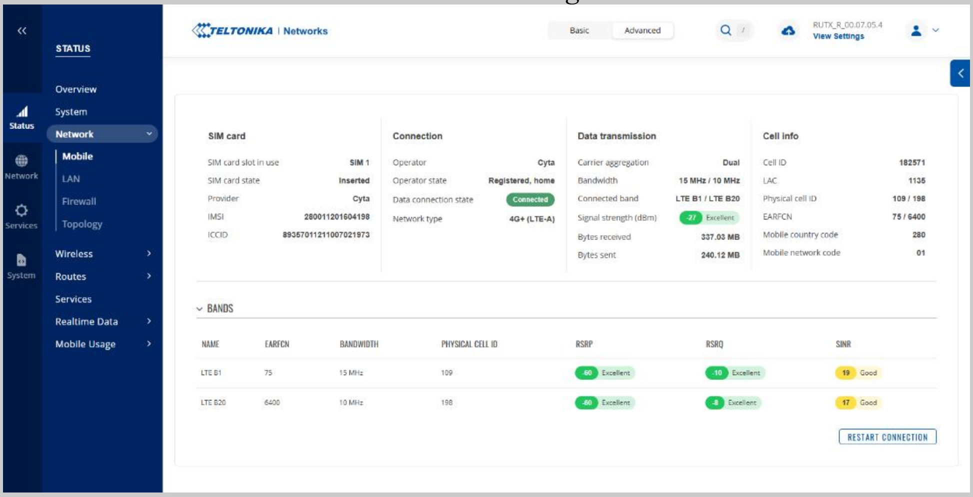


GET https:// {{CAPIF_HOSTNAME}} : {{CAPIF_PORT}} / {{DISCOVER_URL}} ?api-invoker-id= {{INVOKER_ID}}

	Key	Value	Description
☑	api-invoker-id	{{INVOKER_ID}}	

```
{
    "serviceAPIDescriptions": [
        {
            "apiName": "aeros_nef_api",
            "apiId": "296a6fbba1d5114733e7bbdbbf3219",
            "apiStatus": {
                "aefIds": [
                    "AEF22c985eeb831d53a5451cdb3ac947d"
                ]
            },
            "aefProfiles": [
                {
                    "aefId": "AEF22c985eeb831d53a5451cdb3ac947d",
                    "versions": [
                        {
                            "apiVersion": "v1",
                            "expiry": "2031-11-30T10:32:02.004000+00:00",
                            "resources": [
                                {
                                    "resourceName": "aeros-nef-endpoint",
                                    "commType": "REQUEST_RESPONSE",
                                    "uri": "/3gpp-monitoring-event/v1/{scsAsId}/subscriptions",
                                    "custOpName": "string",
                                    "operations": [
                                        "POST"
                                    ],
                                    "description": "Endpoint to register for registering for UE Location notifications"
                                }
                            ],
                            "custOperations": [
                                {
                                    "commType": "REQUEST_RESPONSE",
                                    "custOpName": "string",
                                    "operations": [
                                        "POST"
                                    ],
                                    "description": "string"
                                }
                            ]
                        }
                    ],
```

Teltonika LTE-A monitoring screenshots



SIM card

SIM card slot in use	SIM 1
SIM card state	Inserted
Provider	Cyta
IMSI	280011201604198
ICCID	89357011211007021973

Connection

Operator	Cyta
Operator state	Registered, home
Data connection state	Connected
Network type	4G+ (LTE-A)

Data transmission

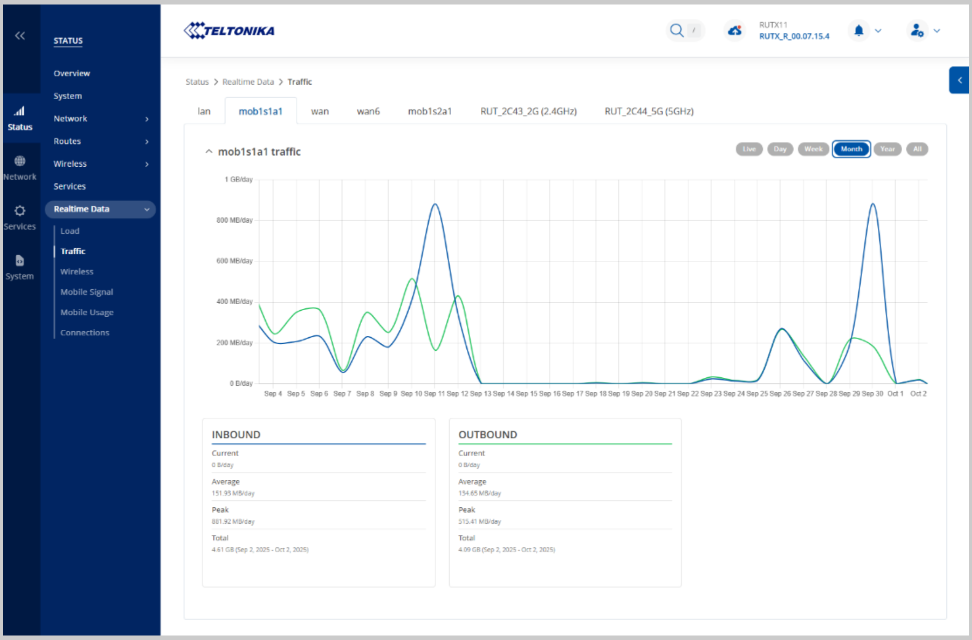
Carrier aggregation	Dual
Bandwidth	15 MHz / 10 MHz
Connected band	LTE B1 / LTE B20
Signal strength (dBm)	-27 Excellent
Bytes received	337.03 MB
Bytes sent	240.12 MB

Cell info

Cell ID	182571
LAC	1135
Physical cell ID	109 / 198
EARFCN	75 / 6400
Mobile country code	280
Mobile network code	01

BANDS

NAME	EARFCN	BANDWIDTH	PHYSICAL CELL ID	RSRP	RSRQ	SINR
LTE B1	75	15 MHz	109	-60 Excellent	-10 Excellent	19 Good
LTE B20	6400	10 MHz	198	-60 Excellent	-8 Excellent	17 Good

	 INBOUND Current 0 B/day Average 151.93 MB/day Peak 881.92 MB/day Total 4.61 GB (Sep 2, 2025 - Oct 2, 2025) OUTBOUND Current 0 B/day Average 134.65 MB/day Peak 515.41 MB/day Total 4.09 GB (Sep 2, 2025 - Oct 2, 2025)

KPI 1.3.2 Total services delivered by aerOS

Table 40: KPI 1.3.2 Total services delivered by aerOS

KPI ID number and partner resp.	**KPI 1.3.2**		
KPI Name	**Total services delivered by aerOS**		
Description	This KPI refers to the total number of basic and auxiliary services that aerOS consists of and delivers respectively to the users. This collection of services provides to the aerOS users the capability to interact with the Data Fabric plane of the Meta-OS, allowing them to discover IoT data, to deploy a distributed service across the continuum and to manage the aerOS nodes that can join the continuum.		
Motivation	This KPIs denotes the complexity of the aerOS Meta-OS, but at the same time reflects also the complexity of the defined Meta-OS in terms of features and services offered to the aerOS users towards realizing the IoT-edge-cloud continuum in its full potential.		
Target value	*> 50* aux and basic aerOS services deployed		
Prerequisites	HW & Infrastructure integrated as aerOS IEs within aerOS domains		
aerOS components (task)	All aerOS basic and auxiliary services from WP3 and WP4		
Evaluation means	Every aerOS continuum leader will provide a list of the aerOS basic and auxiliary services that are deployed along all their inside domains. The number will be listed in D5.5, and D5.6. The proof of this list will be shown as K9s screenshots in D5.4 per pilot.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**

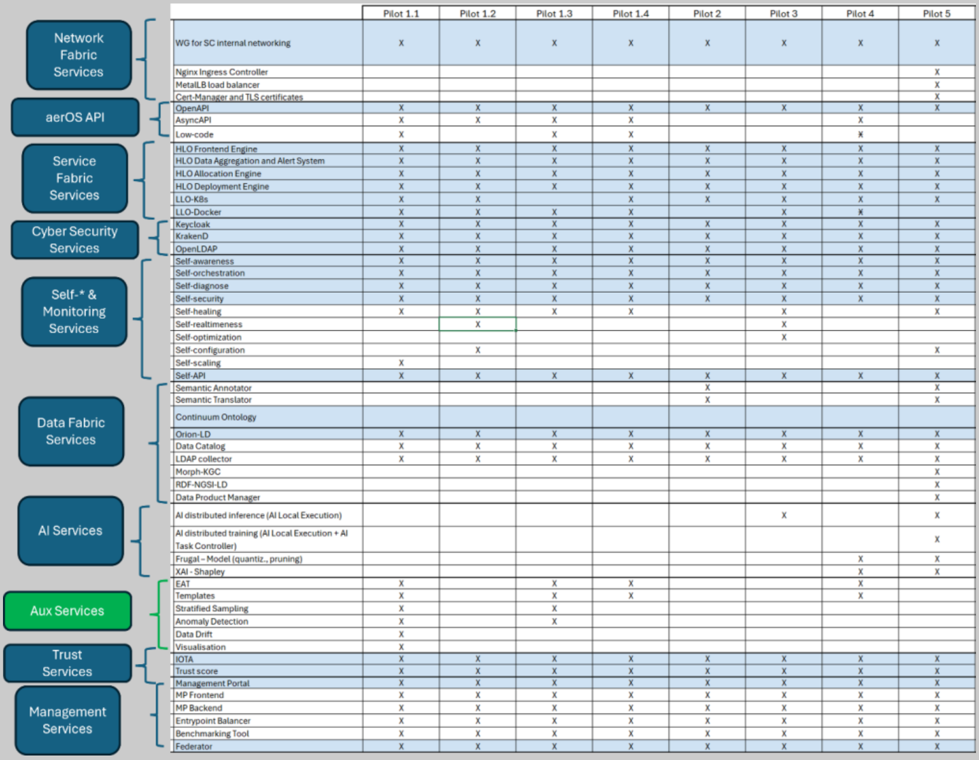
Measured value (% achieved)	**N/A**	**20 (40%)**	**> 100 (>200%)**

Outcome elaboration (M38)

Already in D5.5 we reported that 20 aerOS services had been deployed as part of the MVP phase, including LLO/HLO, Orion-LD, Federator, the Self-* components, EAT, CI/CD (flux), CNI, Certification Manager, Ingress and LoadBalancer, Krakend, LDAP/IdM, and Management framework.

This report now extends the validation to include the pilot deployments. As shown in the following table, services are classified into broader categories and the deployment status per pilot is indicated. Evidence of these deployments, including outputs from tools such as "kubectl get" and k9s, has already been presented in "D5.4 – Use cases deployment and implementation (2)" and is referenced here as proof of the deployments carried out at pilot sites.

It should be noted that each pilot typically integrates more than one aerOS domain, and each domain includes the set of baseline services. For this reason, the cumulative service count represented in the table underestimates the actual deployment footprint. Based on the integration across all pilots, the total number of services delivered by aerOS is far above 50 (target value), which demonstrates both the scalability and maturity of the platform in supporting a wide range of functionality across the continuum. A rough but conservative estimation places the total number significantly beyond this threshold.

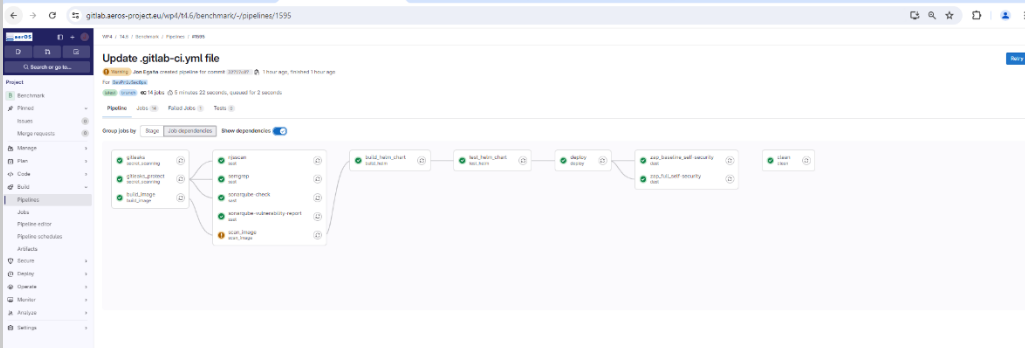


		Pilot 1.1	Pilot 1.2	Pilot 1.3	Pilot 1.4	Pilot 2	Pilot 3	Pilot 4	Pilot 5
Network Fabric Services	WG for SC internal networking	X	X	X	X	X	X	X	X
	Nginx Ingress Controller								X
	MetalLB load balancer								X
	Cert-Manager and TLS certificates								X
aerOS API	OpenAPI	X	X	X	X	X	X	X	X
	AsyncAPI	X	X	X	X			X	
	Low-code	X		X	X			X	
Service Fabric Services	HLO Frontend Engine	X	X	X	X	X	X	X	X
	HLO Data Aggregation and Alert System	X	X	X	X	X	X	X	X
	HLO Allocation Engine	X	X	X	X	X	X	X	X
	HLO Deployment Engine	X	X	X	X	X	X	X	X
	LLO-K8s	X	X		X	X	X	X	X
	LLO-Docker	X	X	X	X		X	X	
Cyber Security Services	Keycloak	X	X	X	X	X	X	X	X
	KrakenD	X	X	X	X	X	X	X	X
	OpenLDAP	X	X	X	X	X	X	X	X
Self-* & Monitoring Services	Self-awareness	X	X	X	X	X	X	X	X
	Self-orchestration	X	X	X	X	X	X	X	X
	Self-diagnose	X	X	X	X	X	X	X	X
	Self-security	X	X	X	X	X	X	X	X
	Self-healing	X	X	X	X		X		X
	Self-realtimeness		X				X		
	Self-optimization						X		
	Self-configuration		X						X
	Self-scaling	X							
	Self-API	X	X	X	X	X	X	X	X
Data Fabric Services	Semantic Annotator					X			X
	Semantic Translator					X			X
	Continuum Ontology								
	Orion-LD	X	X	X	X	X	X	X	X
	Data Catalog	X	X	X	X	X	X	X	X
	LDAP collector	X	X	X	X	X	X	X	X
	Morph-KGC								X
	RDF-NGSI-LD								X
	Data Product Manager								X
AI Services	AI distributed inference (AI Local Execution)						X		X
	AI distributed training (AI Local Execution + AI Task Controller)								X
	Frugal – Model (quantiz., pruning)							X	X
	XAI - Shapley							X	X
Aux Services	EAT	X		X	X			X	
	Templates	X		X	X			X	
	Stratified Sampling	X		X					
	Anomaly Detection	X		X					
	Data Drift	X							
	Visualisation	X							
Trust Services	IOTA	X	X	X	X	X	X	X	X
	Trust score	X	X	X	X	X	X	X	X
Management Services	Management Portal	X	X	X	X	X	X	X	X
	MP Frontend	X	X	X	X	X	X	X	X
	MP Backend	X	X	X	X	X	X	X	X
	Entrypoint Balancer	X	X	X	X	X	X	X	X
	Benchmarking Tool	X	X	X	X	X	X	X	X
	Federator	X	X	X	X	X	X	X	X

This cumulative table demonstrates that deployments span all major service categories and pilots, with each pilot typically integrating multiple aerOS domains. As each domain brings a complete set of baseline services, the total number of services delivered across the continuum is conservatively estimated to exceed fifty, thereby confirming the achievement of the KPI.

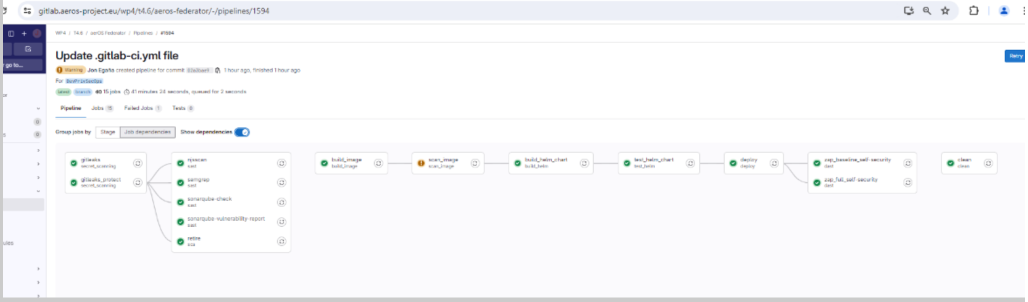
KPI 1.3.3 # of successful CI/CD pipelines implemented in the project

Table 41: KPI 1.3.3 # of successful CI/CD pipelines implemented in the project

KPI ID number and partner resp.	**KPI 1.3.3**		
KPI Name	**# of successful CI/CD pipelines implemented in the project**		
Description	The number of repositories that have successfully completed the Continuous Integration (CI) / Continuous Deployment (CD) test designed for the project.		
Motivation	Successfully passing the CI/CD tests designed for the project leads to the conclusion that the developed code complies with the security and privacy requirements defined in the project and its correct functioning in the deployment environment.		
Target value	>4		
Prerequisites	Implementation of all the phases that were presented in the DevPrivSecOps methodology in D2.5.		
aerOS components (task)	All aerOS software components are invited to implement the DevPrivSecOps methodology presented in deliverable D2.4 with the tools and guidelines provided in D2.5.		
Evaluation means	The evaluation will assess whether all phases of the methodology have been successfully completed in the development of aerOS components. This can be seen in the GitLab pipeline of the repository, and all phases must be completed successfully (the tests of each phase must be completed successfully).		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*N/A*	*2 (50%)*	*7 (175%)*
Outcome elaboration (M38)	The full pipeline methodology has now been successfully implemented in seven aerOS components (seven repositories): **Benchmark** (link to the pipeline):  gitlab.aeros-project.eu/wp4/t4.6/benchmark/-/pipelines/1595 Update .gitlab-ci.yml file Jon Egaña created pipeline for commit 1 hour ago, finished 1 hour ago 14 jobs 5 minutes 22 seconds, queued for 2 seconds 		

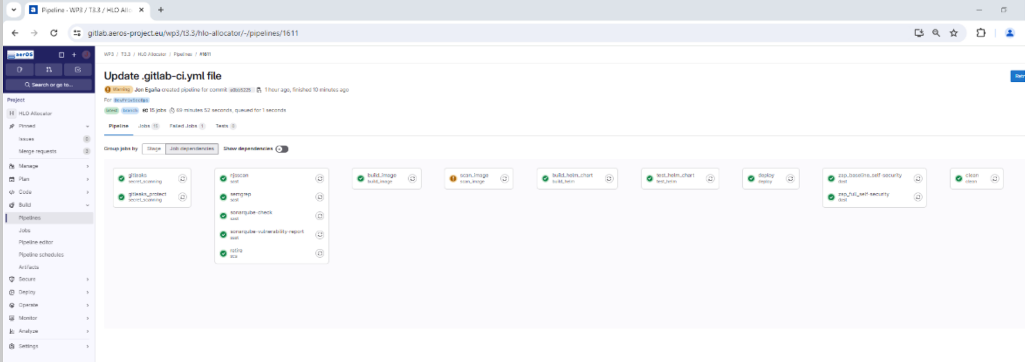
In this case it can be seen that the pipeline has a warning detected in the *Image Scan* stage that needs to be analysed and remediated.

Federator (link to the pipeline):



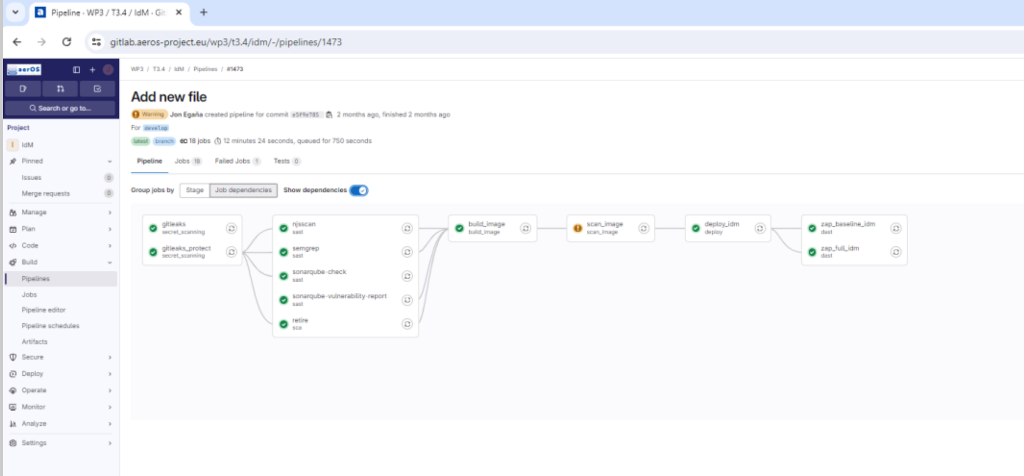
In this case it can be seen that the pipeline has a warning detected in the *Image Scan* stage that needs to be analysed and remediated.

HLO (link to the pipeline):



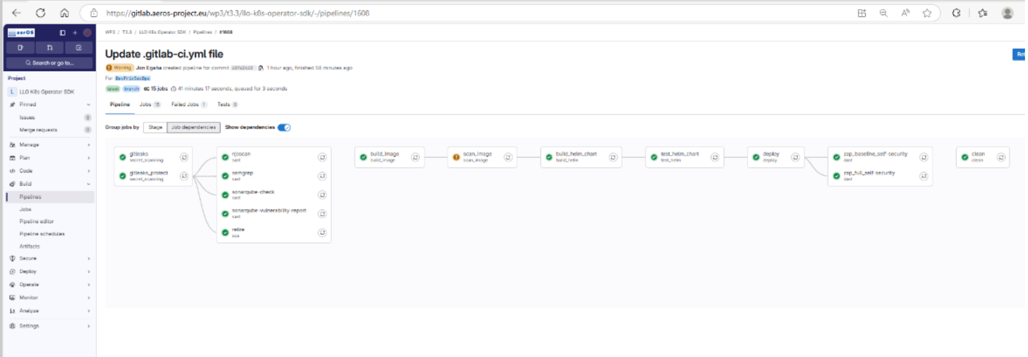
In this case it can be seen that the pipeline has a warning detected in the *Image Scan* stage that needs to be analysed and remediated.

IdM (link to the pipeline):



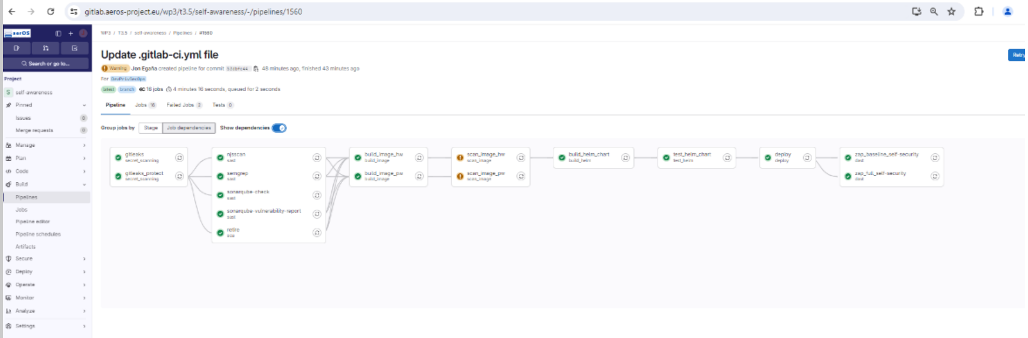
In this case it can be seen that the pipeline has a warning detected in the *Image Scan* stage that needs to be analysed and remediated.

LLO (link to the pipeline):



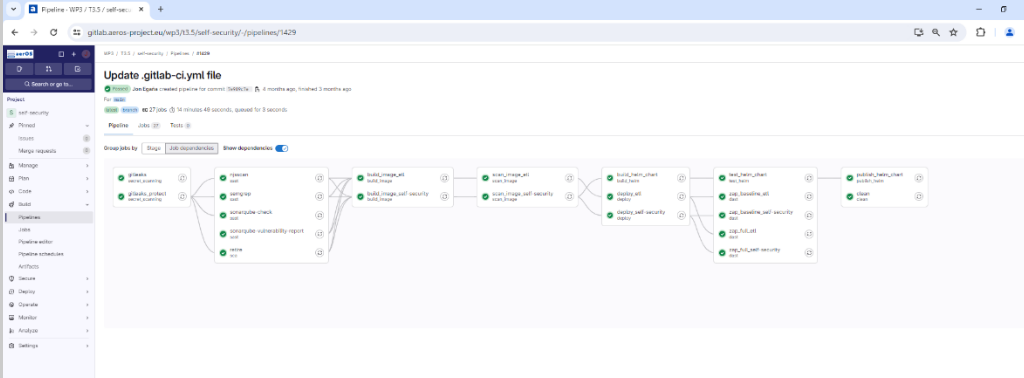
In this case it can be seen that the pipeline has a warning detected in the *Image Scan* stage that needs to be analysed and remediated.

Self-awareness (link to the pipeline):



In this case it can be seen that the pipeline has a warning detected in the *Image Scan* stage that needs to be analysed and remediated.

self-security (link to the pipeline):



The CI/CD pipeline has been successfully completed, providing evidence that the code meets security and privacy needs.

KPI 1.3.4 Number of different service components running in different domains that form functional services thanks to aerOS network components

Table 42: KPI 1.3.4 Number of different service components running in different domains that form functional services thanks to aerOS network components

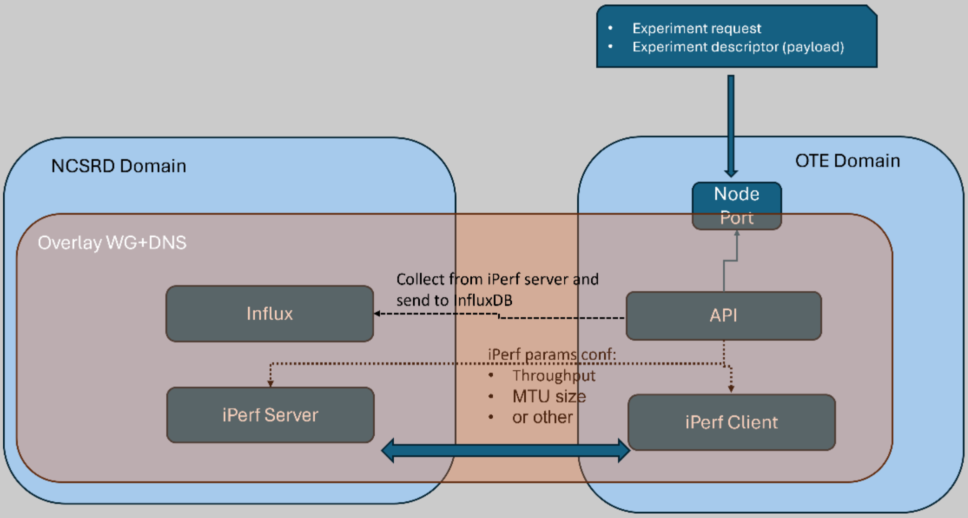
<table>
<tr><th>KPI ID number and partner resp.</th><th colspan="3">KPI 1.3.4</th></tr>
<tr><td>KPI Name</td><td colspan="3">Number of different service components running in different domains that form functional services thanks to aerOS network components</td></tr>
<tr><td>Description</td><td colspan="3">This KPI refers to the number of independent software components, which if unified under the same continuum structure, then altogether form a distributed service.</td></tr>
<tr><td>Motivation</td><td colspan="3">This KPI is important because it quantifies the number of components that a continuum realization connects in order to form a distributed service provision, which without the existence of the Meta-OS would not have been possible.</td></tr>
<tr><td>Target value</td><td colspan="3">At least 4 components to be interfaced for the realization of a pilot.</td></tr>
<tr><td>Prerequisites</td><td colspan="3">At least 2 aerOS domains setups complete.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">Ingress (T3.1), TLS (T3.1), HLO (T3.3), LLO (T3.3), API Gateway (T3.4), Self-configurator (T3.5), Data Fabric (T4.2)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">Screenshots, of management and reporting tools, which will explicitly show the deployment domains of service components. K9s will be used to provide evidence that service components are deployed beyond the borders of a single domain.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>2 (50%)</i></td><td><i>4 (100%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">To validate this KPI, a distributed service was deployed across two aerOS domains, demonstrating the ability of aerOS to orchestrate and interconnect heterogeneous service components into a single, functional service over the continuum. The selected use case involves a network performance measurement service built on top of the iperf toolchain, chosen because of its ability to quantify network throughput and latency — key indicators of connectivity performance within aerOS-managed environments.
The service consists of four distinct components:
1. An iperf server hosted in Domain A,
2. An iperf client hosted in Domain B,
3. An orchestrator component exposing a REST API to start, stop, and retrieve measurement experiments, and
4. An InfluxDB time-series database that receives and stores performance data, providing timeline-based visualization of network metrics.
These components were deployed in two geographically separate aerOS domains and connected through an isolated overlay network managed by aerOS network services. The overlay is based on WireGuard tunnels, dynamically configured and secured through aerOS, ensuring seamless inter-domain communication among all service</td></tr>
</table>

components. Internal DNS resolution and configuration maps were automatically generated to maintain interconnectivity using overlay-specific hostnames.

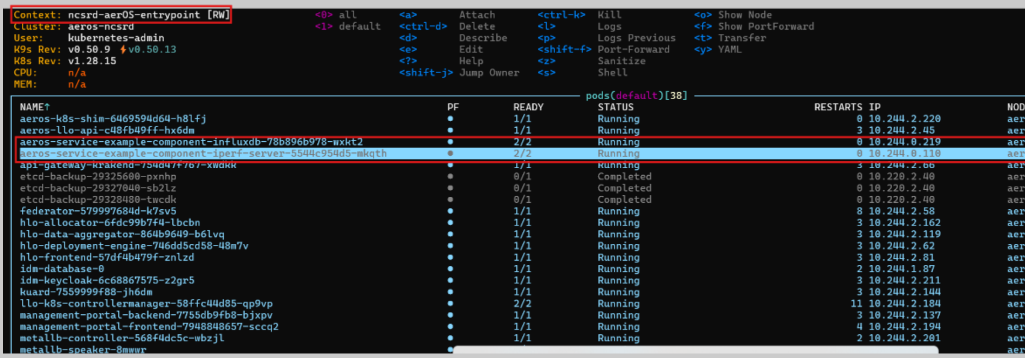
During validation, successful WireGuard handshakes, inter-component pings, and API calls confirmed that all four components interacted correctly through the overlay network, forming a unified distributed service. The orchestrator successfully triggered measurement sessions between the iperf agents, collected results, and stored them in the InfluxDB instance, proving that functional service composition across multiple domains is achieved through aerOS.

This deployment validates that aerOS enables the creation and operation of multi-domain functional services by abstracting network complexity, ensuring service-level communication, and supporting secure overlay creation. Through this experiment, four interdependent service components were effectively orchestrated and executed across two domains, verifying the KPI.

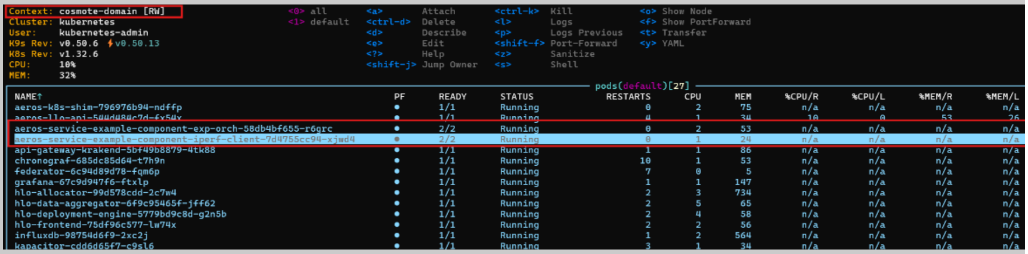
Schematic of service topology



k9s deployment screenshots — two components per domain

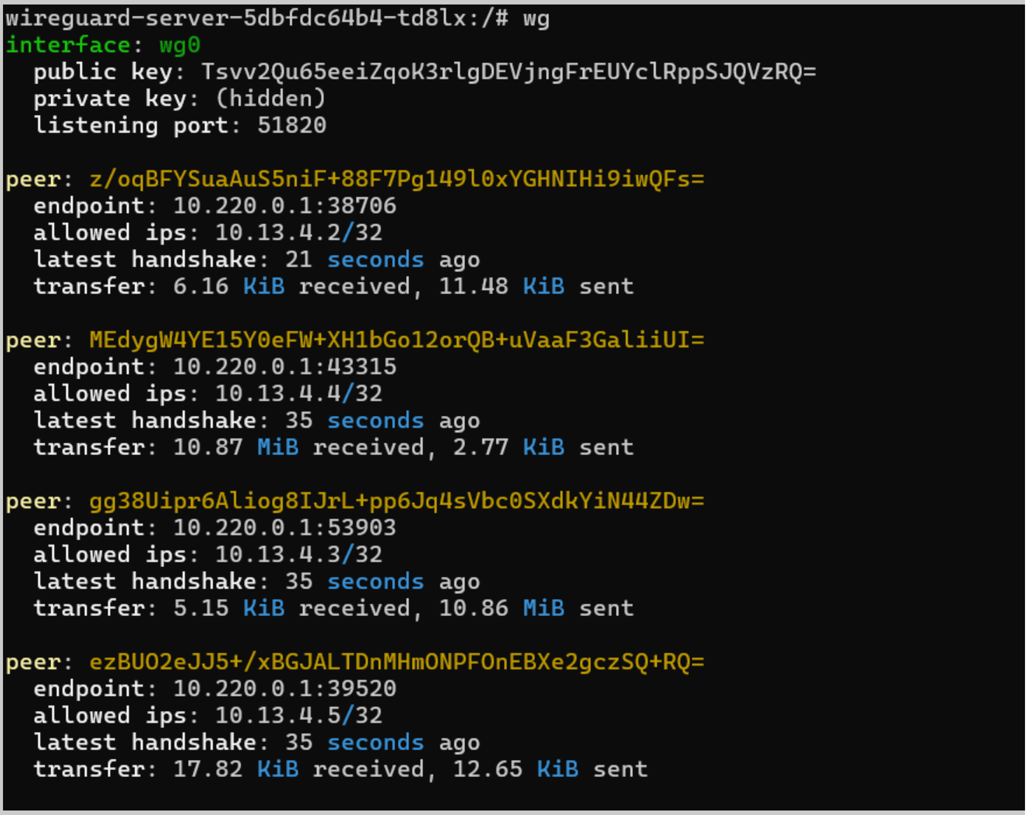


```
Context: ncsrd-aerOS-entrypoint [RW]          <0> all      <a>         Attach       <ctrl-k>   Kill            <o> Show Node
Cluster: aeros-ncsrd                          <1> default  <ctrl-d>    Delete       <l>        Logs            <f> Show PortForward
User:    kubernetes-admin                                  <d>         Describe     <p>        Logs Previous   <t> Transfer
K9s Rev: v0.50.9 ⚡v0.50.13                                 <e>         Edit         <shift-f>  Port-Forward    <y> YAML
K8s Rev: v1.28.15                                          <?>         Help         <z>        Sanitize
CPU:     n/a                                               <shift-j>   Jump Owner   <s>        Shell
MEM:     n/a
                                                                      pods(default)[38]
 NAME↑                                                   PF       READY      STATUS                          RESTARTS IP               NOD
 aeros-k8s-shim-6469594d64-h8lfj                         ●        1/1        Running                                0 10.244.2.220     aer
 aeros-llo-api-c48fb49ff-hx6dm                           ●        1/1        Running                                3 10.244.2.45      aer
 aeros-service-example-component-influxdb-78b896b978-wxkt2 ●      2/2        Running                                0 10.244.0.219     aer
 aeros-service-example-component-iperf-server-5544c954d5-mkqth ●  2/2        Running                                0 10.244.0.110     aer
 api-gateway-krakend-754d47f767-xwdkk                    ●        1/1        Running                                3 10.244.2.66      aer
 etcd-backup-29325600-pxnhp                              ●        0/1        Completed                              0 10.220.2.40      aer
 etcd-backup-29327040-sb2lz                              ●        0/1        Completed                              0 10.220.2.40      aer
 etcd-backup-29328480-twcdk                              ●        0/1        Completed                              0 10.220.2.40      aer
 federator-579997684d-k7sv5                              ●        1/1        Running                                8 10.244.2.58      aer
 hlo-allocator-6fdc99b7f4-lbcbn                          ●        1/1        Running                                3 10.244.2.162     aer
 hlo-data-aggregator-864b9649-b6lvq                      ●        1/1        Running                                3 10.244.2.119     aer
 hlo-deployment-engine-746dd5cd58-48m7v                  ●        1/1        Running                                3 10.244.2.62      aer
 hlo-frontend-57df4b479f-znlzd                           ●        1/1        Running                                3 10.244.2.81      aer
 idm-database-0                                          ●        1/1        Running                                2 10.244.1.87      aer
 idm-keycloak-6c68867575-z2gr5                           ●        1/1        Running                                3 10.244.2.211     aer
 kuard-7559999f88-jh6dm                                  ●        1/1        Running                                3 10.244.2.144     aer
 llo-k8s-controllermanager-58ffc44d85-qp9vp              ●        2/2        Running                               11 10.244.2.184     aer
 management-portal-backend-7755db9fb8-bjxpv              ●        1/1        Running                                3 10.244.2.137     aer
 management-portal-frontend-7948848657-sccq2             ●        1/1        Running                                4 10.244.2.194     aer
 metallb-controller-568f4dc5c-wbzjl                      ●        1/1        Running                                2 10.244.2.201     aer
 metallb-speaker-8mwwr                                   ●                                                                           aer
```

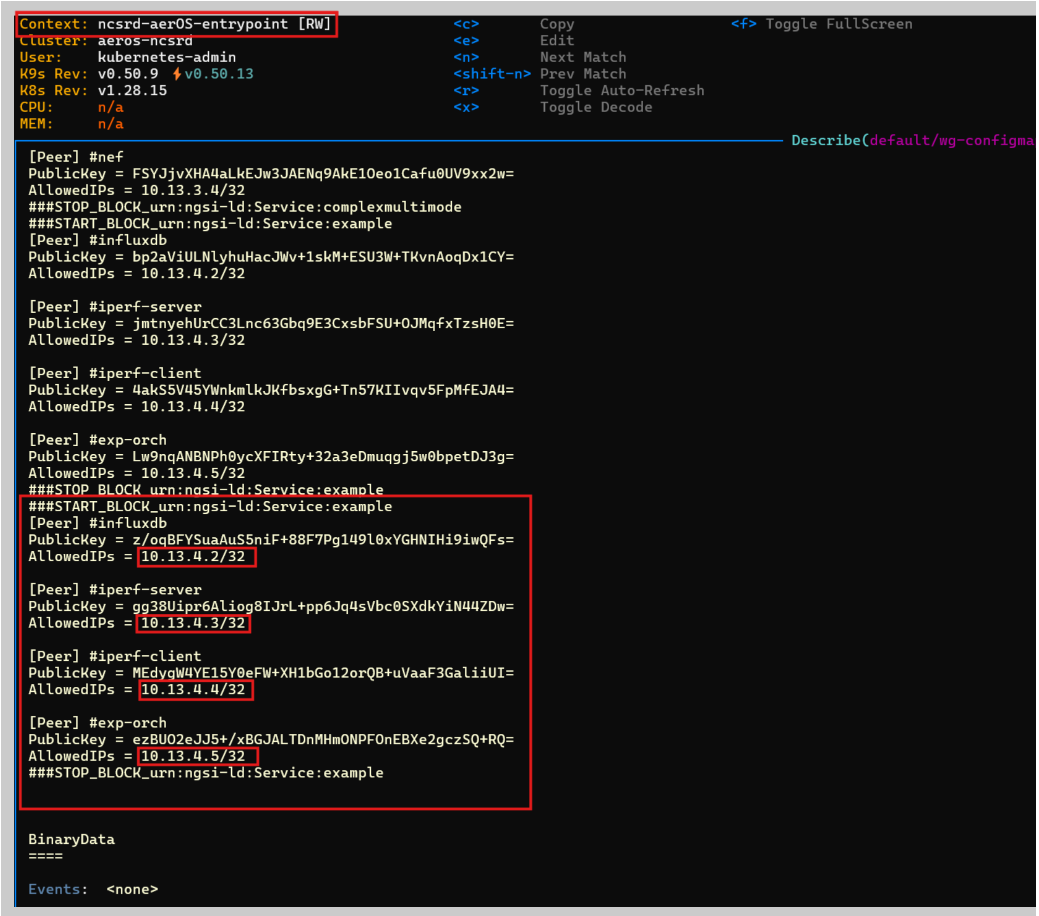


```
Context: cosmote-domain [RW]          <0> all      <a>        Attach       <ctrl-k>   Kill            <o> Show Node
Cluster: kubernetes                   <1> default  <ctrl-d>   Delete       <l>        Logs            <f> Show PortForward
User:    kubernetes-admin                          <d>        Describe     <p>        Logs Previous   <t> Transfer
K9s Rev: v0.50.6 ⚡v0.50.13                         <e>        Edit         <shift-f>  Port-Forward    <y> YAML
K8s Rev: v1.32.6                                   <?>        Help         <z>        Sanitize
CPU:     10%                                       <shift-j>  Jump Owner   <s>        Shell
MEM:     32%
                                                              pods(default)[27]
 NAME↑                                                    PF     READY    STATUS      RESTARTS   CPU    MEM   %CPU/R   %CPU/L   %MEM/R   %MEM/L
 aeros-k8s-shim-796976b94-ndffp                           ●      1/1      Running            0     2     75      n/a      n/a      n/a      n/a
 aeros-llo-api-544d484c7d-fx54x                           ●      1/1      Running            4     1     34       10        0       53       26
 aeros-service-example-component-exp-orch-58db4bf655-r6grc ●     2/2      Running            0     2     53      n/a      n/a      n/a      n/a
 aeros-service-example-component-iperf-client-7d4755cc94-xjwd4 ● 2/2      Running            0     1     24      n/a      n/a      n/a      n/a
 api-gateway-krakend-5bf49b8879-4tk88                     ●      1/1      Running            1     1     86      n/a      n/a      n/a      n/a
 chronograf-685dc85d64-t7h9n                              ●      1/1      Running           10     1     53      n/a      n/a      n/a      n/a
 federator-6c94d89d78-fqm6p                               ●      1/1      Running            7     0      5      n/a      n/a      n/a      n/a
 grafana-67c9d947f6-ftxlp                                 ●      1/1      Running            1     1    147      n/a      n/a      n/a      n/a
 hlo-allocator-99d578cdd-2c7w4                            ●      1/1      Running            2     3    734      n/a      n/a      n/a      n/a
 hlo-data-aggregator-6f9c95465f-jff62                     ●      1/1      Running            2     5     65      n/a      n/a      n/a      n/a
 hlo-deployment-engine-5779bd9c8d-g2n5b                   ●      1/1      Running            2     4     58      n/a      n/a      n/a      n/a
 hlo-frontend-75df96c577-lw74x                            ●      1/1      Running            2     2     56      n/a      n/a      n/a      n/a
 influxdb-98754d6f9-2xc2j                                 ●      1/1      Running            1     2    564      n/a      n/a      n/a      n/a
 kapacitor-cdd6d65f7-c9sl6                                ●      1/1      Running            3     1     34      n/a      n/a      n/a      n/a
```

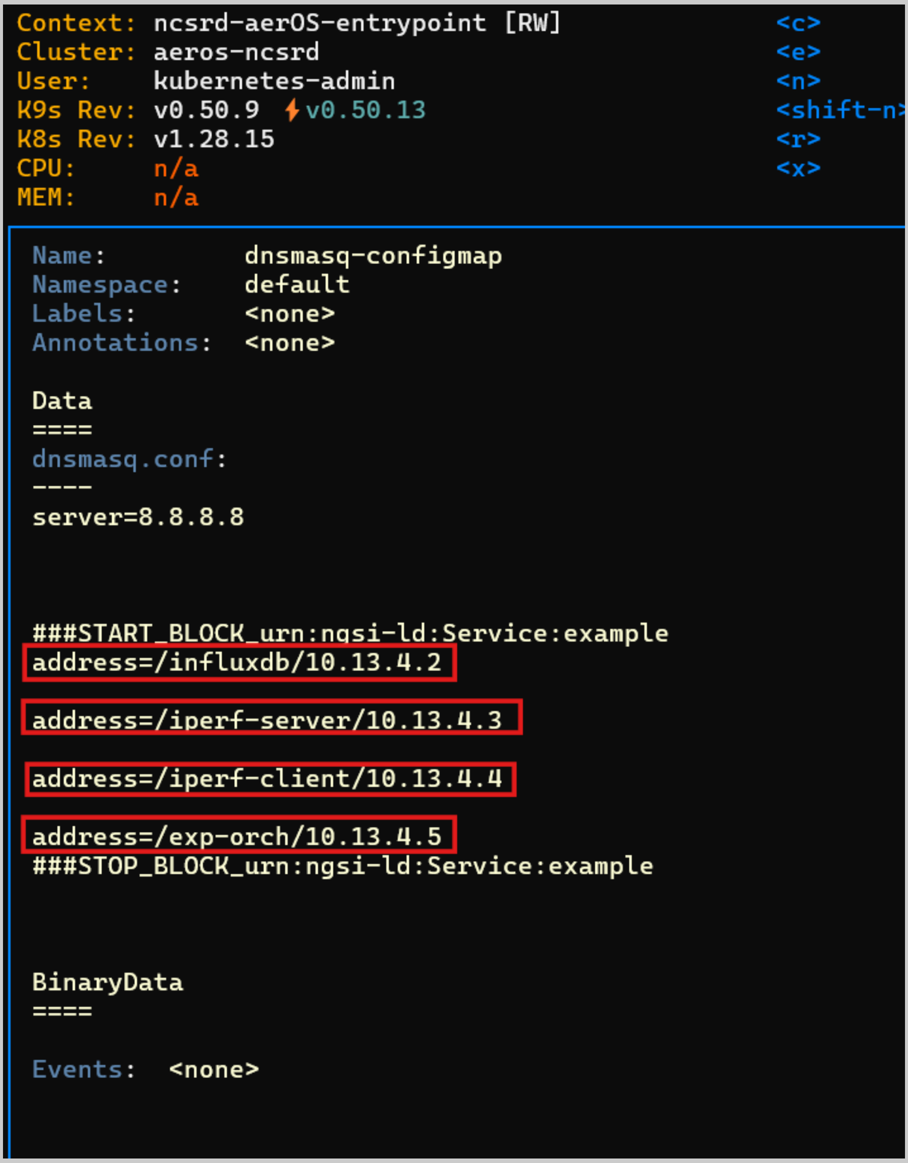
WireGuard overlay configuration and handshake logs confirm inter-domain tunnel establishment.



```
wireguard-server-5dbfdc64b4-td8lx:/# wg
interface: wg0
  public key: Tsvv2Qu65eeiZqoK3rlgDEVjngFrEUYclRppSJQVzRQ=
  private key: (hidden)
  listening port: 51820

peer: z/oqBFYSuaAuS5niF+88F7Pg149l0xYGHNIHi9iwQFs=
  endpoint: 10.220.0.1:38706
  allowed ips: 10.13.4.2/32
  latest handshake: 21 seconds ago
  transfer: 6.16 KiB received, 11.48 KiB sent

peer: MEdygW4YE15Y0eFW+XH1bGo12orQB+uVaaF3GaliiUI=
  endpoint: 10.220.0.1:43315
  allowed ips: 10.13.4.4/32
  latest handshake: 35 seconds ago
  transfer: 10.87 MiB received, 2.77 KiB sent

peer: gg38Uipr6Aliog8IJrL+pp6Jq4sVbc0SXdkYiN44ZDw=
  endpoint: 10.220.0.1:53903
  allowed ips: 10.13.4.3/32
  latest handshake: 35 seconds ago
  transfer: 5.15 KiB received, 10.86 MiB sent

peer: ezBUO2eJJ5+/xBGJALTDnMHmONPFOnEBXe2gczSQ+RQ=
  endpoint: 10.220.0.1:39520
  allowed ips: 10.13.4.5/32
  latest handshake: 35 seconds ago
  transfer: 17.82 KiB received, 12.65 KiB sent
```

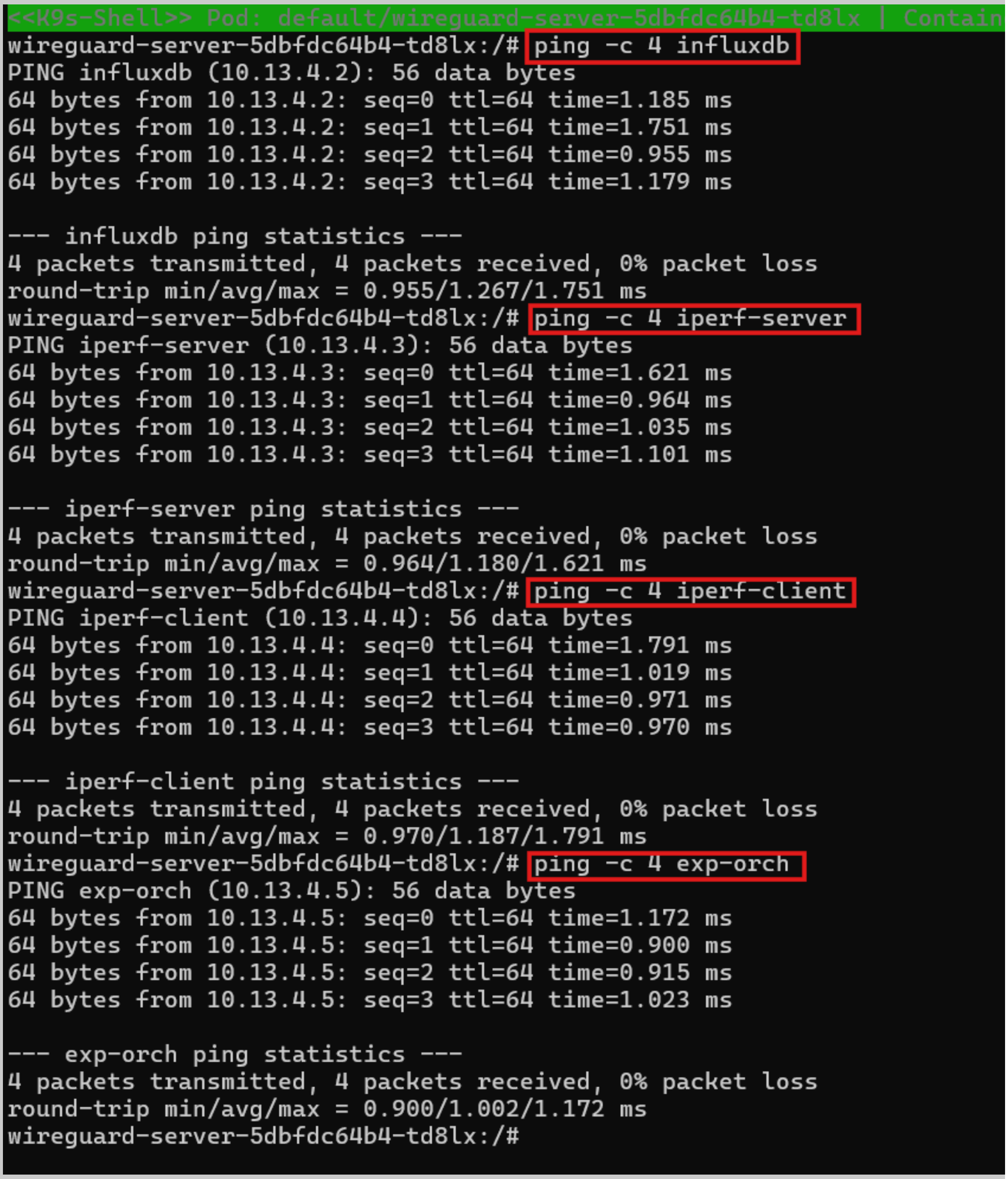


```
Context: ncsrd-aerOS-entrypoint [RW]              <c>        Copy                  <f> Toggle FullScreen
Cluster: aeros-ncsrd                              <e>        Edit
User:    kubernetes-admin                         <n>        Next Match
K9s Rev: v0.50.9  v0.50.13                        <shift-n>  Prev Match
K8s Rev: v1.28.15                                 <r>        Toggle Auto-Refresh
CPU:     n/a                                      <x>        Toggle Decode
MEM:     n/a
                                                                          Describe(default/wg-configma
 [Peer] #nef
 PublicKey = FSYJjvXHA4aLkEJw3JAENq9AkE1Oeo1Cafu0UV9xx2w=
 AllowedIPs = 10.13.3.4/32
 ###STOP_BLOCK_urn:ngsi-ld:Service:complexmultimode
 ###START_BLOCK_urn:ngsi-ld:Service:example
 [Peer] #influxdb
 PublicKey = bp2aViULNlyhuHacJWv+1skM+ESU3W+TKvnAoqDx1CY=
 AllowedIPs = 10.13.4.2/32

 [Peer] #iperf-server
 PublicKey = jmtnyehUrCC3Lnc63Gbq9E3CxsbFSU+OJMqfxTzsH0E=
 AllowedIPs = 10.13.4.3/32

 [Peer] #iperf-client
 PublicKey = 4akS5V45YWnkmlkJKfbsxgG+Tn57KIIvqv5FpMfEJA4=
 AllowedIPs = 10.13.4.4/32

 [Peer] #exp-orch
 PublicKey = Lw9nqANBNPh0ycXFIRty+32a3eDmuqgj5w0bpetDJ3g=
 AllowedIPs = 10.13.4.5/32
 ###STOP_BLOCK_urn:ngsi-ld:Service:example
 ###START_BLOCK_urn:ngsi-ld:Service:example
 [Peer] #influxdb
 PublicKey = z/oqBFYSuaAuS5niF+88F7Pg149l0xYGHNIHi9iwQFs=
 AllowedIPs = 10.13.4.2/32

 [Peer] #iperf-server
 PublicKey = gg38Uipr6Aliog8IJrL+pp6Jq4sVbc0SXdkYiN44ZDw=
 AllowedIPs = 10.13.4.3/32

 [Peer] #iperf-client
 PublicKey = MEdygW4YE15Y0eFW+XH1bGo12orQB+uVaaF3GaliiUI=
 AllowedIPs = 10.13.4.4/32

 [Peer] #exp-orch
 PublicKey = ezBUO2eJJ5+/xBGJALTDnMHmONPFOnEBXe2gczSQ+RQ=
 AllowedIPs = 10.13.4.5/32
 ###STOP_BLOCK_urn:ngsi-ld:Service:example


 BinaryData
 ====

 Events:  <none>
```

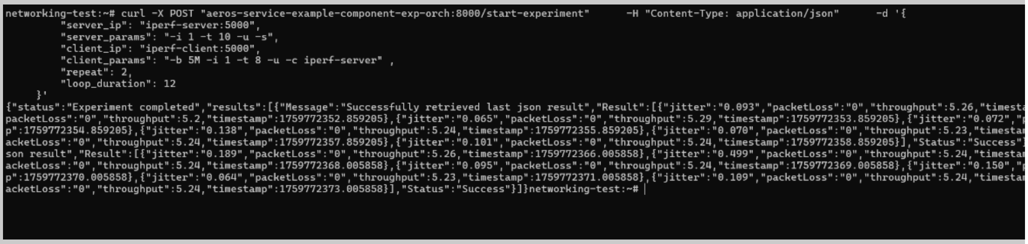


```
Context: ncsrd-aerOS-entrypoint [RW]                <c>
Cluster: aeros-ncsrd                                <e>
User:    kubernetes-admin                           <n>
K9s Rev: v0.50.9 ⚡v0.50.13                          <shift-n
K8s Rev: v1.28.15                                   <r>
CPU:     n/a                                        <x>
MEM:     n/a

 Name:         dnsmasq-configmap
 Namespace:    default
 Labels:       <none>
 Annotations:  <none>

 Data
 ====
 dnsmasq.conf:
 ----
 server=8.8.8.8



 ###START_BLOCK_urn:ngsi-ld:Service:example
 address=/influxdb/10.13.4.2

 address=/iperf-server/10.13.4.3

 address=/iperf-client/10.13.4.4

 address=/exp-orch/10.13.4.5
 ###STOP_BLOCK_urn:ngsi-ld:Service:example



 BinaryData
 ====

 Events:  <none>
```

Ping and API call results verifying inter-component connectivity via overlay hostnames.

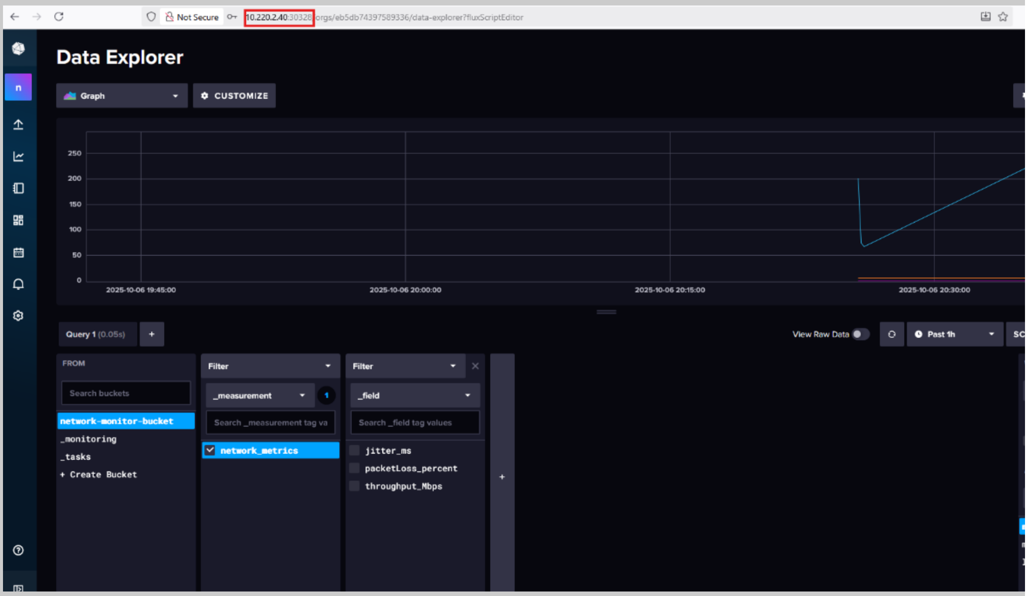


```
<<K9s-Shell>> Pod: default/wireguard-server-5dbfdc64b4-td8lx | Contain
wireguard-server-5dbfdc64b4-td8lx:/# ping -c 4 influxdb
PING influxdb (10.13.4.2): 56 data bytes
64 bytes from 10.13.4.2: seq=0 ttl=64 time=1.185 ms
64 bytes from 10.13.4.2: seq=1 ttl=64 time=1.751 ms
64 bytes from 10.13.4.2: seq=2 ttl=64 time=0.955 ms
64 bytes from 10.13.4.2: seq=3 ttl=64 time=1.179 ms

--- influxdb ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.955/1.267/1.751 ms
wireguard-server-5dbfdc64b4-td8lx:/# ping -c 4 iperf-server
PING iperf-server (10.13.4.3): 56 data bytes
64 bytes from 10.13.4.3: seq=0 ttl=64 time=1.621 ms
64 bytes from 10.13.4.3: seq=1 ttl=64 time=0.964 ms
64 bytes from 10.13.4.3: seq=2 ttl=64 time=1.035 ms
64 bytes from 10.13.4.3: seq=3 ttl=64 time=1.101 ms

--- iperf-server ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.964/1.180/1.621 ms
wireguard-server-5dbfdc64b4-td8lx:/# ping -c 4 iperf-client
PING iperf-client (10.13.4.4): 56 data bytes
64 bytes from 10.13.4.4: seq=0 ttl=64 time=1.791 ms
64 bytes from 10.13.4.4: seq=1 ttl=64 time=1.019 ms
64 bytes from 10.13.4.4: seq=2 ttl=64 time=0.971 ms
64 bytes from 10.13.4.4: seq=3 ttl=64 time=0.970 ms

--- iperf-client ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.970/1.187/1.791 ms
wireguard-server-5dbfdc64b4-td8lx:/# ping -c 4 exp-orch
PING exp-orch (10.13.4.5): 56 data bytes
64 bytes from 10.13.4.5: seq=0 ttl=64 time=1.172 ms
64 bytes from 10.13.4.5: seq=1 ttl=64 time=0.900 ms
64 bytes from 10.13.4.5: seq=2 ttl=64 time=0.915 ms
64 bytes from 10.13.4.5: seq=3 ttl=64 time=1.023 ms

--- exp-orch ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.900/1.002/1.172 ms
wireguard-server-5dbfdc64b4-td8lx:/#
```



```
networking-test:~# curl -X POST "aeros-service-example-component-exp-orch:8000/start-experiment"      -H "Content-Type: application/json"      -d '{
        "server_ip": "iperf-server:5000",
        "server_params": "-i 1 -t 10 -u -s",
        "client_ip": "iperf-client:5000",
        "client_params": "-b 5M -i 1 -t 8 -u -c iperf-server" ,
        "repeat": 2,
        "loop_duration": 12
    }'
{"status":"Experiment completed","results":[{"Message":"Successfully retrieved last json result","Result":[{"jitter":"0.093","packetLoss":"0","throughput":5.26,"timesta
packetLoss":"0","throughput":5.2,"timestamp":1759772352.859205},{"jitter":"0.065","packetLoss":"0","throughput":5.29,"timestamp":1759772353.859205},{"jitter":"0.072","p
p":1759772354.859205},{"jitter":"0.138","packetLoss":"0","throughput":5.24,"timestamp":1759772355.859205},{"jitter":"0.070","packetLoss":"0","throughput":5.23,"timestam
acketLoss":"0","throughput":5.24,"timestamp":1759772357.859205},{"jitter":"0.101","packetLoss":"0","throughput":5.24,"timestamp":1759772358.859205}],"Status":"Success"}
son result","Result":[{"jitter":"0.189","packetLoss":"0","throughput":5.26,"timestamp":1759772366.005858},{"jitter":"0.499","packetLoss":"0","throughput":5.24,"timestam
acketLoss":"0","throughput":5.24,"timestamp":1759772368.005858},{"jitter":"0.095","packetLoss":"0","throughput":5.24,"timestamp":1759772369.005858},{"jitter":"0.150","p
p":1759772370.005858},{"jitter":"0.064","packetLoss":"0","throughput":5.23,"timestamp":1759772371.005858},{"jitter":"0.109","packetLoss":"0","throughput":5.24,"timestam
acketLoss":"0","throughput":5.24,"timestamp":1759772373.005858}],"Status":"Success"}]}networking-test:~# |
```

InfluxDB visualization of iperf measurement results (throughput timeline)



KPI 1.3.5 Different types of networks managed by aerOS in pilot deployment

Table 43: KPI 1.3.5 Different types of networks managed by aerOS in pilot deployment

KPI ID number and partner resp.	**KPI 1.3.5**		
KPI Name	**Different types of networks employed by aerOS in pilot deployment**		
Description	This KPI refers to the number of heterogeneous networks that aerOS is homogenizing within the IoT-Edge-Cloud continuum, offering to the users a unified and homogeneous experienced, independently of the underlying network technology.		
Motivation	This KPI quantifies that level of heterogeneity that that Meta-OS homogenizes and unifies under the same continuum.		
Target value	At least 2 network accesses (e.g. 5G, LAN, WiFi, ZigBee).		
Prerequisites	Computing resources integrated as IEs.		
aerOS components (task)	CNI (T3.1)		
Evaluation means	IE network layer and components reporting interface type and connectivity media used. Screenshots from within IEs explicitly stating their connectivity type will be provided.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*N/A*	*2 (100%)*	*7 (>300%)*

Outcome elaboration (M38)	In the MVP phase, already reported in D5.5, the IEs deployed within the NCSRD, CF, and UPV aerOS domains were interconnected through LAN, WiFi, and 5G network media. Evidence of these connections was provided in the form of topological diagrams and interface screenshots, demonstrating the diversity of network types managed by aerOS even at the early stage. The current report expands this validation to include the pilot deployments. The following table summarizes the different types of networks integrated by aerOS across the pilots, ranging from traditional LAN and WiFi to advanced connectivity technologies such as 5G, 4G, RFID for asset tracking, and IoT-oriented protocols like Zigbee and Z-Wave.

Pilot	***Networks Managed***
Pilot 1.1	*LAN, WiFi*
Pilot 1.2	*LAN, WiFi, 5G*
Pilot 1.3	*LAN, WiFi*
Pilot 1.4	*LAN, WiFi, RFID*
Pilot 2	*LAN, WiFi*
Pilot 3	*LAN, WiFi, 5G, 4G*
Pilot 4	*LAN, 4G*
Pilot 5	*LAN, WiFi, Zigbee, Z-Wave*

As in the previous KPI, detailed evidence of the network topologies, deployment configurations, and site photographs has already been presented in *D5.4 – Use cases deployment and implementation (2)* and is referenced here as proof. The consolidated table illustrates that aerOS has successfully managed a wide spectrum of heterogeneous network types across all pilot sites, far exceeding the baseline demonstrated during the MVP. This confirms aerOS' ability to operate as a MetaOS for the continuum, capable of unifying and orchestrating connectivity across diverse network infrastructures.

The results confirm that aerOS supports not only common LAN and WiFi connectivity, but also integrates 4G/5G, RFID-based tracking, and IoT-specific technologies such as Zigbee and Z-Wave, demonstrating its versatility in heterogeneous network integration

aerOS cybersecurity and trust components

KPI 1.4.1 Delivery of dedicated aerOS components as Open-Source SW for cybersecurity, privacy and trust (KVI-3.1)

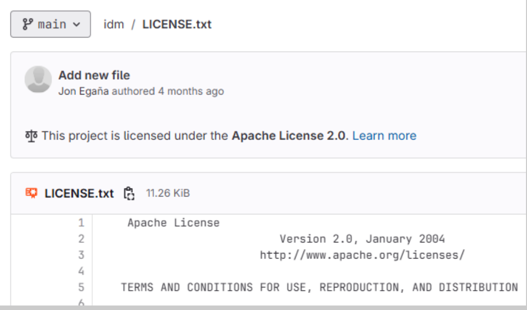
Table 44: KPI 1.4.1 Delivery of dedicated aerOS components as Open-Source SW for cybersecurity, privacy and trust (KVI-3.1)

KPI ID number and partner resp.	**KPI 1.4.1**
KPI Name	**Delivery of dedicated aerOS components as Open-Source SW for cybersecurity, privacy and trust**
Description	The process of making components of aerOS regarding cybersecurity, privacy and trust available to the public as open-source software.

<table>
<tr><td>Motivation</td><td colspan="3">By delivering dedicated aerOS components as open-source software focused on cybersecurity, privacy, and trust, the initiative likely aims to contribute to the broader tech community by providing robust tools for building more secure and trustworthy digital environments</td></tr>
<tr><td>Target value</td><td colspan="3"><i>100% OSS services</i></td></tr>
<tr><td>Prerequisites</td><td colspan="3">The integration of aerOS security, privacy and trust components in aerOS domain</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">aerOS IDM (T3.4), aerOS Secure API Gateway (T3.4), Self-security (T3.5), Trust monitoring component (T4.5)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">All the cybersecurity services will make use of OSS licensing schemes (e.g., Eclipse, GPL, etc.). Moreover, to further boost the use of these tools by the community, the readme files and the aerOS official documentation of these services will include a brief guide about how to contribute to following releases.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>3 (75%)</i></td><td><i>4(100 %)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">Note: the following links and screenshots point to a Gitlab repo that is not yet available to the community. The elements will be uploaded to the Github account of the project, or will be part of the Open Source product ECLIPSE aeriOS before the date of the final review.

aerOS IDM (T3.4): https://gitlab.aeros-project.eu/wp3/t3.4/idm

License: Apache license, version 2.0
License/Readme file: https://gitlab.aeros-project.eu/wp3/t3.4/idm/-/blob/main/LICENSE.txt?ref_type=heads



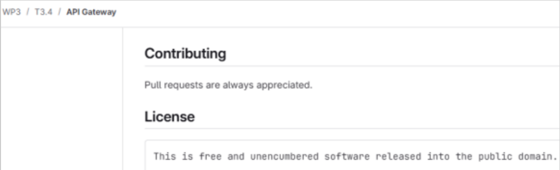
Jon Egaña authored 4 months ago
This project is licensed under the Apache License 2.0.

```
        Apache License
                                  Version 2.0, January 2004
                               http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
```

aerOS Secure API Gateway (T3.4): https://gitlab.aeros-project.eu/wp3/t3.4/api-gateway

License: Unlicensed license
License/Readme file: https://gitlab.aeros-project.eu/wp3/t3.4/api-gateway/-/blob/main/README.md?ref_type=heads

</td></tr>
</table>



WP3 / T3.4 / API Gateway

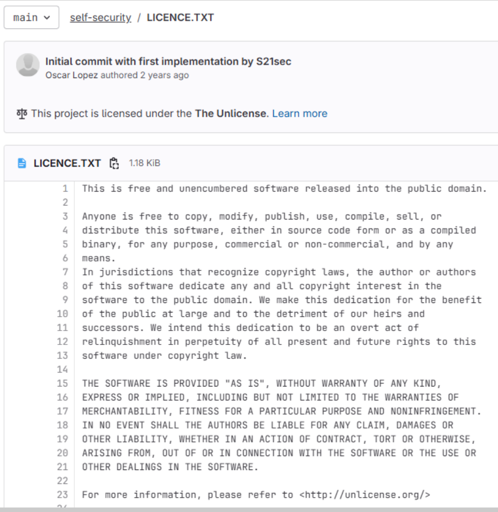
Contributing

Pull requests are always appreciated.

License

```
This is free and unencumbered software released into the public domain.
```

- aerOS Self-security (T3.5) https://gitlab.aeros-project.eu/wp3/t3.5/self-security/-/tree/main
    - License: Unlicensed license
    - License/Readme file: https://gitlab.aeros-project.eu/wp3/t3.5/self-security/-/blob/main/LICENCE.TXT



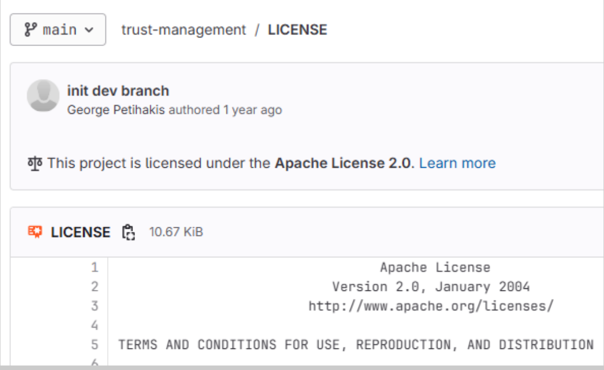
main / self-security / LICENCE.TXT

Initial commit with first implementation by S21sec
Oscar Lopez authored 2 years ago

This project is licensed under the The Unlicense. Learn more

LICENCE.TXT 1.18 KiB

```
This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or
distribute this software, either in source code form or as a compiled
binary, for any purpose, commercial or non-commercial, and by any
means.
In jurisdictions that recognize copyright laws, the author or authors
of this software dedicate any and all copyright interest in the
software to the public domain. We make this dedication for the benefit
of the public at large and to the detriment of our heirs and
successors. We intend this dedication to be an overt act of
relinquishment in perpetuity of all present and future rights to this
software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org/>
```

- aerOS Trust Monitoring component (T4.5) https://gitlab.aeros-project.eu/wp4/t4.5/trust-management
    - License: Apache license, version 2.0
    - License/Readme file: https://gitlab.aeros-project.eu/wp4/t4.5/trust-management/-/blob/main/LICENSE?ref_type=heads
    - 

main / trust-management / LICENSE

init dev branch
George Petihakis authored 1 year ago

This project is licensed under the Apache License 2.0. Learn more

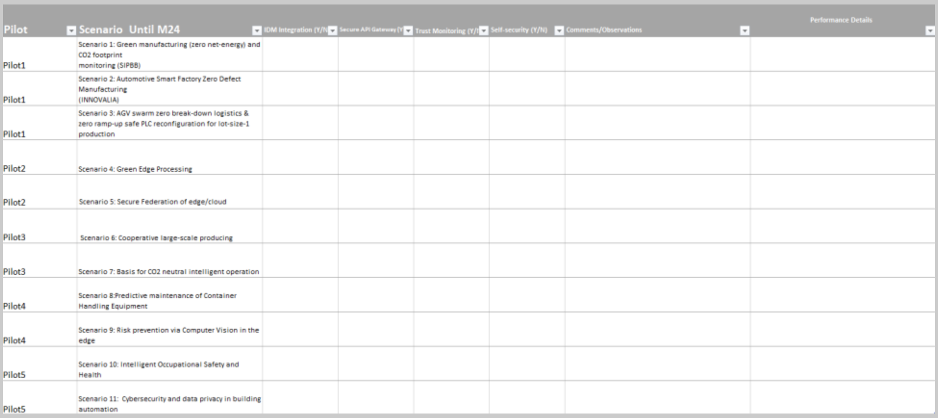
LICENSE 10.67 KiB

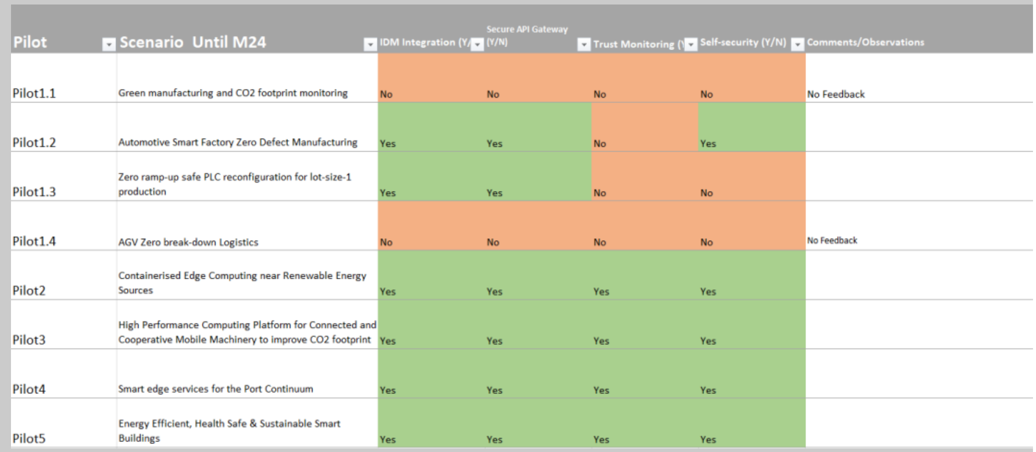
```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
```

KPI 1.4.2 # scenarios with security, privacy and trust by design deployed (KVI-3.2)

Table 45: KPI 1.4.2 # scenarios with security, privacy and trust by design deployed (KVI-3.2)

KPI ID number and partner resp.	**KPI 1.4.2**

<table>
<tr><th>KPI Name</th><th colspan="3"># scenarios with security, privacy and trust by design deployed (KVI-3.2)</th></tr>
<tr><td>Description</td><td colspan="3">Quantifies the number of scenarios where principles of security, privacy, and trust have been integrated by design.</td></tr>
<tr><td>Motivation</td><td colspan="3">This KPI tracks the implementation of these foundational principles from the earliest stages of development, ensuring that each deployment is inherently secure, respects user privacy, and is trustworthy</td></tr>
<tr><td>Target value</td><td colspan="3">>50% scenarios</td></tr>
<tr><td>Prerequisites</td><td colspan="3">The integration of aerOS security, privacy and trust components in aerOS domain</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">aerOS IDM (T3.4), aerOS Secure API Gateway (T3.4), Self-security (T3.5), Trust monitoring component (T4.5)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">A checklist or matrix has been developed to monitor the integration of each security, privacy, and trust component across various scenarios.

<table>
<tr><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th>Performance Details</th></tr>
<tr><th>Pilot</th><th>Scenario Until M24</th><th>IDM Integration (Y/N</th><th>Secure API Gateway (Y</th><th>Trust Monitoring (Y/</th><th>Self-security (Y/N)</th><th>Comments/Observations</th><th></th></tr>
<tr><td>Pilot1</td><td>Scenario 1: Green manufacturing (zero net-energy) and CO2 footprint monitoring (SIPBB)</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Pilot1</td><td>Scenario 2: Automotive Smart Factory Zero Defect Manufacturing (INNOVALIA)</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Pilot1</td><td>Scenario 3: AGV swarm zero break-down logistics & zero ramp-up safe PLC reconfiguration for lot-size-1 production</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Pilot2</td><td>Scenario 4: Green Edge Processing</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Pilot2</td><td>Scenario 5: Secure Federation of edge/cloud</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Pilot3</td><td>Scenario 6: Cooperative large-scale producing</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Pilot3</td><td>Scenario 7: Basis for CO2 neutral intelligent operation</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Pilot4</td><td>Scenario 8:Predictive maintenance of Container Handling Equipment</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Pilot4</td><td>Scenario 9: Risk prevention via Computer Vision in the edge</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Pilot5</td><td>Scenario 10: Intelligent Occupational Safety and Health</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>Pilot5</td><td>Scenario 11: Cybersecurity and data privacy in building automation</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>0 scenarios (0%)</i></td><td><i>4 / 8 pilot scenarios (50%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">At M38, four out of eight pilot scenarios (50%) have integrated the full set of aerOS security, privacy, and trust components (IDM, Secure API Gateway, Trust Monitoring, and Self-security). For pilots 1.1 and 1.4, we did not receive any feedback to confirm whether the target value of >50% scenarios has been reached. The integrated scenarios cover a wide range of domains including manufacturing, ports, energy, occupational safety, and building automation, confirming that security and trust-by-design principles are being validated in diverse real-world conditions. The remaining five scenarios are either in partial deployment or awaiting final partner contributions, with their absence reflecting deployment timelines rather than technical limitations.</td></tr>
</table>

Pilot	Scenario Until M24	IDM Integration (Y/	Secure API Gateway (Y/N)	Trust Monitoring (Y	Self-security (Y/N)	Comments/Observations
Pilot1.1	Green manufacturing and CO2 footprint monitoring	No	No	No	No	No Feedback
Pilot1.2	Automotive Smart Factory Zero Defect Manufacturing	Yes	Yes	No	Yes	
Pilot1.3	Zero ramp-up safe PLC reconfiguration for lot-size-1 production	Yes	Yes	No	No	
Pilot1.4	AGV Zero break-down Logistics	No	No	No	No	No Feedback
Pilot2	Containerised Edge Computing near Renewable Energy Sources	Yes	Yes	Yes	Yes	
Pilot3	High Performance Computing Platform for Connected and Cooperative Mobile Machinery to improve CO2 footprint	Yes	Yes	Yes	Yes	
Pilot4	Smart edge services for the Port Continuum	Yes	Yes	Yes	Yes	
Pilot5	Energy Efficient, Health Safe & Sustainable Smart Buildings	Yes	Yes	Yes	Yes	

KPI 1.4.3 Delivery of a DevPrivSecOps cookbook and good practices manual (KVI-3.3)

Table 46: KPI 1.4.3 Delivery of a DevPrivSecOps cookbook and good practices manual (KVI-3.3)

KPI ID number and partner resp.	**KPI 1.4.3**		
KPI Name	**Delivery of a DevPrivSecOps cookbook and good practices manual (KVI-3.3)**		
Description	Cookbook to guide aerOS developers to implement the DevPrivSecOps methodology		
Motivation	The DevPrivSecOps methodology designed in aerOS allows to guide the project developers (and the developer community) to develop secure and privacy aware code by design.		
Target value	The project expects to produce 3 cookbooks in different formats: a DevPrivSecOps methodology definition report, a methodology implementation manual and an interactive GitLab guide with an example implementation.		
Prerequisites	The only prerequisite is to have a code repository in the project's GitLab.		
aerOS components (task)	This cookbook is used to help aerOS tool owners implement the DevPrivSecOps methodology designed and presented in D2.4 and D2.5.		
Evaluation means	This KPI will be measured by the number of cookbooks that have been distributed to the consortium for this purpose.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M36 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***2 (66%)***	***3 (100%)***
Outcome elaboration (M38)	With the completion of the task T2.4, as the end of this task, the deliverable D2.5 has been generated the implementation guides of the DevPrivSecOps methodology, and the configuration and use guides of the tools selected to implement this methodology		

	have been added. Additionally 3 different cookbooks were generated in order to guide in the implementation of the methodology: A cookbook document has been generated and also a Read The Docs page has been generated with the implementation guidelines (https://docs.aeros-project.eu/en/latest/methodology/index.html). The aerOS DevPrivSecOps cookbook document has been generated and it has been published in the project web page (https://aeros-project.eu/wp-content/uploads/2024/07/aerOS_DevPrivSecOps_CB.pdf). Finally, a video with an example of the implementation of the methodology has been generated and uploaded to the aerOS Youtube channel (https://youtu.be/38o_GrY8w_E?si=4106PNXORS1ZbNpH).

KPI 1.4.4 % of users/device/services properly authenticated

Table 47: KPI 1.4.4 % of users/device/services properly authenticated

KPI ID number and partner resp.	**KPI 1.4.4**		
KPI Name	**% of users/device/services properly authenticated**		
Description	The percentage of users/device/services properly authenticated through the aerOS identity management (IdM) service.		
Motivation	Monitoring properly authenticated users’ device/services (provided they have submitted the correct credentials) allows to verify the correct functioning of the aerOS IdM.		
Target value	> 95%		
Prerequisites	The prerequisite for this KPI is to have Keycloak and OpenLDAP installed, configured and federated to share data between them.		
aerOS components (task)	Keycloak and OpenLDAP (T3.4)		
Evaluation means	To evaluate this KPI, the Keycloak database will be accessed where the record of authentication attempts and which of these attempts have been successful is stored.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*N/A*	*30/31 –> 96,77% (100%)*	*97,77% (accomplished)*
Outcome elaboration (M38)	Target value achieved >95% in the aerOS pilots The percentage of users/device/services properly authenticated through the aerOS identity management (IdM) service.		

The following tables describes the outcome. These tables are the result of executing a custom developed script over the aerOS continuum in all the pilots, to check the security and authentication setup.

Pilot 1.1	97%
Pilot 1.2	98,60%
Pilot 1.3	95,20%
Pilot 1.4	96,40%
Pilot 2	99,99%
Pilot 3	100%
Pilot 4	95%
Pilot 5	100%

There was a requirement to enable login on IdM service to capture the data. Also, there was provided enough info to gather data from logs using a python script.

The following script enhances and facilitates the process of capturing the values for the KPI 1.4.4 The main purpose of the script is:

- Connects to a PostgreSQL database using the provided host, port, database name, user, and password.
- Executes an SQL query to retrieve realm and user data from the user_entity and realm tables.
- Saves the query results into a CSV file named realm_users.csv with headers included.
- Executes another SQL query to retrieve all data from the event_entity table.
- Exports the event query results into a CSV file named event_entities.csv.

kpi_1.4.4_data_generator.py

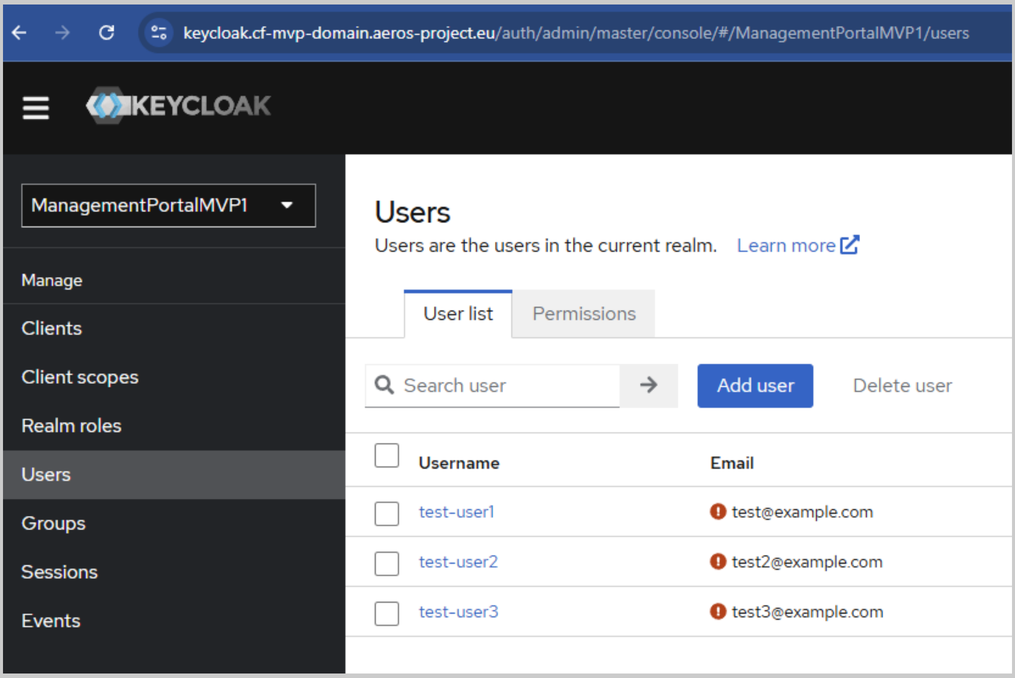
```
import psycopg2
import csv

# Configuración de la base de datos
DB_HOST = 'XXX.XXX.XXX.XXX'         # Ejemplo: 'localhost' o '192.168.1.100'
DB_PORT = '5432'             # Puerto por defecto de PostgreSQL
DB_NAME = 'keycloak'
DB_SCHEMA = 'public'      # Esquema en tu base de datos
DB_USER = 'keycloak'
DB_PASSWORD = 'password'

# Consulta SQL que deseas ejecutar
QUERY_USERS = '''
select r.id as realm_id, r.name as realm_name, ue.* from realm r, user_entity ue where r.id = ue.realm_id;
'''.format(schema=DB_SCHEMA)

QUERY_EVENTS = '''
select * from event_entity;
'''.format(schema=DB_SCHEMA)

# Archivo CSV de salida
OUTPUT_FILE_USERS = 'realm_users.csv'
OUTPUT_FILE_EVENTS = 'event_entities.csv'

def main():
    try:
        # Conectar a la base de datos
        conn = psycopg2.connect(
            host=DB_HOST,
```

```
            port=DB_PORT,
            dbname=DB_NAME,
            user=DB_USER,
            password=DB_PASSWORD
        )
        cursor = conn.cursor()

        # Ejecutar la consulta
        cursor.execute(QUERY_USERS)

        # Obtener los nombres de las columnas
        column_names = [desc[0] for desc in cursor.description]

        # Obtener todos los resultados
        rows = cursor.fetchall()

        # Escribir resultados en un archivo CSV
        with open(OUTPUT_FILE_USERS, 'w', newline='', encoding='utf-8') as csvfile:
            writer = csv.writer(csvfile)
            # Escribir encabezados
            writer.writerow(column_names)
            # Escribir filas
            writer.writerows(rows)

        print(f"Realm users data successfully exported to {OUTPUT_FILE_USERS}")

        # Ejecutar la consulta
        cursor.execute(QUERY_EVENTS)

        # Obtener los nombres de las columnas
        column_names = [desc[0] for desc in cursor.description]

        # Obtener todos los resultados
        rows = cursor.fetchall()

        # Escribir resultados en un archivo CSV
        with open(OUTPUT_FILE_EVENTS, 'w', newline='', encoding='utf-8') as csvfile:
            writer = csv.writer(csvfile)
            # Escribir encabezados
            writer.writerow(column_names)
            # Escribir filas
            writer.writerows(rows)

        print(f"Realm users data successfully exported to {OUTPUT_FILE_EVENTS}")


    except Exception as e:
        print(f"Ha ocurrido un error: {e}")
    finally:
        # Cerrar la conexión
        if cursor:
            cursor.close()
        if conn:
            conn.close()

if __name__ == '__main__':
    main()
```

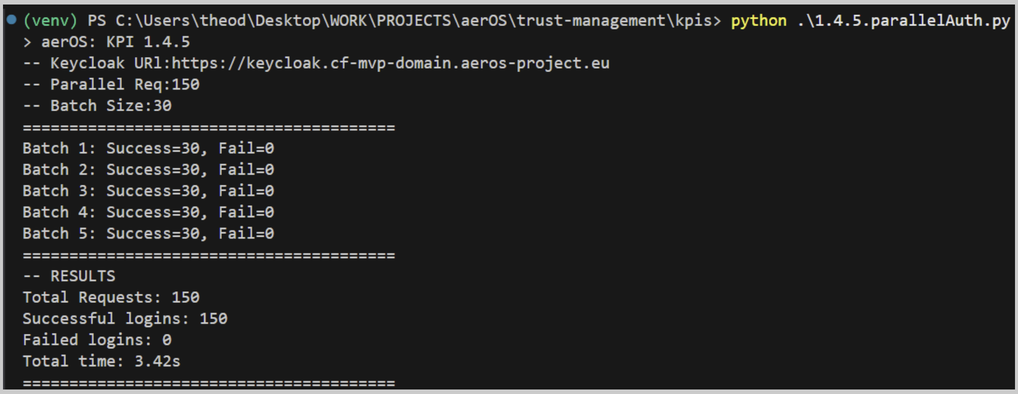
KPI 1.4.5 # of parallel successfully authenticated user/devices/services

Table 48: KPI 1.4.5 # of parallel successfully authenticated user/devices/services

KPI ID number and partner resp.	**KPI 1.4.5**
KPI Name	**# of parallel successfully authenticated user/devices/services**

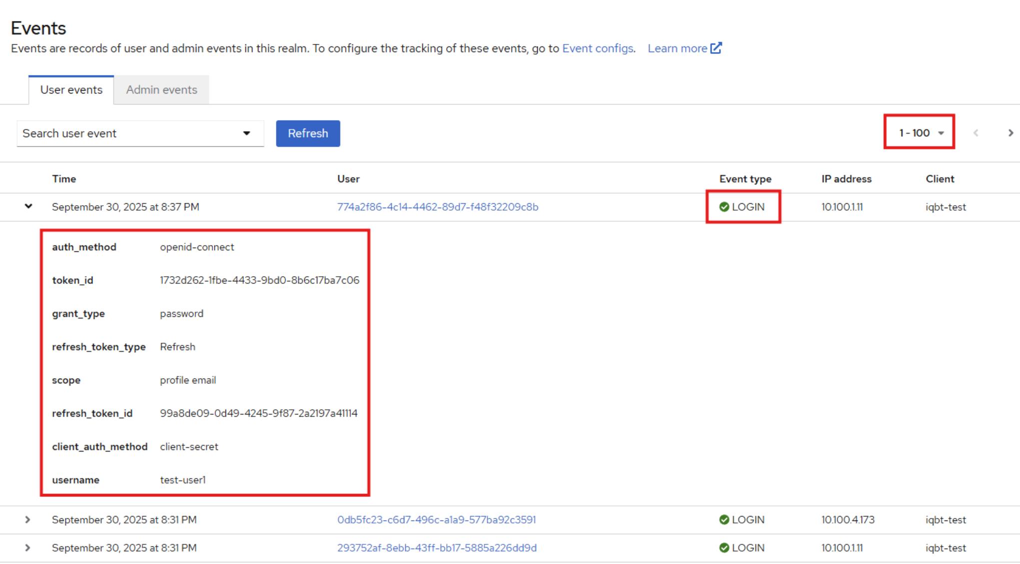
<table>
<tr><td>Description</td><td colspan="3">This KPI offers insights on the simultaneous load that the authentication system can proficiently manage</td></tr>
<tr><td>Motivation</td><td colspan="3">By monitoring this KPI, potential bottlenecks are identified, and informed decisions about necessary upgrades or optimizations to accommodate growing demand are made. It also helps in stressing testing and capacity planning, ensuring that the aerOS remains responsive and secure even as the number of simultaneous authentication requests increases.</td></tr>
<tr><td>Target value</td><td colspan="3">>150</td></tr>
<tr><td>Prerequisites</td><td colspan="3">The prerequisite for this KPI is to have Keycloak and OpenLDAP installed, configured and federated to share data between them.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">Keycloak and OpenLDAP (T3.4)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">To evaluate this KPI, the Keycloak database where the record of authentication attempts and which of these attempts have been successful stored will be accessed and analysed.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>151/155 (94.2%)</i></td><td><i>150/150 (100%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">For the M38 evaluation, we extended our methodology by creating three dedicated test users in the realm. This allowed us to balance the load across multiple accounts and better simulate realistic usage, rather than reusing a single account.


☐ Username | Email
☐ test-user1 | test@example.com
☐ test-user2 | test2@example.com
☐ test-user3 | test3@example.com</td></tr>
</table>



```
● (venv) PS C:\Users\theod\Desktop\WORK\PROJECTS\aerOS\trust-management\kpis> python .\1.4.5.parallelAuth.py
> aerOS: KPI 1.4.5
-- Keycloak URl:https://keycloak.cf-mvp-domain.aeros-project.eu
-- Parallel Req:150
-- Batch Size:30
========================================
Batch 1: Success=30, Fail=0
Batch 2: Success=30, Fail=0
Batch 3: Success=30, Fail=0
Batch 4: Success=30, Fail=0
Batch 5: Success=30, Fail=0
========================================
-- RESULTS
Total Requests: 150
Successful logins: 150
Failed logins: 0
Total time: 3.42s
========================================
```

The Python script (developed by IQB) issued parallel login requests against the Keycloak token endpoint, distributing 150 attempts equally across the three accounts (50 each). This approach not only verified the capacity of Keycloak to handle parallel user sessions but also provided response latency measurements and failure ratios. The results confirmed that the authentication infrastructure can scale to the defined KPI target in a balanced multi-user scenario.



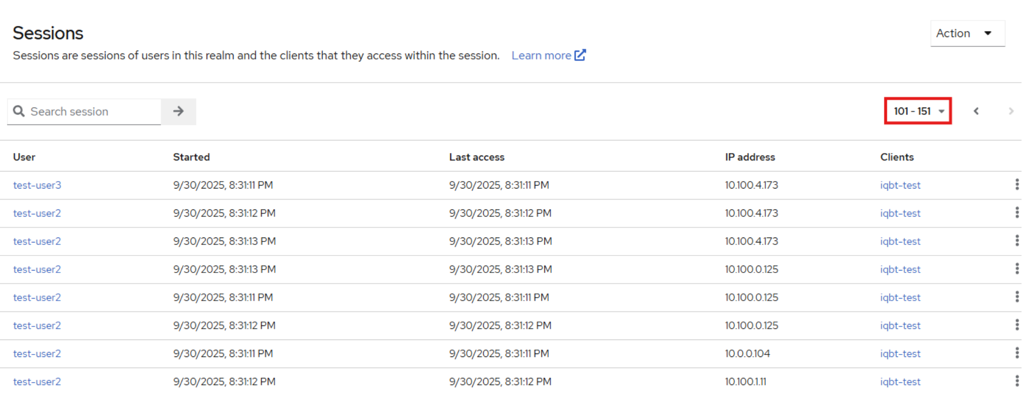
Events

Events are records of user and admin events in this realm. To configure the tracking of these events, go to Event configs. Learn more

	Time	User	Event type	IP address	Client
⌄	September 30, 2025 at 8:37 PM	774a2f86-4c14-4462-89d7-f48f32209c8b	✓ LOGIN	10.100.1.11	iqbt-test

auth_method	openid-connect
token_id	1732d262-1fbe-4433-9bd0-8b6c17ba7c06
grant_type	password
refresh_token_type	Refresh
scope	profile email
refresh_token_id	99a8de09-0d49-4245-9f87-2a2197a41114
client_auth_method	client-secret
username	test-user1

	Time	User	Event type	IP address	Client
›	September 30, 2025 at 8:31 PM	0db5fc23-c6d7-496c-a1a9-577ba92c3591	✓ LOGIN	10.100.4.173	iqbt-test
›	September 30, 2025 at 8:31 PM	293752af-8ebb-43ff-bb17-5885a226dd9d	✓ LOGIN	10.100.1.11	iqbt-test



Sessions

Sessions are sessions of users in this realm and the clients that they access within the session. Learn more

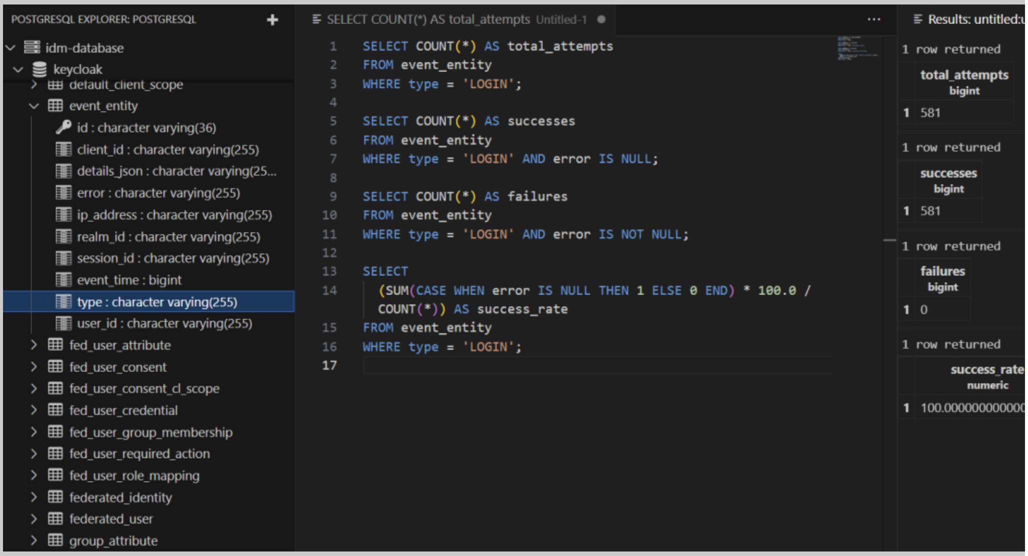
User	Started	Last access	IP address	Clients
test-user3	9/30/2025, 8:31:11 PM	9/30/2025, 8:31:11 PM	10.100.4.173	iqbt-test
test-user2	9/30/2025, 8:31:12 PM	9/30/2025, 8:31:12 PM	10.100.4.173	iqbt-test
test-user2	9/30/2025, 8:31:13 PM	9/30/2025, 8:31:13 PM	10.100.4.173	iqbt-test
test-user2	9/30/2025, 8:31:13 PM	9/30/2025, 8:31:13 PM	10.100.0.125	iqbt-test
test-user2	9/30/2025, 8:31:11 PM	9/30/2025, 8:31:11 PM	10.100.0.125	iqbt-test
test-user2	9/30/2025, 8:31:12 PM	9/30/2025, 8:31:12 PM	10.100.0.125	iqbt-test
test-user2	9/30/2025, 8:31:11 PM	9/30/2025, 8:31:11 PM	10.0.0.104	iqbt-test
test-user2	9/30/2025, 8:31:12 PM	9/30/2025, 8:31:12 PM	10.100.1.11	iqbt-test

The event and session records of Keycloak confirm that the parallel authentications were successfully established across the three test users. This validates that Keycloak is capable of sustaining 150+ concurrent authenticated sessions without service disruption or significant failure rates. The results demonstrate compliance with the KPI's target, providing confidence that the aerOS trust management layer can support large-scale authentication scenarios in pilot deployments.

KPI 1.4.6 % of users/device/services properly authorized

Table 49: KPI 1.4.6 % of users/device/services properly authorized

KPI ID number and partner resp.	**KPI 1.4.6**		
KPI Name	**% of users/device/services properly authorized**		
Description	Percentage of users/devices/services successfully authorised through the aerOS Identity Management (IdM) service.		
Motivation	Monitoring of properly authorised user devices/services (provided they have the permissions to access the target service/data) allows to verify the correct functioning of the authorisation component of the aerOS IdM.		
Target value	*>95%*		
Prerequisites	The prerequisite for this KPI is to have Keycloak and OpenLDAP installed, configured and federated to share data between them. Connection of Keycloak with the Management Portal for the users' authorization, and with KrakenD for the API access authorization.		
aerOS components (task)	Keycloak, OpenLDAP and KrakenD (T3.4), and Management Portal (T3.6)		
Evaluation means	To evaluate this KPI, the Keycloak database will be accessed where the record of authentication attempts and which of these attempts have been successful is stored.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***100%***	***100%***
Outcome elaboration (M38)	For the M38 evaluation, authentication attempts were analyzed directly from the Keycloak database using the **event_entity** table. The results showed that 100% of the login attempts were successful, thereby exceeding the KPI target of >95% successful authentications. This validates the robustness of the aerOS IdM service in managing user authentications under concurrent load.		

```sql
SELECT COUNT(*) AS total_attempts
FROM event_entity
WHERE type = 'LOGIN';

SELECT COUNT(*) AS successes
FROM event_entity
WHERE type = 'LOGIN' AND error IS NULL;

SELECT COUNT(*) AS failures
FROM event_entity
WHERE type = 'LOGIN' AND error IS NOT NULL;

SELECT
  (SUM(CASE WHEN error IS NULL THEN 1 ELSE 0 END) * 100.0 /
  COUNT(*)) AS success_rate
FROM event_entity
WHERE type = 'LOGIN';
```

```
1 row returned
   total_attempts
   bigint
1  581

1 row returned
   successes
   bigint
1  581

1 row returned
   failures
   bigint
1  0

1 row returned
   success_rate
   numeric
1  100.00000000000
```

KPI 1.4.7 # of petitions handled by the API Gateway per second

Table 50: KPI 1.4.7 # of petitions handled by the API Gateway per second

KPI ID number and partner resp.	**KPI 1.4.7**		
KPI Name	**# of petitions handled by the API Gateway per second**		
Description	The total number of petitions that the API Gateway is capable of handling per second without getting overloaded.		
Motivation	It is important to set a minimum number of petitions that the API Gateway must be able to fulfil without overloading to handle the average operations of the aerOS platforms.		
Target value	15 petitions per second		
Prerequisites	*The aerOS* domain deployed and ready with KrakenD deployed as the sole entrypoint into the domain (Kubernetes Ingress, etc). The backends also must be deployed and ready to receive traffic from KrakenD.		
aerOS components (task)	*KrakenD API Gateway (T3.4), Keycloak IAM (T3.4), OpenLDAP (T3.4),* Orion-LD (T4.2 and T4.6). All the necessary components to validate and authenticate a user as well as the backend to send the petitions to.		
Evaluation means	Using open-source benchmarking tools such as "autocannon," an extremely high number of requests per second will be sent to the API Gateway to test its ability to withstand the load.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**

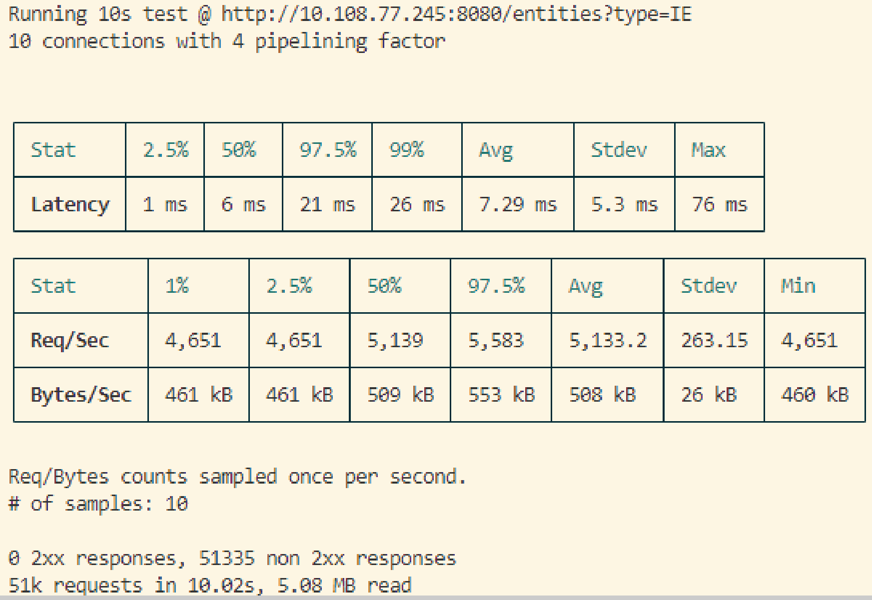
Measured value (% achieved)	***10 petitions per second***	***Average of 156Kb/s – see the different scenarios analysed)***	**14 to 15 petitions per second at minimum**

Outcome elaboration (M38)

KrakenD deployed in all aerOS domains alongside all other T3.4 tools, token caching done to reduce latency and the number of times a petition must call the backend. Improvements over the initial performance allowed for the latency to be greatly reduced, as the tests will show below.

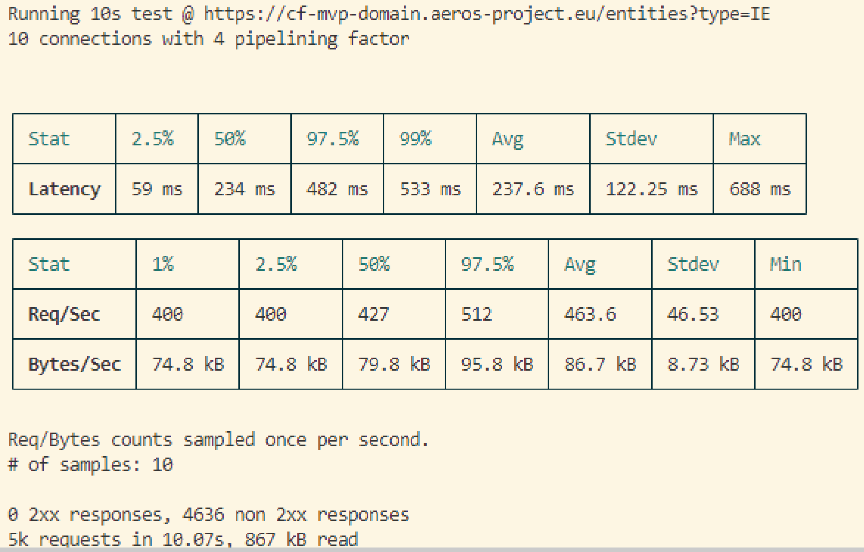
Two tests were performed on the UPV testing domain, the CloudFerro testing environment and the NCSRD pilot environment, i) one where the user sends a petition with an invalid user token, ii) another where the token is valid.

The first test showed these results on the UPV domain:



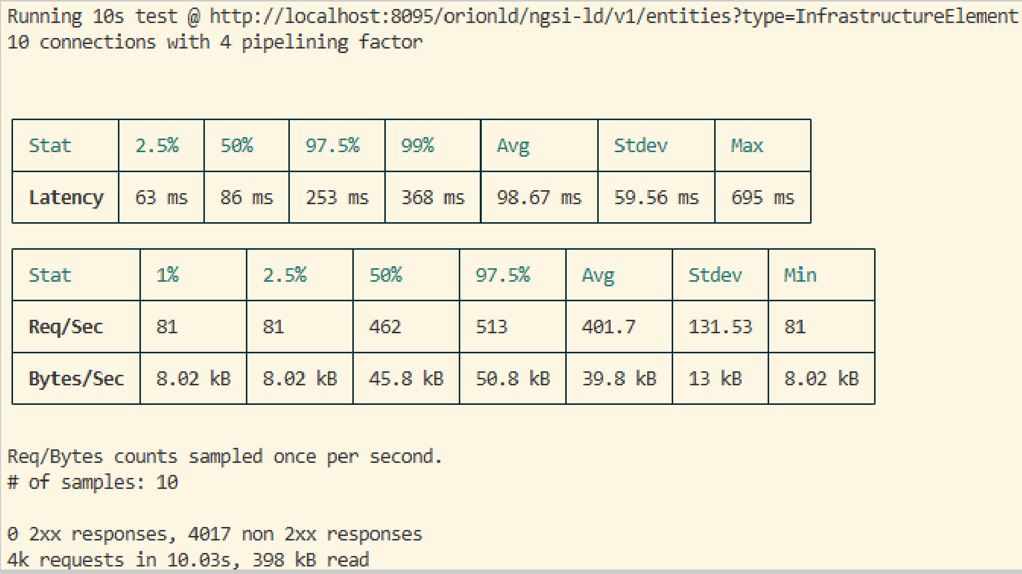
```
Running 10s test @ http://10.108.77.245:8080/entities?type=IE
10 connections with 4 pipelining factor


| Stat    | 2.5% | 50%  | 97.5% | 99%   | Avg     | Stdev  | Max   |
| Latency | 1 ms | 6 ms | 21 ms | 26 ms | 7.29 ms | 5.3 ms | 76 ms |

| Stat      | 1%     | 2.5%   | 50%    | 97.5%  | Avg     | Stdev  | Min    |
| Req/Sec   | 4,651  | 4,651  | 5,139  | 5,583  | 5,133.2 | 263.15 | 4,651  |
| Bytes/Sec | 461 kB | 461 kB | 509 kB | 553 kB | 508 kB  | 26 kB  | 460 kB |

Req/Bytes counts sampled once per second.
# of samples: 10

0 2xx responses, 51335 non 2xx responses
51k requests in 10.02s, 5.08 MB read
```

These are the results on the CloudFerro domain:



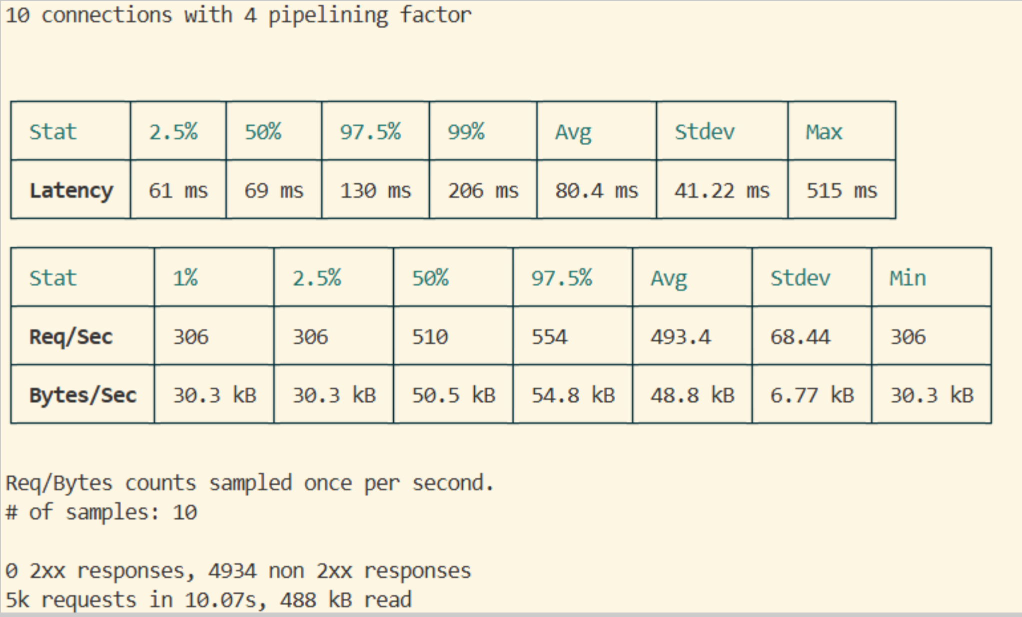
```
Running 10s test @ https://cf-mvp-domain.aeros-project.eu/entities?type=IE
10 connections with 4 pipelining factor


| Stat    | 2.5%  | 50%    | 97.5%  | 99%    | Avg      | Stdev     | Max    |
| Latency | 59 ms | 234 ms | 482 ms | 533 ms | 237.6 ms | 122.25 ms | 688 ms |

| Stat      | 1%      | 2.5%    | 50%     | 97.5%   | Avg     | Stdev   | Min     |
| Req/Sec   | 400     | 400     | 427     | 512     | 463.6   | 46.53   | 400     |
| Bytes/Sec | 74.8 kB | 74.8 kB | 79.8 kB | 95.8 kB | 86.7 kB | 8.73 kB | 74.8 kB |

Req/Bytes counts sampled once per second.
# of samples: 10

0 2xx responses, 4636 non 2xx responses
5k requests in 10.07s, 867 kB read
```

These are the results on the NCSRD domain:



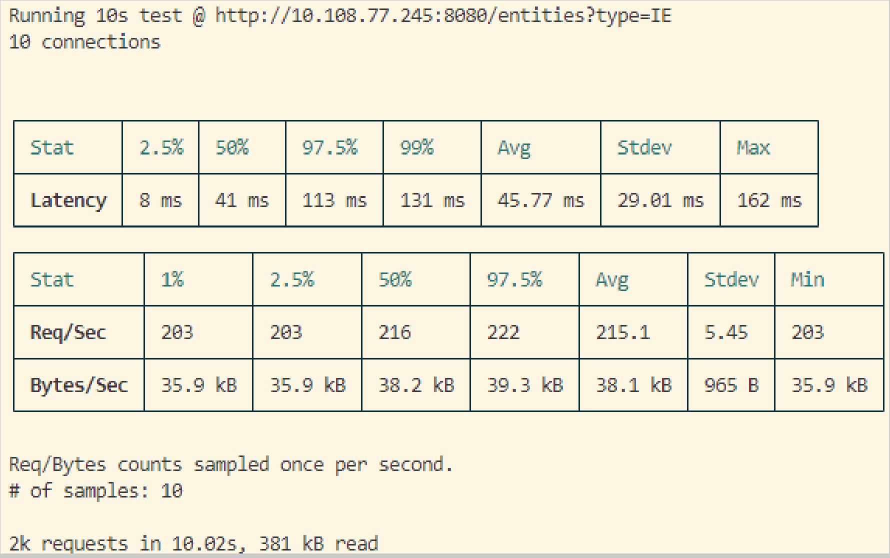
```
Running 10s test @ http://localhost:8095/orionld/ngsi-ld/v1/entities?type=InfrastructureElement
10 connections with 4 pipelining factor


| Stat    | 2.5%  | 50%   | 97.5%  | 99%    | Avg      | Stdev    | Max    |
| Latency | 63 ms | 86 ms | 253 ms | 368 ms | 98.67 ms | 59.56 ms | 695 ms |

| Stat      | 1%      | 2.5%    | 50%     | 97.5%   | Avg     | Stdev  | Min     |
| Req/Sec   | 81      | 81      | 462     | 513     | 401.7   | 131.53 | 81      |
| Bytes/Sec | 8.02 kB | 8.02 kB | 45.8 kB | 50.8 kB | 39.8 kB | 13 kB  | 8.02 kB |

Req/Bytes counts sampled once per second.
# of samples: 10

0 2xx responses, 4017 non 2xx responses
4k requests in 10.03s, 398 kB read
```

And these are the results in the OTE domain (main staekeholder of pilot 5):



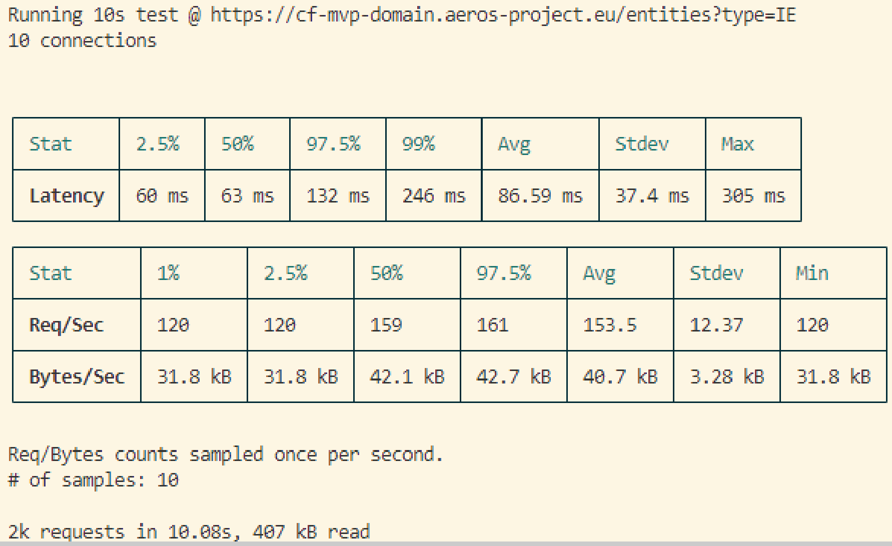
```
10 connections with 4 pipelining factor


| Stat    | 2.5%  | 50%   | 97.5%  | 99%    | Avg     | Stdev    | Max    |
| Latency | 61 ms | 69 ms | 130 ms | 206 ms | 80.4 ms | 41.22 ms | 515 ms |

| Stat      | 1%      | 2.5%    | 50%     | 97.5%   | Avg     | Stdev   | Min     |
| Req/Sec   | 306     | 306     | 510     | 554     | 493.4   | 68.44   | 306     |
| Bytes/Sec | 30.3 kB | 30.3 kB | 50.5 kB | 54.8 kB | 48.8 kB | 6.77 kB | 30.3 kB |

Req/Bytes counts sampled once per second.
# of samples: 10

0 2xx responses, 4934 non 2xx responses
5k requests in 10.07s, 488 kB read
```

The second test showed these results on the UPV domain:



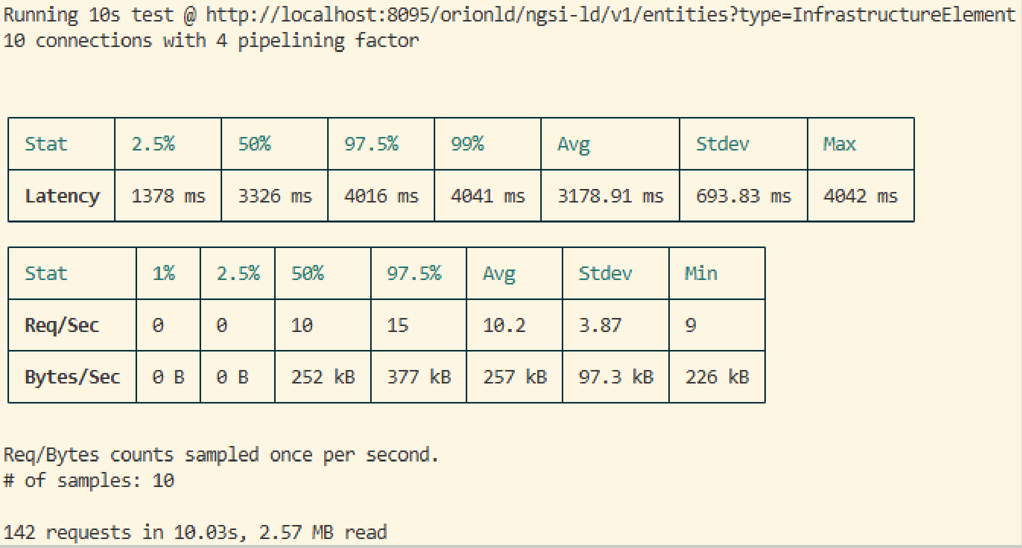
```
Running 10s test @ http://10.108.77.245:8080/entities?type=IE
10 connections


┌─────────┬──────┬───────┬────────┬────────┬──────────┬──────────┬────────┐
│ Stat    │ 2.5% │ 50%   │ 97.5%  │ 99%    │ Avg      │ Stdev    │ Max    │
├─────────┼──────┼───────┼────────┼────────┼──────────┼──────────┼────────┤
│ Latency │ 8 ms │ 41 ms │ 113 ms │ 131 ms │ 45.77 ms │ 29.01 ms │ 162 ms │
└─────────┴──────┴───────┴────────┴────────┴──────────┴──────────┴────────┘
┌───────────┬─────────┬─────────┬─────────┬─────────┬─────────┬───────┬─────────┐
│ Stat      │ 1%      │ 2.5%    │ 50%     │ 97.5%   │ Avg     │ Stdev │ Min     │
├───────────┼─────────┼─────────┼─────────┼─────────┼─────────┼───────┼─────────┤
│ Req/Sec   │ 203     │ 203     │ 216     │ 222     │ 215.1   │ 5.45  │ 203     │
├───────────┼─────────┼─────────┼─────────┼─────────┼─────────┼───────┼─────────┤
│ Bytes/Sec │ 35.9 kB │ 35.9 kB │ 38.2 kB │ 39.3 kB │ 38.1 kB │ 965 B │ 35.9 kB │
└───────────┴─────────┴─────────┴─────────┴─────────┴─────────┴───────┴─────────┘

Req/Bytes counts sampled once per second.
# of samples: 10

2k requests in 10.02s, 381 kB read
```

These are the results on the CloudFerro domain:



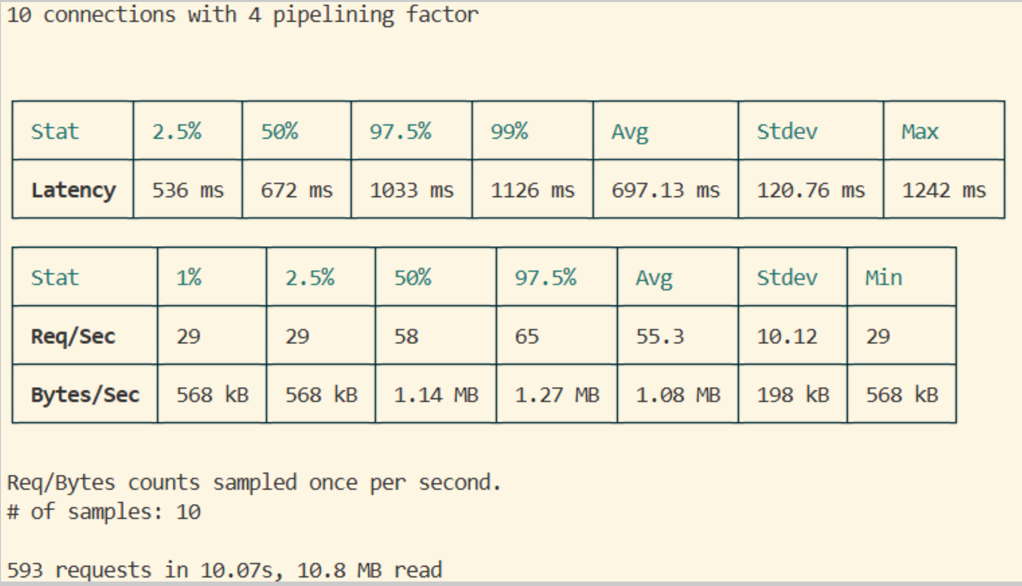
```
Running 10s test @ https://cf-mvp-domain.aeros-project.eu/entities?type=IE
10 connections


┌─────────┬───────┬───────┬────────┬────────┬──────────┬─────────┬────────┐
│ Stat    │ 2.5%  │ 50%   │ 97.5%  │ 99%    │ Avg      │ Stdev   │ Max    │
├─────────┼───────┼───────┼────────┼────────┼──────────┼─────────┼────────┤
│ Latency │ 60 ms │ 63 ms │ 132 ms │ 246 ms │ 86.59 ms │ 37.4 ms │ 305 ms │
└─────────┴───────┴───────┴────────┴────────┴──────────┴─────────┴────────┘
┌───────────┬─────────┬─────────┬─────────┬─────────┬─────────┬─────────┬─────────┐
│ Stat      │ 1%      │ 2.5%    │ 50%     │ 97.5%   │ Avg     │ Stdev   │ Min     │
├───────────┼─────────┼─────────┼─────────┼─────────┼─────────┼─────────┼─────────┤
│ Req/Sec   │ 120     │ 120     │ 159     │ 161     │ 153.5   │ 12.37   │ 120     │
├───────────┼─────────┼─────────┼─────────┼─────────┼─────────┼─────────┼─────────┤
│ Bytes/Sec │ 31.8 kB │ 31.8 kB │ 42.1 kB │ 42.7 kB │ 40.7 kB │ 3.28 kB │ 31.8 kB │
└───────────┴─────────┴─────────┴─────────┴─────────┴─────────┴─────────┴─────────┘

Req/Bytes counts sampled once per second.
# of samples: 10

2k requests in 10.08s, 407 kB read
```

These are the results on the NCSRD domain:



```
Running 10s test @ http://localhost:8095/orionld/ngsi-ld/v1/entities?type=InfrastructureElement
10 connections with 4 pipelining factor


┌─────────┬─────────┬─────────┬─────────┬─────────┬────────────┬───────────┬─────────┐
│ Stat    │ 2.5%    │ 50%     │ 97.5%   │ 99%     │ Avg        │ Stdev     │ Max     │
├─────────┼─────────┼─────────┼─────────┼─────────┼────────────┼───────────┼─────────┤
│ Latency │ 1378 ms │ 3326 ms │ 4016 ms │ 4041 ms │ 3178.91 ms │ 693.83 ms │ 4042 ms │
└─────────┴─────────┴─────────┴─────────┴─────────┴────────────┴───────────┴─────────┘
┌───────────┬─────┬──────┬────────┬────────┬────────┬─────────┬────────┐
│ Stat      │ 1%  │ 2.5% │ 50%    │ 97.5%  │ Avg    │ Stdev   │ Min    │
├───────────┼─────┼──────┼────────┼────────┼────────┼─────────┼────────┤
│ Req/Sec   │ 0   │ 0    │ 10     │ 15     │ 10.2   │ 3.87    │ 9      │
├───────────┼─────┼──────┼────────┼────────┼────────┼─────────┼────────┤
│ Bytes/Sec │ 0 B │ 0 B  │ 252 kB │ 377 kB │ 257 kB │ 97.3 kB │ 226 kB │
└───────────┴─────┴──────┴────────┴────────┴────────┴─────────┴────────┘

Req/Bytes counts sampled once per second.
# of samples: 10

142 requests in 10.03s, 2.57 MB read
```

Ant these in the OTE domain:



```
10 connections with 4 pipelining factor


┌─────────┬────────┬────────┬─────────┬─────────┬───────────┬───────────┬─────────┐
│ Stat    │ 2.5%   │ 50%    │ 97.5%   │ 99%     │ Avg       │ Stdev     │ Max     │
├─────────┼────────┼────────┼─────────┼─────────┼───────────┼───────────┼─────────┤
│ Latency │ 536 ms │ 672 ms │ 1033 ms │ 1126 ms │ 697.13 ms │ 120.76 ms │ 1242 ms │
└─────────┴────────┴────────┴─────────┴─────────┴───────────┴───────────┴─────────┘
┌───────────┬────────┬────────┬─────────┬─────────┬─────────┬────────┬────────┐
│ Stat      │ 1%     │ 2.5%   │ 50%     │ 97.5%   │ Avg     │ Stdev  │ Min    │
├───────────┼────────┼────────┼─────────┼─────────┼─────────┼────────┼────────┤
│ Req/Sec   │ 29     │ 29     │ 58      │ 65      │ 55.3    │ 10.12  │ 29     │
├───────────┼────────┼────────┼─────────┼─────────┼─────────┼────────┼────────┤
│ Bytes/Sec │ 568 kB │ 568 kB │ 1.14 MB │ 1.27 MB │ 1.08 MB │ 198 kB │ 568 kB │
└───────────┴────────┴────────┴─────────┴─────────┴─────────┴────────┴────────┘

Req/Bytes counts sampled once per second.
# of samples: 10

593 requests in 10.07s, 10.8 MB read
```

As can be seen from both tests on all domains, KrakenD is very resilient to the load tests, being able of taking a load of over five thousand petitions per second if the token is invalid and thus KrakenD does not need to send any traffic to the backend.

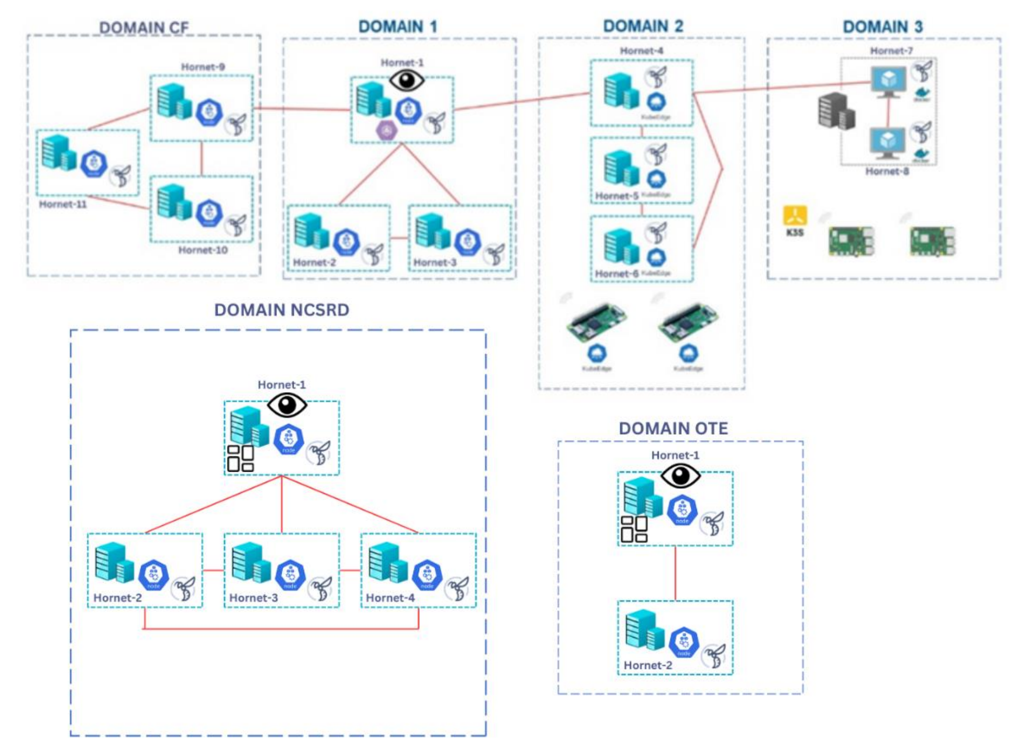
The discrepancies between the number of petitions in the first test can be attributed to the distance between the UPV and CF / NCSRD / OTE domains. Since the tests were made from within the UPV it takes considerably less time for the petitions to reach the server.

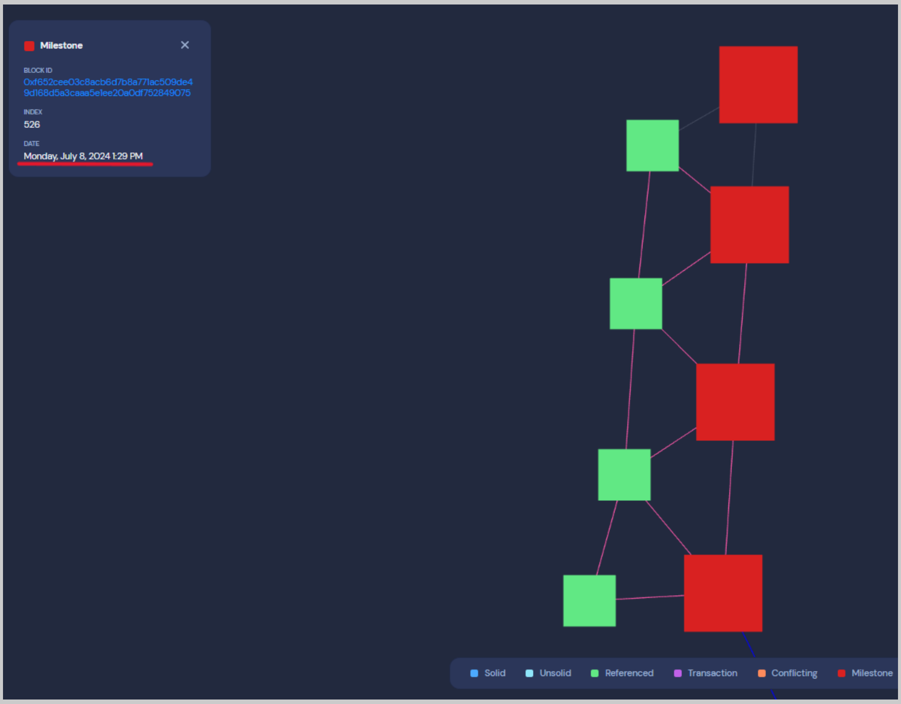
As for the second test, the number of petitions is similar in the first two cases since most of the bottleneck here happens when KrakenD processes the valid petition into the backend and returns the valid response, not so much due to the distance or the amount of data, even though it is still relevant. In the third and fourth cases, a note must be made on the amount of data read, with it being 2.57MB and 10.8MB respectively while in the other tests it was 380 – 400KB, this is directly responsible for the low amount of requests per second (14 and 59, depending on the test), as well as the state of connectivity, but it is still above the baseline

KPI 1.4.8 % trusted scenarios that make use of IOTA's DLT

Table 51: KPI 1.4.8 % trusted scenarios that make use of IOTA's DLT

KPI ID number and partner resp.	**KPI 1.4.8**
KPI Name	**% trusted scenarios that make use of IOTA's DLT**
Description	Being able to share continuum-wide relevant information using the IOTA DLT via Hornet node peer-to-peer message exchange.
Motivation	One of the main tools that bring trust into the aerOS platform is IOTA, the number of messages shared by the different nodes is crucial for the platform to understand the global status of the continuum.
Target value	5 data transactions per minute

Prerequisites	*The aerOS* multiple interconnected domains		
aerOS components (task)	*IOTA (T4.5), Trust score calculator* (T4.5), Self* tools (T3.5), Use case tools (T5.2). Multiple elements of the aerOS continuum will generate events that need to be registered in the DLT and will share them in the IOTA Tangle to all the other IEs in the continuum.		
Evaluation means	*The IOTA tools themselves will be used to monitor the traffic of any given deployment.*		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***1 data transaction per minute***	***3-4 transactions per minute in a single domain demonstrated (see below)***	***At least 3-4 transactions per minute per pilot domain***
Outcome elaboration (M38)	Deployed 11 separate Hornet Nodes in a controlled testing environment among 4 different interconnected domains and another 4 Hornet nodes in the NCSRD pilot and 2 in the OTE pilot testing environments. All elements necessary deployed, testing integration with self* features and the trust score calculator. For these tests the following architecture is used:  The events generated will be sent to the hornet nodes in the CloudFerro domain (hornet-9, hornet-10 and hornet-11). The inclusion of these data blocks will be verified by the Coordinator found in Domain 1 in the UPV infrastructure in the form of milestones, which are launched automatically by the coordinator every 20 seconds. The traffic received by node "hornet-9" in the span of a minute can be seen in the following images:		



Milestone

BLOCK ID
0xf652cee03c8acb6d7b8a771ac509de49d168d5a3caaa5e1ee20a0df752849075

INDEX
526

DATE
Monday, July 8, 2024 1:29 PM

Milestone

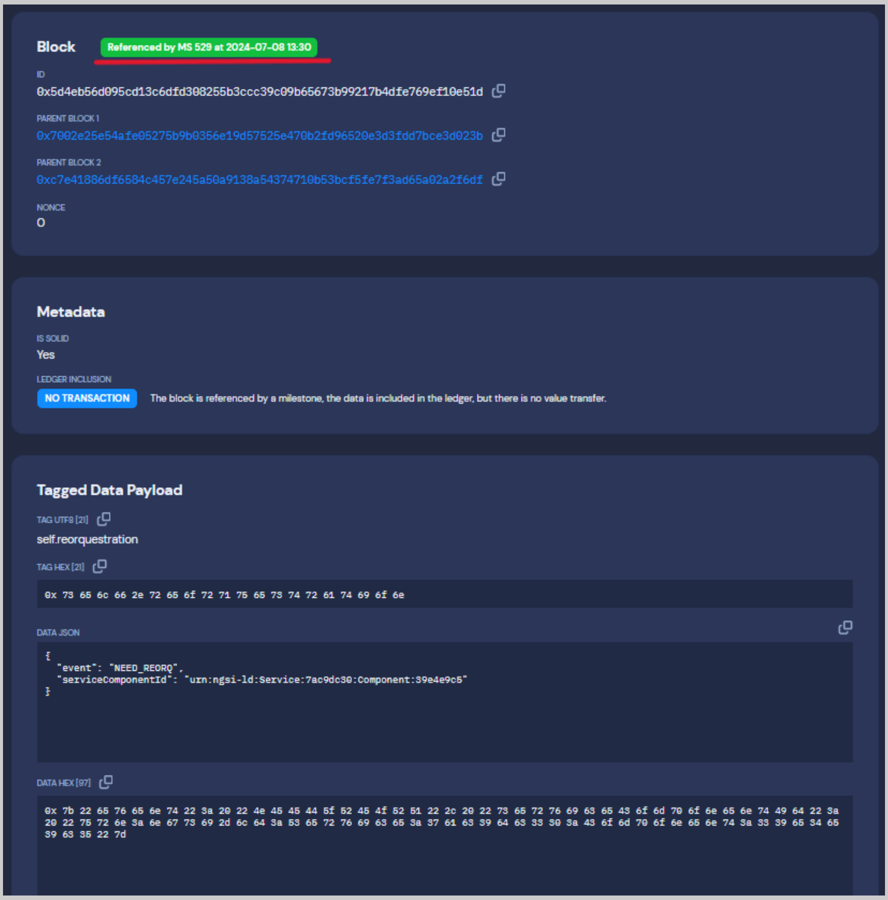
BLOCK ID
0x5162f39af69e2419d381064a563e01e4bf5e2f10853fba72c802b1a0f65d196e

INDEX
529

DATE
Monday, July 8, 2024 1:30 PM

These show that in the span of a minute roughly 4 data transactions are made and that all of them are referenced by a milestone and thus included in the DLT (note the green colour of the blocks, which showcases this reference).

An example of one such referenced block can be seen below:



Block Referenced by MS 529 at 2024-07-08 13:30

ID
0x5d4eb56d095cd13c6dfd308255b3ccc39c09b65673b99217b4dfe769ef10e51d

PARENT BLOCK 1
0x7002e25e54afe05275b9b0356e19d57525e470b2fd96520e3d3fdd7bce3d023b

PARENT BLOCK 2
0xc7e41886df6584c457e245a50a9138a54374710b53bcf5fe7f3ad65a02a2f6df

NONCE
0

Metadata

IS SOLID
Yes

LEDGER INCLUSION
NO TRANSACTION The block is referenced by a milestone, the data is included in the ledger, but there is no value transfer.

Tagged Data Payload

TAG UTF8 [21]
self.reorquestration

TAG HEX [21]

```
0x 73 65 6c 66 2e 72 65 6f 72 71 75 65 73 74 72 61 74 69 6f 6e
```

DATA JSON

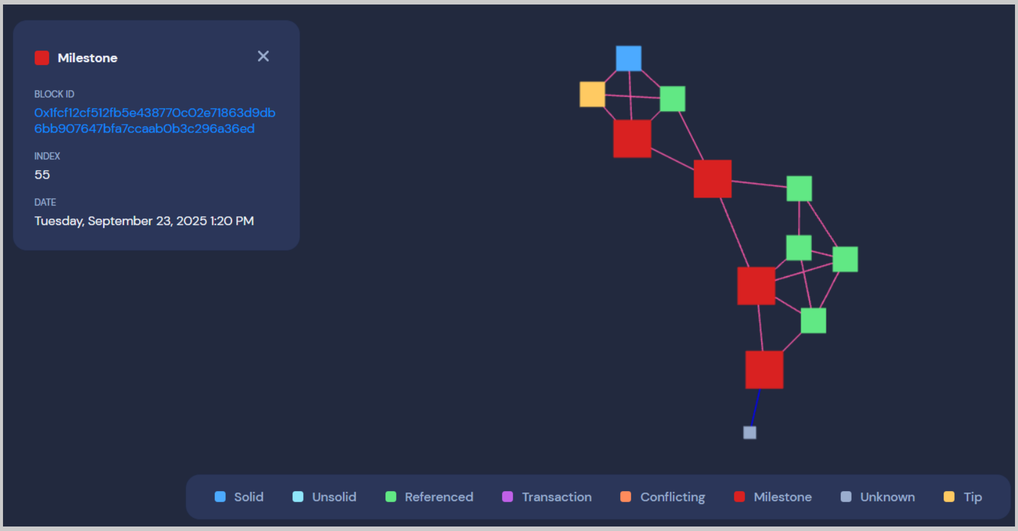
```
{
  "event": "NEED_REORQ",
  "serviceComponentId": "urn:ngsi-ld:Service:7ac9dc30:Component:39e4e9c5"
}
```

DATA HEX [97]

```
0x 7b 22 65 76 65 6e 74 22 3a 20 22 4e 45 45 44 5f 52 45 4f 52 51 22 2c 20 22 73 65 72 76 69 63 65 43 6f 6d 70 6f 6e 65 6e 74 49 64 22 3a
20 22 75 72 6e 3a 6e 67 73 69 2d 6c 64 3a 53 65 72 76 69 63 65 3a 37 61 63 39 64 63 33 30 3a 43 6f 6d 70 6f 6e 65 6e 74 3a 33 39 65 34 65
39 63 35 22 7d
```

This block is referenced by milestone 529 seen in the previous image and is thus included in the DLT. Also, the contents of the block can be seen below, the tag "*self.reorquestation*" indicates that this is a reorchestration petition sent by the self* components, referencing the ID of such component.

Additional tests following the same process were made in the NCSRD and OTE domains, in which roughly 7 petitions were being made per minute, as can be seen in the image below, showcasing milestone 55 (please note that the blue and yellow colours indicate two blocks that have not yet been included officially in the DLT via a milestone):



🟥 **Milestone**

BLOCK ID
0x1fcf12cf512fb5e438770c02e71863d9db6bb907647bfa7ccaab0b3c296a36ed

INDEX
55

DATE
Tuesday, September 23, 2025 1:20 PM

Now after a new milestone (milestone 58) the “tip” block is added to the DLT and changes colour to green, as can be seen below.

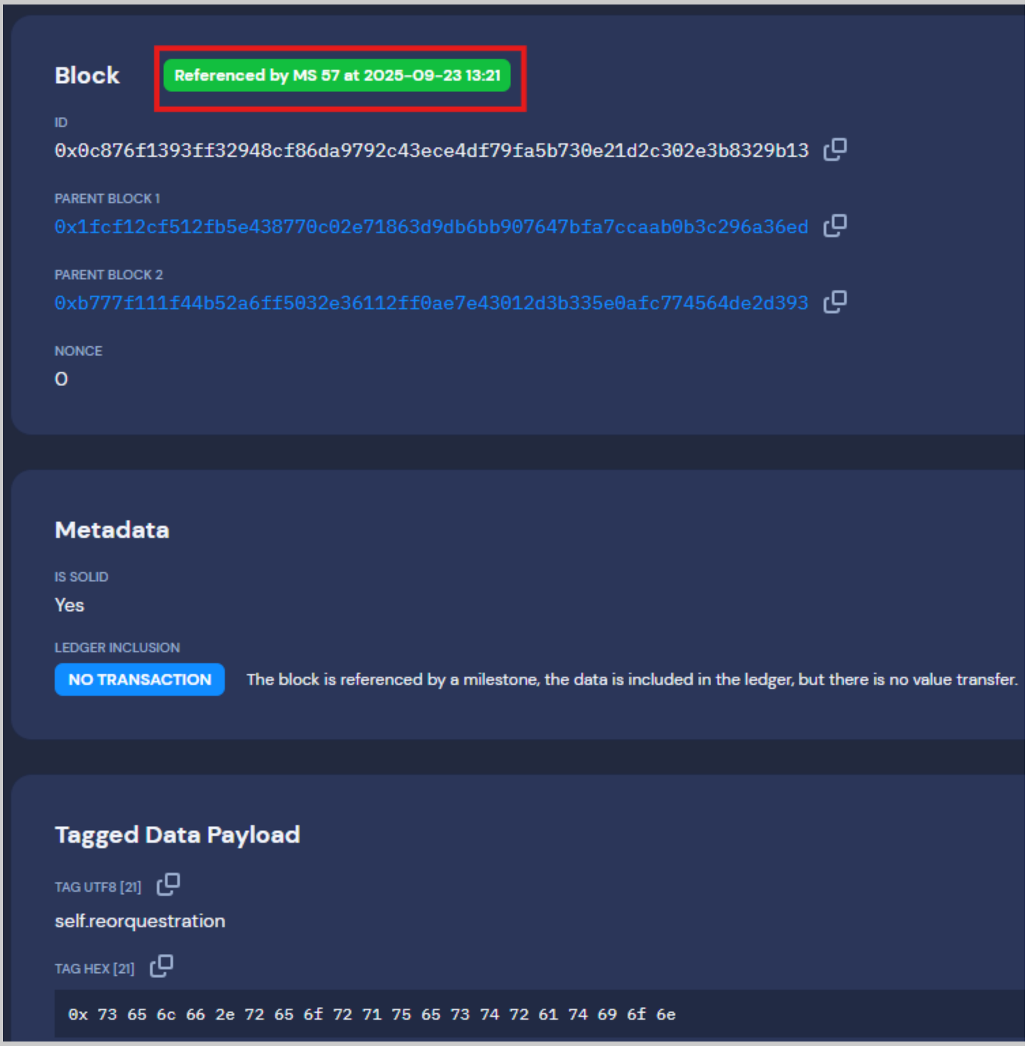
Milestone

BLOCK ID
0x9a06c111bdbebceee46e68473cb732eca984cc0beb4a4ffd9228c103f930b230

INDEX
58

DATE
Tuesday, September 23, 2025 1:21 PM

Below an example of one of the confirmed blocks can be seen, note that the PARENT BLOCK 1 hexcode is the same as milestone 55 since the block was added after said milestone.



Block Referenced by MS 57 at 2025-09-23 13:21

ID
0x0c876f1393ff32948cf86da9792c43ece4df79fa5b730e21d2c302e3b8329b13

PARENT BLOCK 1
0x1fcf12cf512fb5e438770c02e71863d9db6bb907647bfa7ccaab0b3c296a36ed

PARENT BLOCK 2
0xb777f111f44b52a6ff5032e36112ff0ae7e43012d3b335e0afc774564de2d393

NONCE
0

Metadata

IS SOLID
Yes

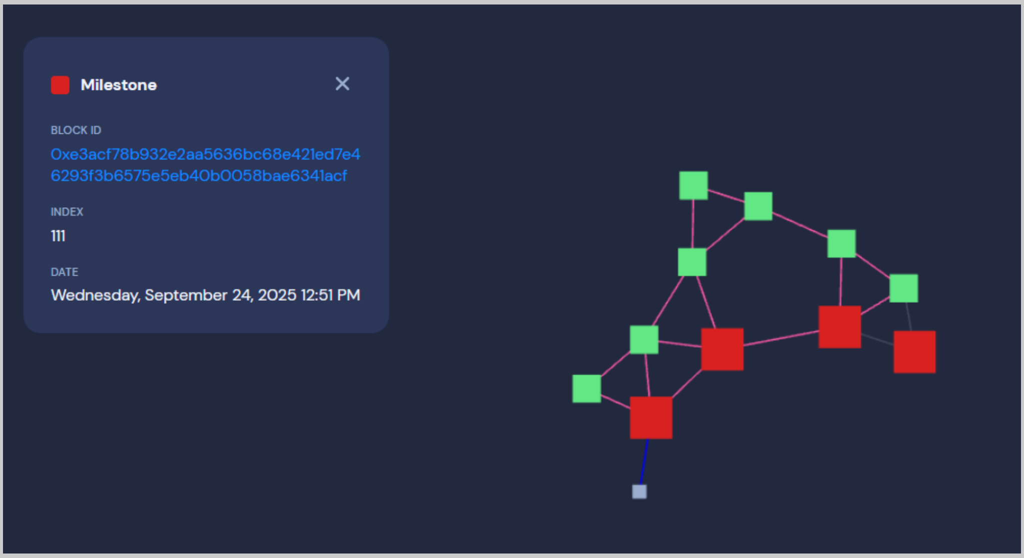
LEDGER INCLUSION
NO TRANSACTION The block is referenced by a milestone, the data is included in the ledger, but there is no value transfer.

Tagged Data Payload

TAG UTF8 [21]
self.reorquestration

TAG HEX [21]
0x 73 65 6c 66 2e 72 65 6f 72 71 75 65 73 74 72 61 74 69 6f 6e

The same tests were made in the OTE domain, which showed similar levels of transactions per minute:



BLOCK ID
0xe3acf78b932e2aa5636bc68e421ed7e46293f3b6575e5eb40b0058bae6341acf

INDEX
111

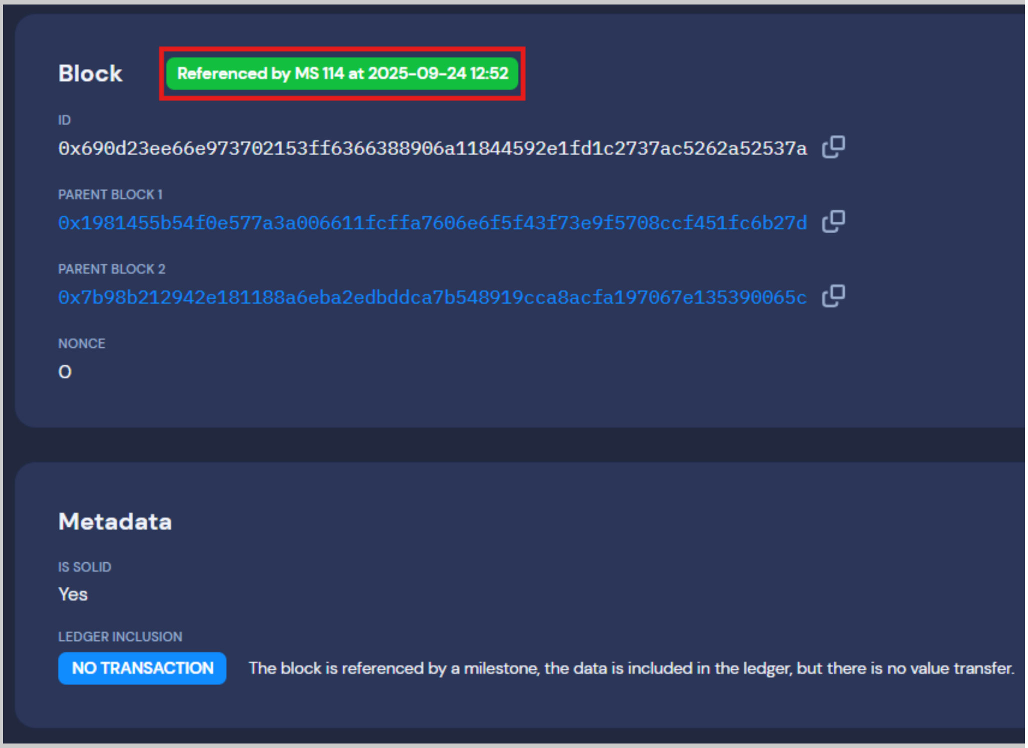
DATE
Wednesday, September 24, 2025 12:51 PM

BLOCK ID
0x78e017322da54f0cd51e2b10d79e031fbd97c8742271716c75e1f5584c9427b9

INDEX
114

DATE
Wednesday, September 24, 2025 12:52 PM

Now an example of a referenced data block can be seen below, where it was included in the DLT the moment it was referenced by milestone 114:

	 Block Referenced by MS 114 at 2025-09-24 12:52 ID 0x690d23ee66e973702153ff6366388906a11844592e1fd1c2737ac5262a52537a PARENT BLOCK 1 0x1981455b54f0e577a3a006611fcffa7606e6f5f43f73e9f5708ccf451fc6b27d PARENT BLOCK 2 0x7b98b212942e181188a6eba2edbddca7b548919cca8acfa197067e135390065c NONCE 0 Metadata IS SOLID Yes LEDGER INCLUSION NO TRANSACTION The block is referenced by a milestone, the data is included in the ledger, but there is no value transfer.

KPI 1.4.9 Network overload limit due to the usage of IOTA and Tangle

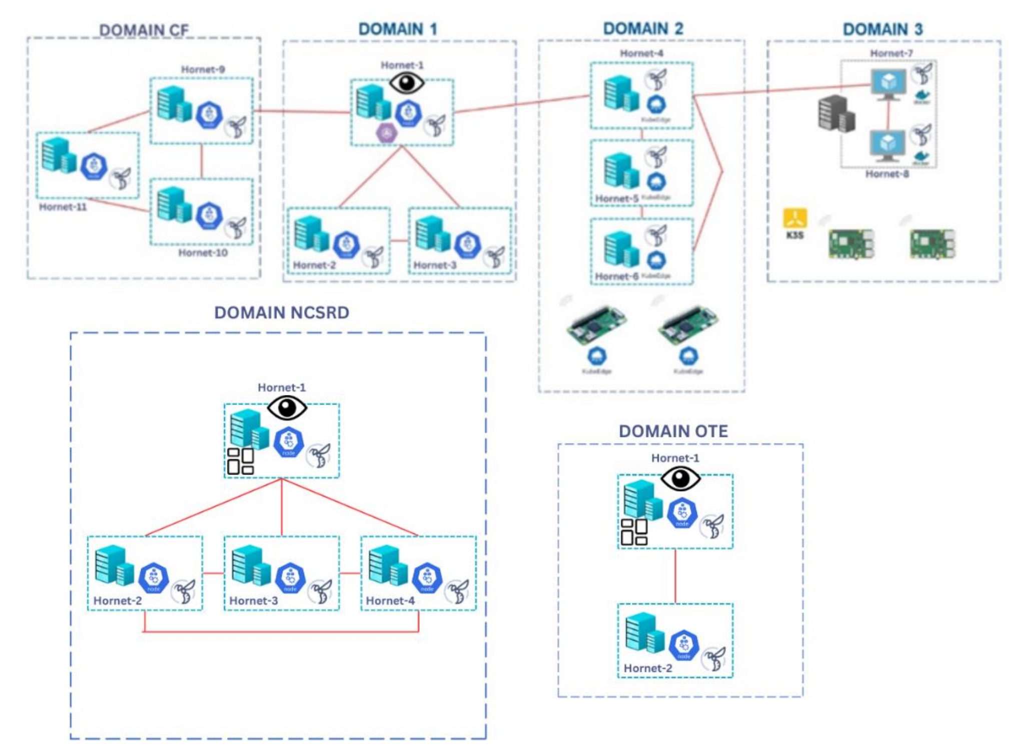
Table 52: KPI 1.4.9 Network overload limit due to the usage of IOTA and Tangle

KPI ID number and partner resp.	**KPI 1.4.9**
KPI Name	**Network overload limit due to the usage of IOTA and Tangle**
Description	Creating and implementing an IOTA Tangle network of nodes that share information between them without managing to overload the network.
Motivation	An IOTA Tangle network allows for peer to peer sharing of information between nodes, benefiting the entire continuum. However, it must be done in a way that does not completely overload the network.
Target value	aerOS private IOTA Tangle network deployed and running without increasing the network load by more than 30%
Prerequisites	The aerOS multiple interconnected domains
aerOS components (task)	IOTA (T4.5), Trust score calculator (T4.5), Self-* components (T3.5), Use-cases tools (T5.2). Multiple elements of the aerOS continuum will generate events that need to be registered in the DLT and will share them in the IOTA Tangle to all the other IEs in the continuum.

<table>
<tr><td>Evaluation means</td><td colspan="3">The impact of IOTA on the network will be evaluated using the Kubernetes tools themselves, as well as the IOTA metrics plugins. A custom script will be used to generate an unusual load on the environment.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>aerOS services up and running as expected.</i></td><td><i>Minimal load increase with the expected network traffic.</i></td><td><i>Minimal load increased with the expected network traffic. At least 30 petitions per second are needed to increase the network load by 30%.</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">

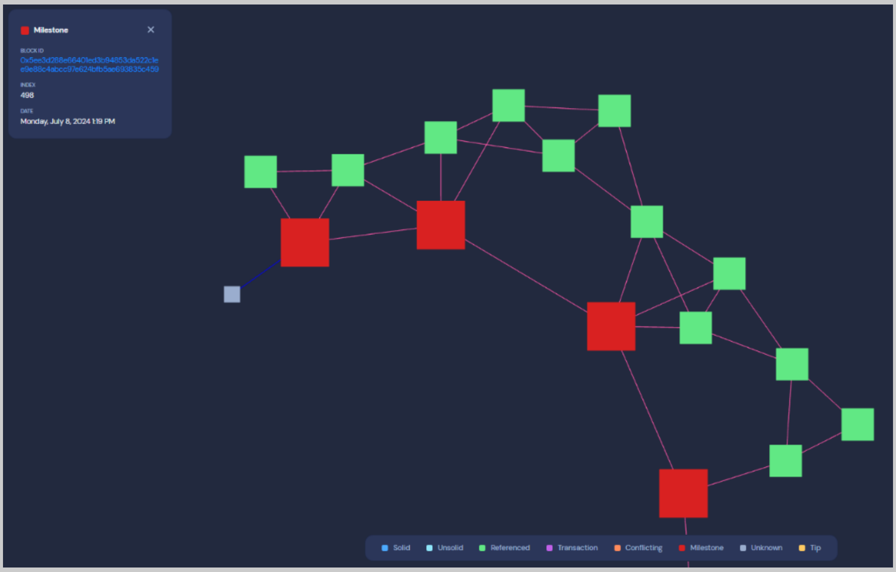
Deployed 11 separate Hornet Nodes in a controlled testing environment among 4 different interconnected domains and another 4 Hornet nodes in the NCSRD pilot and 2 in the OTE pilot testing environments. All elements necessary deployed, testing integration with self* features and the trust score calculator.

For these tests the following architecture is used:



For these tests a custom script will be used to generate an unusually high amount of petitions (around 12 per minute, double of what is expected at the current time) and see if there's a noticeable difference in the cluster metrics before performing a stress test of the network, with dozens of petitions per second. The following images show 12 messages being received roughly in a minute:

</td></tr>
</table>



Milestone

BLOCK ID
0x5ee3d288e66401ed3b94853da522c1e
e9e88c4abcc97e624bfb5ae693835c459

INDEX
498

DATE
Monday, July 8, 2024 1:19 PM

Milestone

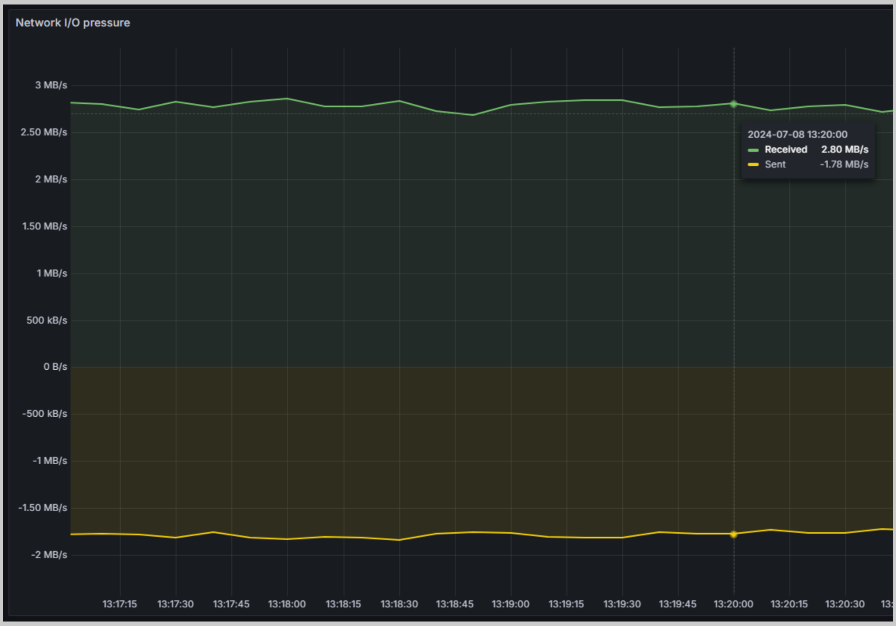
BLOCK ID
0xa2a612edef4027a1a74f0e0042d48a95
c56d448edcf189e5b107b341f6061d35

INDEX
501

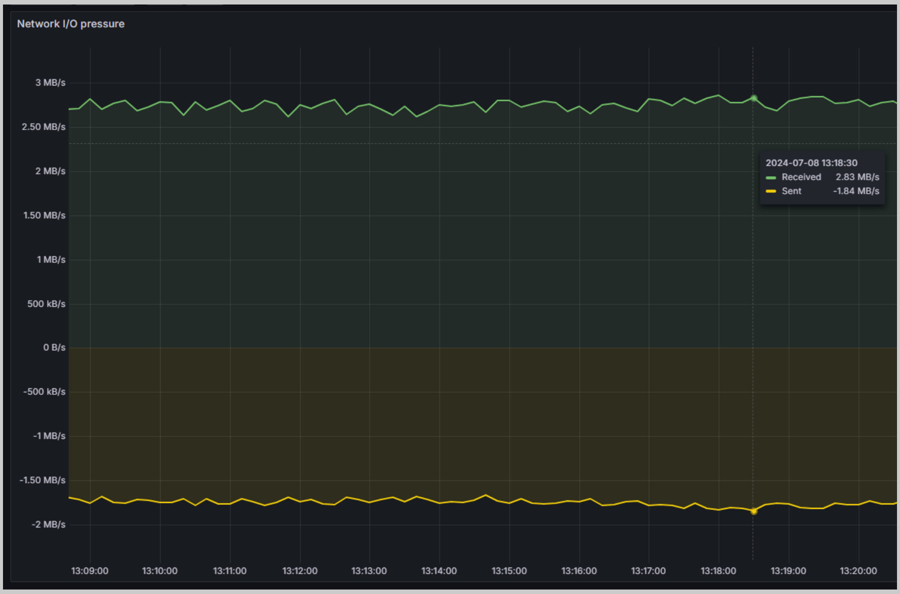
DATE
Monday, July 8, 2024 1:20 PM

Note again that the red blocks represent milestones being launched every 20 seconds while the green ones represent the new blocks being added to the DLT.

As can be seen in the upper left corner of the image the timestamp for the test is 13:20, checking that timestamp in the cluster Grafana metrics component shows the following:



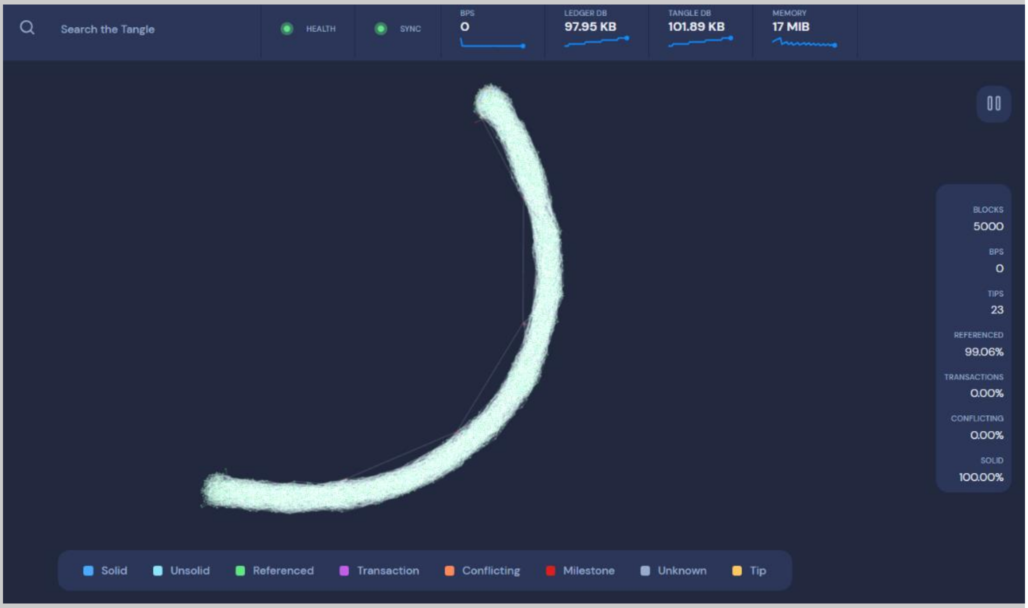
In the entire minute where the test was being made the difference in network load is practically non-existent, this can be further verified by checking the traffic in a wider time bracket:



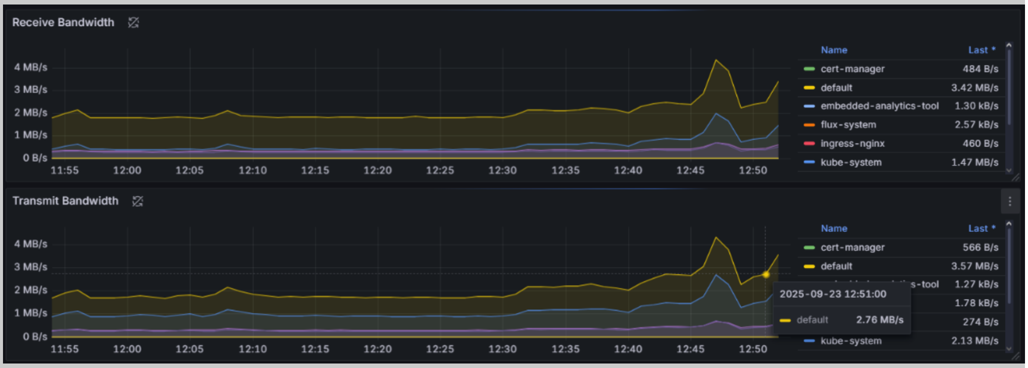
It can be safely said that there is no noticeable difference in the network load when the different IOTA Hornet nodes are sharing information.

Additionally, stress tests were performed in the NCSRD domain to ascertain how many petitions per second needed to be made in order to reach a 30% or higher network load increase. The script would now launch around 30 petitions per second, the results of which can be seen below.

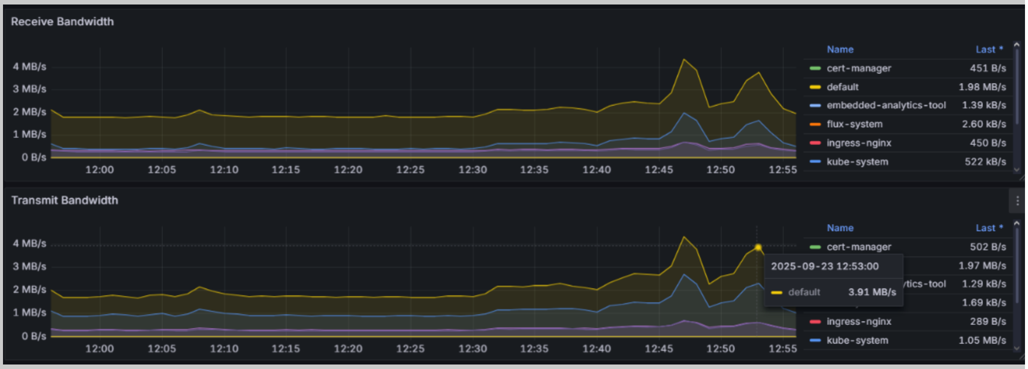
The first image showcases the Hornet Dashboard block visualization tool after one of the tests, there are so many blocks added that they can barely be seen individually. 5000 blocks were uploaded in less than 3 minutes.



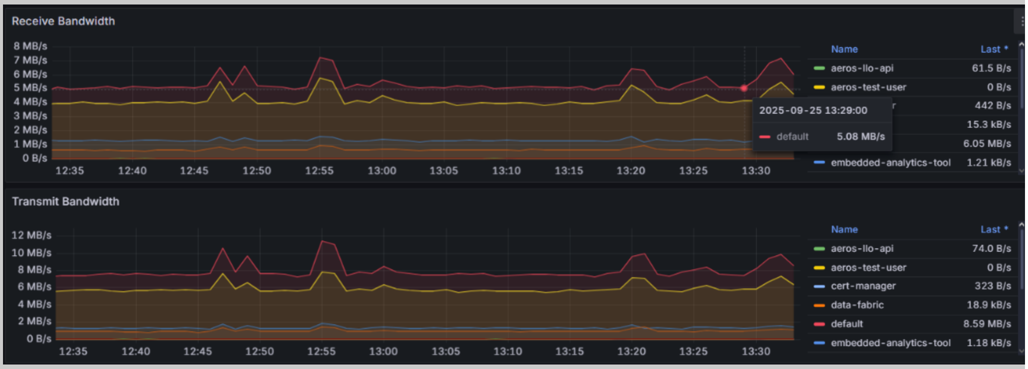
Now the impact on the network can be seen below, first with the starting point before the test started with 2.76MB/s in traffic:



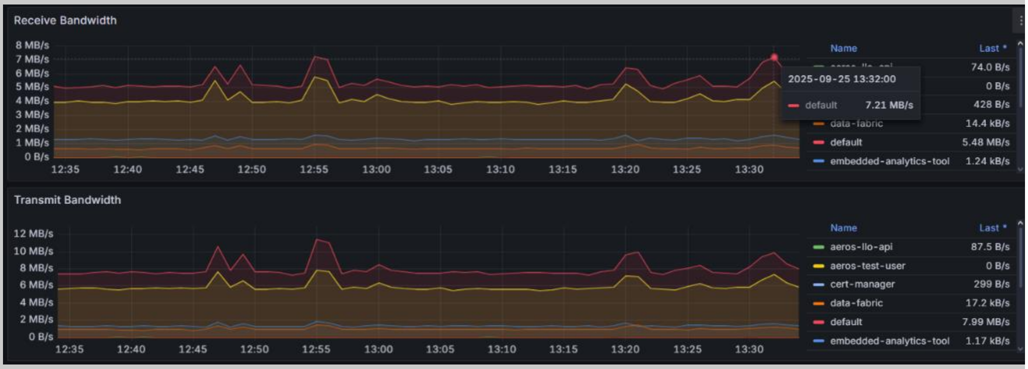
And finally just after the test is concluded, with a peak of 3.91MB/s.



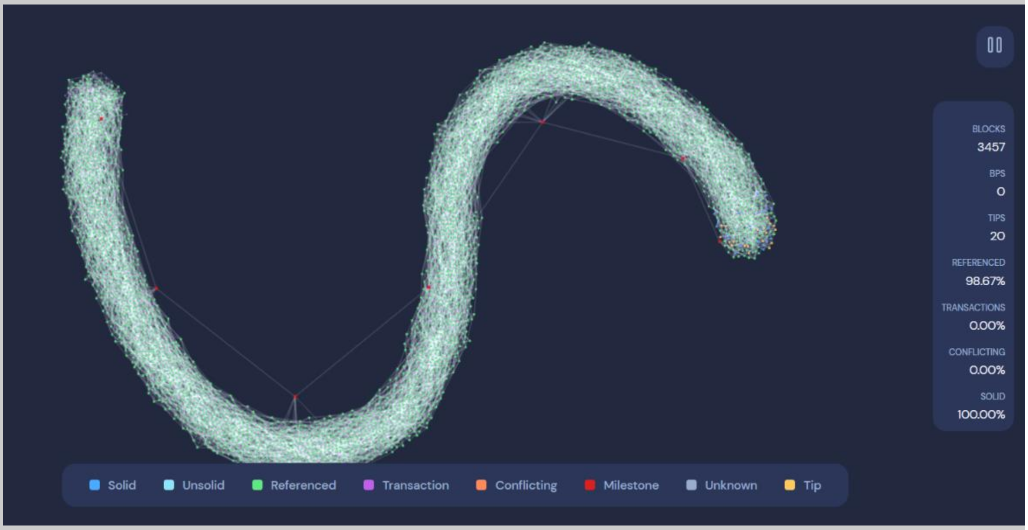
The same tests were performed in the OTE pilot testing environment, where similar results were witnessed. Before the test the receive bandwidth was of 5.06 MB/s:



And at the peak it reached 7.21 MB/s, well over 30%.



The dashboard visualizer shows a similar result as in the NCSRD cluster.



KPI 1.4.10 Trust Score Recalculation and Resource Balance

Table 53: KPI 1.4.10 Trust Score Recalculation and Resource Balance

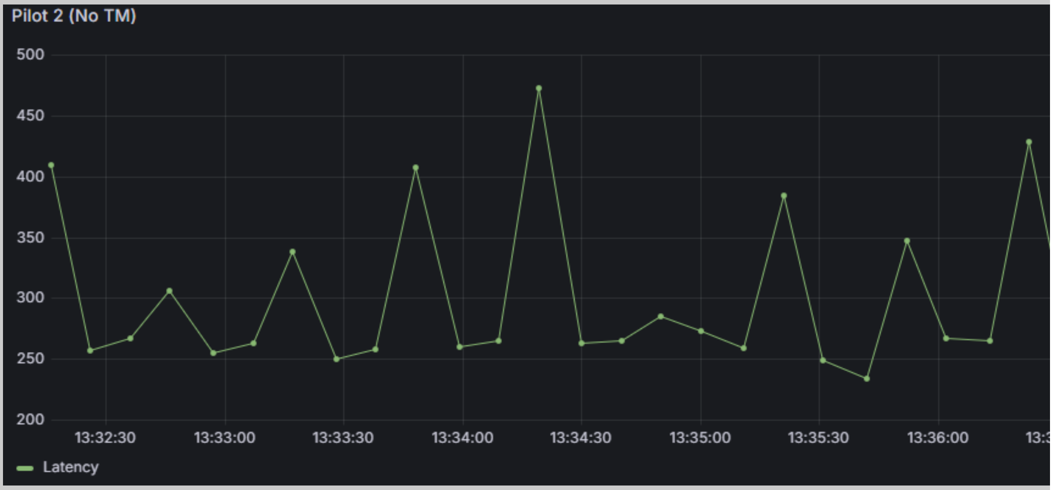
KPI ID number and partner resp.	**KPI 1.4.10**
KPI Name	**Trust Score Recalculation and Resource Balance**

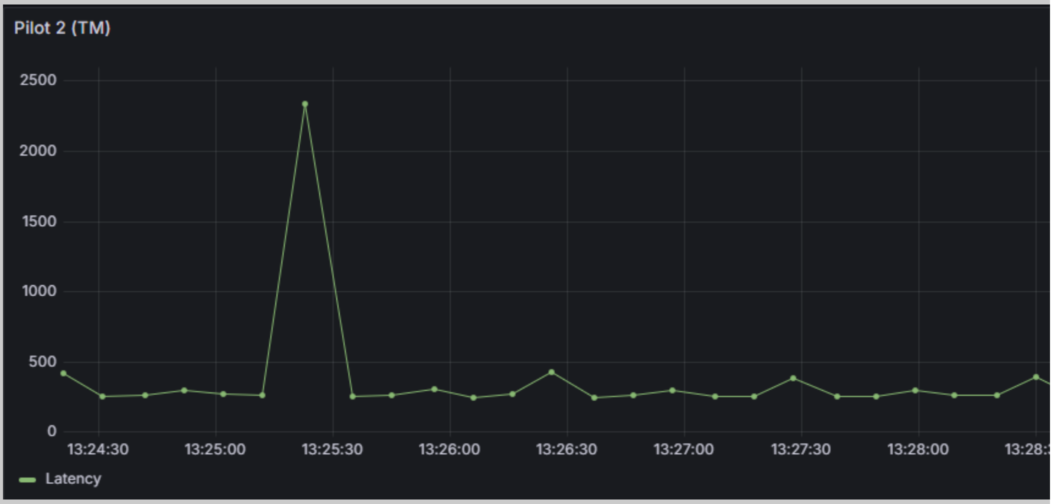
Description	This KPI evaluates the efficiency of the trust score recalculation process in relation to the consumption of aerOS resources, ensuring that the operational demands of maintaining updated trust scores do not lead to excessive use of resources.		
Motivation	This KPI is motivated by the imperative to harmonize the necessity for dynamic and robust trust management with the overarching need to preserve system performance and reliability		
Target value	Mean increase in resource usage due to trust score recalculation activities < 30% regular use		
Prerequisites	Trust manager and Orion-LD Context Broker running in a domain		
aerOS components (task)	Self-awareness (T3.5), Context Broker (T4.2), aerOS Trust Component (T4.5)		
Evaluation means	To evaluate this, the latency of queries to the Orion-LD Context Broker will be measured with and without the use of a trust manager in the domain. A custom script has been created to emulate multiple IEs and simulate the load generated by the self-awareness module. Additionally, another custom script has been used to calculate the latency of the queries to the Orion-LD Context Broker. The output of this script is a CSV file containing the latency and timestamps of the queries. This CSV file is later used as input to Grafana for visualizing the results.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***aerOS services up and running as expected***	***<2% in 5 IEs scenario*** ***<1% in 20 IEs scenario***	***The average latency difference remained below 2% across pilots, with no degradation in responsiveness.***
Outcome elaboration (M38)	For the M38 evaluation, latency measurements with and without the trust manager were conducted in the Pilots 2,3,4,5. The exported CSV files were ingested into Grafana using the Infinity data source plugin and visualized as time-series plots. The observed results are consistent with the M24 evaluation: the inclusion of the trust manager introduced no significant increase in query latency. In all pilots, the average latency difference remained below 2%, with no degradation in system responsiveness. This demonstrates that the trust score recalculation and resource balancing mechanisms of aerOS can be applied in real deployment scenarios without negatively impacting performance. The KPI target has therefore been achieved at M38. **Pilot 1.1:** In this scenario the Trust Manager was not deployed, as the use case focuses on predictive maintenance and cognitive digital twins where trust recalculation is not required for the orchestration workflows. Therefore, no latency metrics were collected, and the absence of results is justified by the pilot's scope. **Pilot 1.2:** The scenario in this pilot emphasizes low-latency orchestration and human-robot collaboration, the evaluation proceeded without integrating trust score recalculation. As a result, no KPI metrics are available, and the omission is justified by the functional priorities of the pilot.		

Pilot 1.3: This scenario does not integrate the Trust Score mechanism, as confirmed by the pilot partners. The focus here is on worker-centric dashboards, AR/VR interfaces, and ergonomics rather than trust-based resource balancing. Hence, trust recalculation is out of scope and no KPI metrics have been produced.

Pilot 1.4: In this scenario the Trust Manager was not deployed, as the use case focuses on predictive maintenance and cognitive digital twins where trust recalculation is not required for the orchestration workflows. Therefore, no latency metrics were collected, and the absence of results is justified by the pilot's scope.

Pilot 2



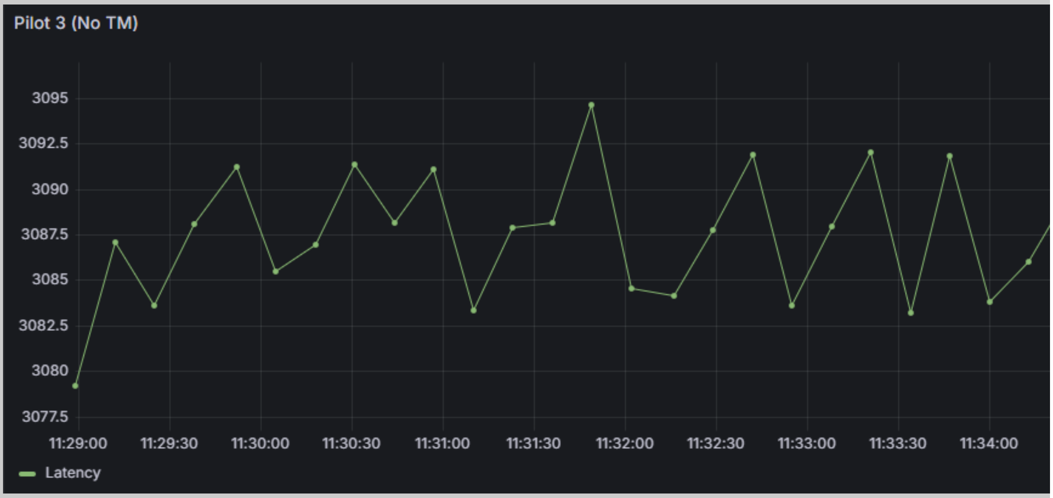


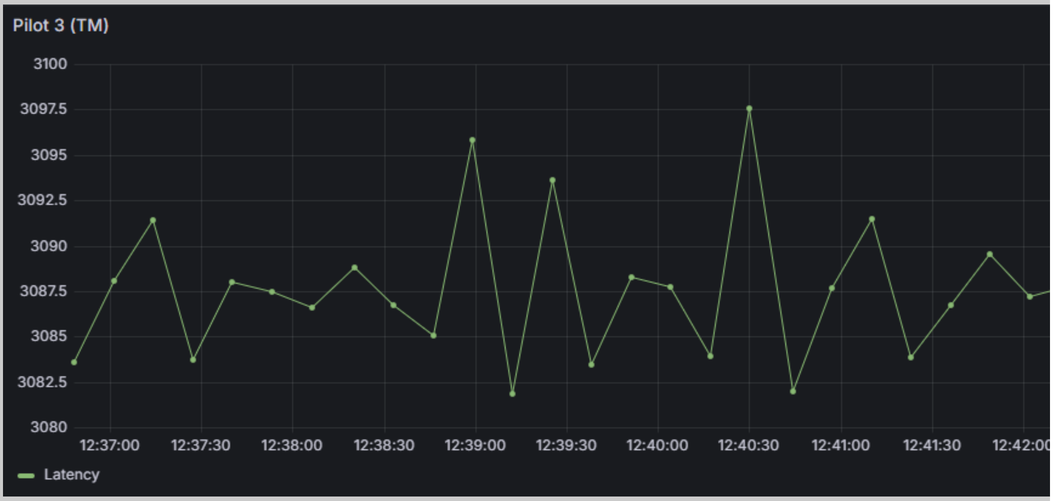
The statistics are as follows:

	Without trust score	With trust score
Mean Latency (ms)	295.70	352.38
Highest Latency (ms)	473.37	2336.13
Lowest Latency (ms)	233.66	241.01

The average latency without the trust manager was 295.70 ms, while with the trust manager it was 352.38 ms. This corresponds to an increase of 19.2%. A single worst-case spike (2336.13 ms) was observed due to load bursts, but overall responsiveness remained within acceptable operational thresholds.

Pilot 3



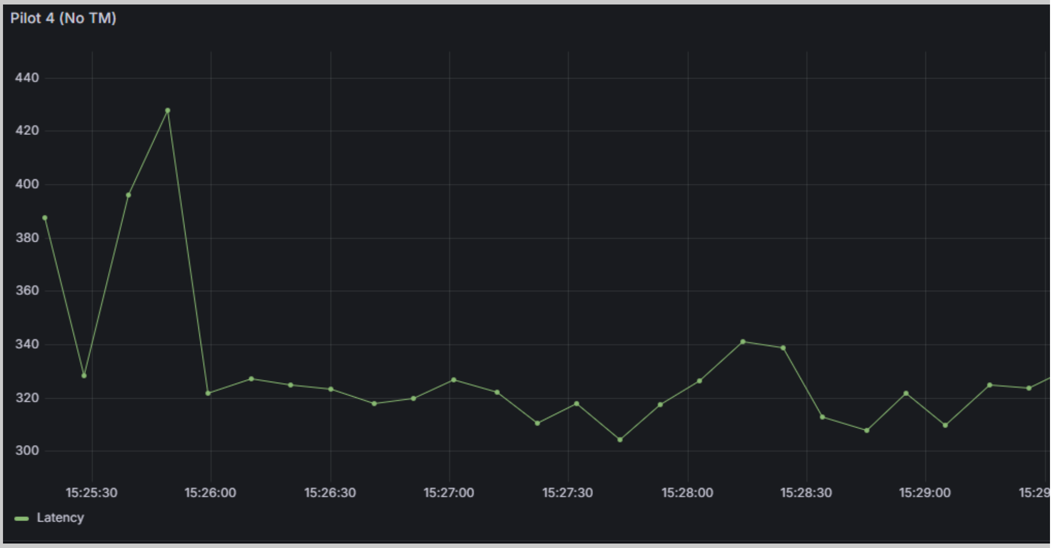


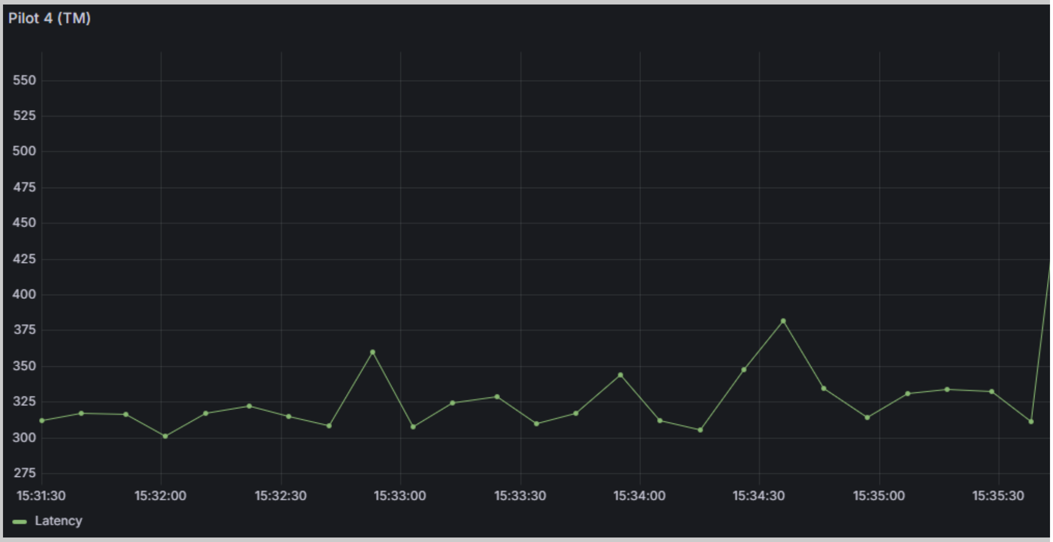
The statistics are as follows:

	Without trust score	With trust score
Mean Latency (ms)	3087.60	3087.65
Highest Latency (ms)	3094.63	3097.62
Lowest Latency (ms)	3079.24	3081.88

The average latency without the trust manager was 3087.60 ms, while with the trust manager it was 3087.65 ms. This represents a negligible increase of 0.0016%, confirming that trust score recalculation introduces no measurable overhead in HPC workflows.

Pilot 4





The statistics are as follows:

	Without trust score	With trust score
Mean Latency (ms)	333.56	330.03
Highest Latency (ms)	427.97	537.57
Lowest Latency (ms)	304.25	301.02

The average latency without the trust manager was 333.56 ms, while with the trust manager it was 330.03 ms. This represents a small improvement of -1.06%, confirming that the inclusion of the trust manager does not negatively affect system responsiveness.

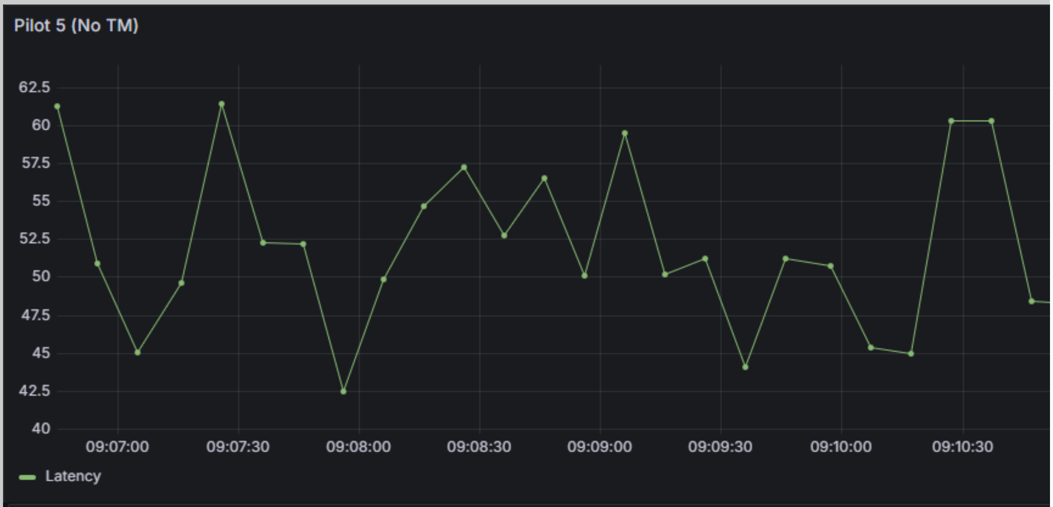
Pilot 4b

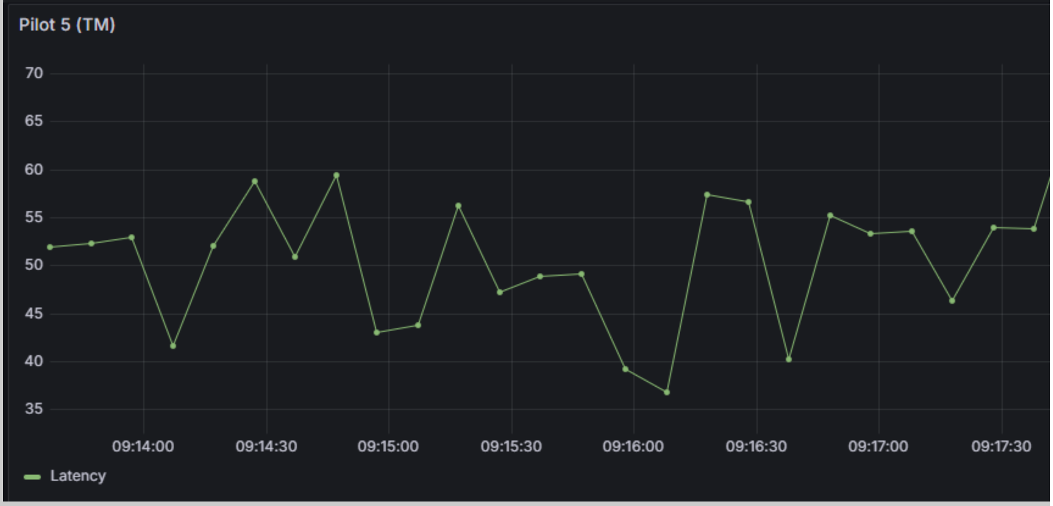
The statistics are as follows:

	Without trust score	With trust score
Mean Latency (ms)	5043.61	5036.21
Highest Latency (ms)	5064.80	5043.65
Lowest Latency (ms)	5030.69	5023.41

The average latency without the trust manager was 5043.61 ms, while with the trust manager it was 5036.21 ms. This corresponds to a negligible difference of -0.15%, showing that trust score recalculation does not introduce overhead.

Pilot 5





The statistics are as follows:

	Without trust score	With trust score
Mean Latency (ms)	51.70	51.16
Highest Latency (ms)	61.40	67.17
Lowest Latency (ms)	42.49	36.75

The average latency without the trust manager was 51.70 ms, while with the trust manager it was 51.16 ms. This represents a small improvement of -1.04%, showing that recalculation does not negatively affect system responsiveness and may in some cases enhance performance.

aerOS self-* and monitoring

KPI 1.5.1 Average overload time of IEs

Table 54: KPI 1.5.1 Average overload time of IEs

KPI ID number and partner resp.	**KPI 1.5.1**
KPI Name	**Average overload time of IEs**

Description	The amount of time an IE has a system load above 80 %, and therefore its performance may decrease considerably		
Motivation	Knowing how long an IE is overloaded allows actions to be taken to reduce its load, keeping it operational for longer		
Target value	Reduction of 20 %		
Prerequisites	Hardware info sub-module of the self-awareness module running on one IE of study		
aerOS components (task)	Self-awareness module (T3.5)		
Evaluation means	Using the hardware info sub-module of the self-awareness module of a node's self-* capabilities set, the node's performance will be obtained over time by measuring the percentage of use of both the CPU and the RAM memory. In this way, it will be possible to estimate the average time that the node remains in an overloaded state. For this purpose, those time slots will be selected in which the CPU or RAM memory usage exceeds 80 %. Several tests will be carried out over a certain period of time in which the aspects mentioned above will be analysed. Subsequently, the times obtained will be averaged in order to know with a certain degree of accuracy the average overload time of the node. Between the two measurement periods all the modules that compose aerOS will undergo changes and improvements, adding new functions and enhancing existing ones. Including the components that form the set of self-* capabilities, such as the self-optimisation and adaptation, the self-orchestrator, the self-scaling, etc. The improvements introduced in the set of self-* capabilities allow optimising the resource consumption of an IE by analysing its current state and predicting future overload events. This makes it possible to re-orchestrate services on other nodes to avoid overloading the IE, optimising its resource usage and reducing the time it remains in an overloaded state due to excessive CPU or RAM usage.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***This value shall be obtained by laboratory load tests to determine the actual average overload time of a node as a function of the actual workload. Through the self-awareness module, the % processor usage and RAM load will be obtained. These values will indicate the total system load***	***37.5 % of the total running time of a node***	***Reduction 30 % of the total running time of a node***

Outcome elaboration (M38)	At the beginning of July (1 July to 4 July) tests are being carried out to analyse how long the nodes remain in an overloaded state. For this purpose, a certain amount of workloads are being executed 3 times, during a period of 10 minutes (600 seconds) each time, on each of the nodes. For each period the loads will be different to simulate different scenarios that the aerOS IEs will have to face. Workloads will be combined with the execution of different services, such as server backend, databases, basic aerOS services, etc. During each period the self-awareness will measure the CPU and RAM usage, counting the amount of seconds that the node remains in an overloaded state through a test script in Python. Finally, the results obtained (the 3 periods of each node) will be averaged to obtain a realistic measurement of the average overload time of a node in the aerOS computing continuum. The tests will be performed on three different sets of IEs, in order to cover the widest possible heterogeneity of architectures and technical specifications. The first set consists of 4 CloudFerro cluster 2 nodes. Two of them (nodes 0 and 1) have a 4-core CPU and AMD64 architecture, 16 GB of RAM and run Fedora 35. The other two (nodes 7 and 8) have a 2-core CPU, AMD64 architecture, 8 GB of RAM and run Fedora 35. The second set consists of 4 nodes from domain 3 of the UPV “continuum”. This set consists of 2 virtual machines that have a 1-core CPU and AMD64 architecture, 8 GB of RAM and run Ubuntu 24.04 LTS. In addition, it also consists of a Raspberry Pi 3 Model B+ and a Raspberry Pi 4 Model B. The RPi 3 runs Raspbian 12 and the RPi 4 runs Ubuntu 22.04 LTS. The last set consists of a Pilot 1.1 test virtual machine. This virtual machine has a 2-core CPU, AMD64 architecture, 12 GB of RAM and runs Ubuntu 24.04 LTS. After running the tests on all the nodes described above, the following results (in seconds) were obtained: • CloudFerro cluster 2: o Node 0: 84, 218 and 119. Mean = 140 (23 %). o Node 1: 108, 295 and 67. Mean = 157 (26 %). o Node 7: 104, 242 and 142. Mean = 163 (27 %). o Node 8: 167, 230 and 157. Mean = 185 (31 %). • Domain 3 of the UPV “continuum”: o VM 1: 150, 162 and 48. Mean = 120 (20 %). o VM 2: 145, 276 and 208. Mean = 210 (35 %). o RPi 3: 379, 274 and 411. Mean = 355 (59 %). o RPi 4: 91, 168 and 103. Mean = 121 (21 %). • Pilot 1.1: o VM 1: 133, 225 and 155. Mean = 171 (28 %). The results indicate that the overload time of a node is between 20 % and 59 % of the execution time (on average). However, these values are highly dependent on both the power-related workloads assigned and the performance of the node itself. A lower-performing node (such as the RasPi 3) will overload more easily and for longer periods of time, but more powerful cloud nodes will have relatively short overload times. Taking into account the heterogeneity of aerOS compute nodes, the average overload time of a node is 180 seconds, equivalent to 30 % of the total running time of a node. As can be seen, both the efficiency improvements introduced in the components that form aerOS and the use of the self-orchestrator or self-optimisation and adaptation make it possible to reduce the overload time of a node, allowing it to operate longer with lower workloads. This not only avoids overload, but also allows the node to be available for a greater amount of time.

KPI 1.5.2 Number of different topologies and hardware/software combinations of IEs supported

Table 55: KPI 1.5.2 Number of different topologies and hardware/software combinations of IEs supported

<table>
<tr><td>KPI ID number and partner resp.</td><td>KPI 1.5.2</td></tr>
<tr><td>KPI Name</td><td>Number of different topologies and hardware/software combinations of IEs supported</td></tr>
<tr><td>Description</td><td>Indicates the number of different nodes on which self-* capabilities are capable of running depending on the type of IE, its operating system, hardware architecture or performance</td></tr>
<tr><td>Motivation</td><td>The aerOS computing continuum is composed by a multitude of domains, which belong to different organizations and companies. Each domain has different topologies and hosts very heterogeneous IEs. This variety in the nodes is due to differences in the hardware and software of each node. From variations in processor architectures to variations in specs, performance, operating system or task execution capabilities and more. Increasing the number of different types of supported nodes by aerOS will allow for greater heterogeneity</td></tr>
<tr><td>Target value</td><td>10 different topologies and hardware/software combinations</td></tr>
<tr><td>Prerequisites</td><td>Basic set of self-* capabilities (self-awareness, self-orchestrator, self-optimisation and adaptation and self-API) running on one IE of study</td></tr>
<tr><td>aerOS components (task)</td><td>Self-awareness, self-orchestrator, self-optimisation and adaptation and self-API modules (T3.5)</td></tr>
<tr><td>Evaluation means</td><td>The most representative combinations of future aerOS nodes forming the computing continuum will be selected by combining different architectures, operating systems, available resources and execution environments to test the flexibility of deployment of the set of self-* capabilities. At least the basic self-* capabilities will be installed in each selected combination, and all functional combinations will be counted. To determine the heterogeneity of hardware-software combinations that will be able to support the set of self-* capabilities, the following combinations will be attempted:
<table>
<tr><th>Power</th><th>Node</th><th>Platform</th><th>Containerisation</th><th>OS</th></tr>
<tr><td>High-powered</td><td>Physical (SBC / laptop)</td><td>AMD64</td><td>Kubernetes (K3s…)</td><td>GNU/Linux (distro)</td></tr>
<tr><td>High-powered</td><td>Physical</td><td>AMD64</td><td>Docker</td><td>GNU/Linux</td></tr>
<tr><td>Low-powered</td><td>Physical</td><td>AMD64</td><td>Kubernetes</td><td>GNU/Linux</td></tr>
<tr><td>Low-powered</td><td>Physical</td><td>AMD64</td><td>Docker</td><td>GNU/Linux</td></tr>
</table>
</td></tr>
</table>

Low-powered	Physical	ARM64	Kubernetes	GNU/Linux
Low-powered	Physical	ARM64	Docker	GNU/Linux
High-powered	Virtual	AMD64	Kubernetes	GNU/Linux
High-powered	Virtual	AMD64	Docker	GNU/Linux
Low-powered	Virtual	AMD64	Kubernetes	GNU/Linux
Low-powered	Virtual	AMD64	Docker	GNU/Linux
Low-powered	Virtual	ARM64	Kubernetes	GNU/Linux
Low-powered	Virtual	ARM64	Docker	GNU/Linux

Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***3***	***4 different topologies and hardware/software combinations***	***10 different topologies and hardware/software combinations***

Outcome elaboration (M38)

The aerOS computing continuum is formed by different domains that can belong to different stakeholders. Among these continuous are, for example, the one used for the MVP (formed by the domains of the NCSRD and CF partners), the continuum of each of the Pilots or the testing continuum of the UPV partner (formed by three domains). Each continuum has different topologies and hosts very heterogeneous. This variety in the nodes is due to differences in the hardware and software of each node. From variations in processor architectures to variations in specs, performance, operating system or task execution capabilities and more.

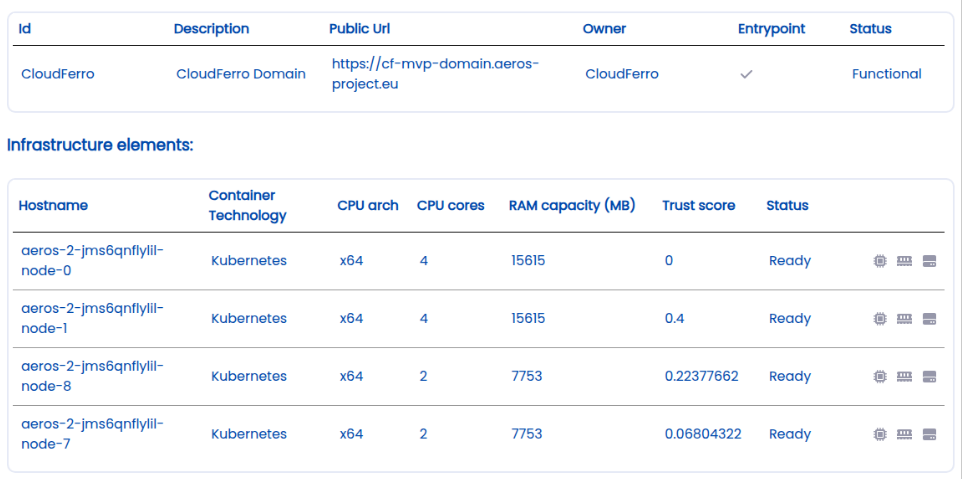
To determine the heterogeneity of hardware-software combinations that will be able to support the set of self-* capabilities, the following combinations have been tested:

Power	Node	Platform	Containerisation	OS
High-powered	Physical	AMD64	Kubernetes	GNU/Linux (Ubuntu 22.04)
High-powered	Physical	AMD64	Docker	GNU/Linux (Ubuntu 22.04)
Low-powered	Physical (laptop)	AMD64	Kubernetes (K3s)	GNU/Linux (Ubuntu 22.04)

Low-powered	Physical (PC)	AMD64	Docker	GNU/Linux (Ubuntu 24.04)
Low-powered	Physical (SBC)	ARM64	Kubernetes (K3s)	GNU/Linux (Raspbian 12 & Ubuntu 22.04)
Low-powered	Physical (SBC)	ARM64	Docker	GNU/Linux (Ubuntu 22.04)
High-powered	Virtual (cloud provider)	AMD64	Kubernetes (K8s)	GNU/Linux (Ubuntu 22.04 & Fedora 35)
High-powered	Virtual	AMD64	Docker	GNU/Linux (Ubuntu 22.04)
Low-powered	Virtual	AMD64	Kubernetes	GNU/Linux (Ubuntu 22.04)
Low-powered	Virtual	AMD64	Docker	GNU/Linux (Ubuntu 22.04)

The last two hardware/software combinations (virtual machines on ARM64 architecture) have not been tested. This is because they are not usual combinations that will be used in aerOS, since all the ARM64 devices that will be used will be physical. Furthermore, they do not make sense in any current aerOS scenario (pilots, open calls, etc.) and are very unlikely to appear in future cases. However, the target of 10 combinations has been achieved.

The following screenshots show the multiple hardware/software combinations reflected in the different Management Portals of the aerOS domains:

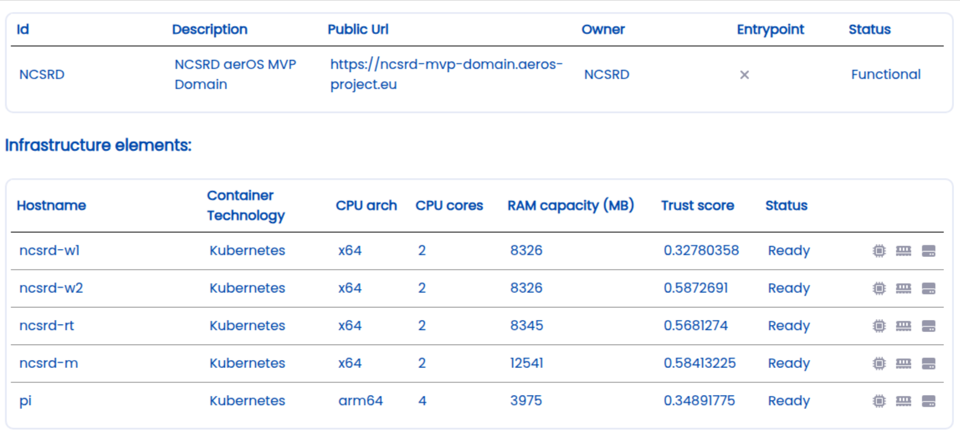


Id	Description	Public Url	Owner	Entrypoint	Status
CloudFerro	CloudFerro Domain	https://cf-mvp-domain.aeros-project.eu	CloudFerro	✓	Functional

Infrastructure elements:

Hostname	Container Technology	CPU arch	CPU cores	RAM capacity (MB)	Trust score	Status
aeros-2-jms6qnflylil-node-0	Kubernetes	x64	4	15615	0	Ready
aeros-2-jms6qnflylil-node-1	Kubernetes	x64	4	15615	0.4	Ready
aeros-2-jms6qnflylil-node-8	Kubernetes	x64	2	7753	0.22377662	Ready
aeros-2-jms6qnflylil-node-7	Kubernetes	x64	2	7753	0.06804322	Ready

The image above corresponds to the domain of the partner CloudFerro. The following image shows the NCSRD partner domain. Both belong to the MVP continuum:

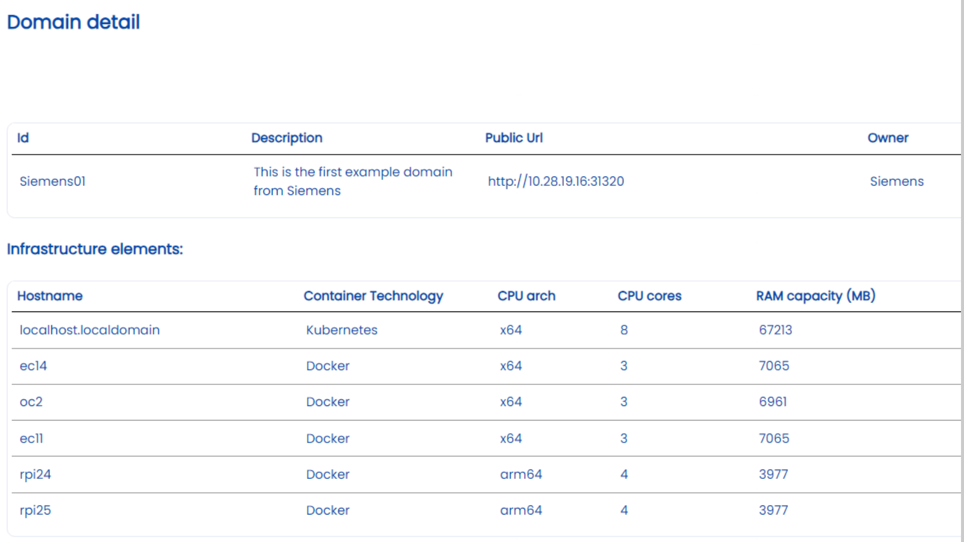


Id	Description	Public Url	Owner	Entrypoint	Status
NCSRD	NCSRD aerOS MVP Domain	https://ncsrd-mvp-domain.aeros-project.eu	NCSRD	×	Functional

Infrastructure elements:

Hostname	Container Technology	CPU arch	CPU cores	RAM capacity (MB)	Trust score	Status
ncsrd-w1	Kubernetes	x64	2	8326	0.32780358	Ready
ncsrd-w2	Kubernetes	x64	2	8326	0.5872691	Ready
ncsrd-rt	Kubernetes	x64	2	8345	0.5681274	Ready
ncsrd-m	Kubernetes	x64	2	12541	0.58413225	Ready
pi	Kubernetes	arm64	4	3975	0.34891775	Ready

The following image shows the SIEMENS partner domain of Pilot 1.3:



Domain detail

Id	Description	Public Url	Owner
Siemens01	This is the first example domain from Siemens	http://10.28.19.16:31320	Siemens

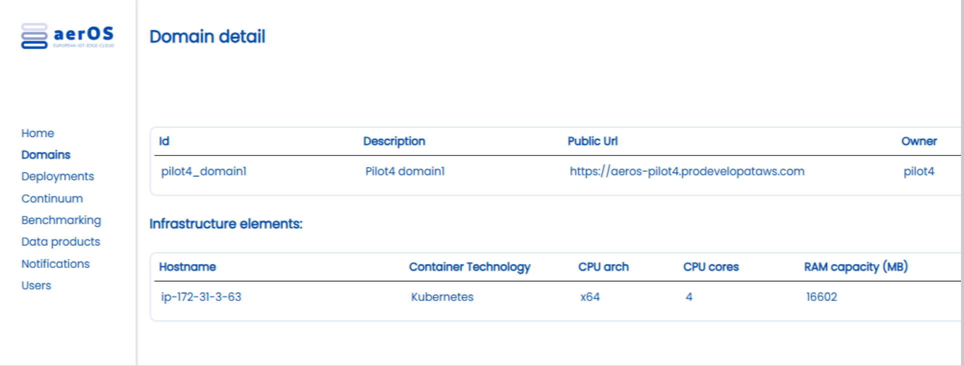
Infrastructure elements:

Hostname	Container Technology	CPU arch	CPU cores	RAM capacity (MB)
localhost.localdomain	Kubernetes	x64	8	67213
ec14	Docker	x64	3	7065
oc2	Docker	x64	3	6961
ec11	Docker	x64	3	7065
rpi24	Docker	arm64	4	3977
rpi25	Docker	arm64	4	3977

The following screenshots show the different domains of Pilot 4 and the IEs of each:

Domains list

Id	Description	Public Url	Owner	Entrypoint	Status
pilot4_domain1	Pilot4 domain1	https://aeros-pilot4.prodevelopataws.com	pilot4	✓	Preliminary
eurogate_limassol	EUROGATE CTL domain	https://mqtt-kubernetes.eurogate-limassol.net	pilot4	×	Functional
cutdomain	This is a new domain, which is not the entrypoint	https://cutdom.aeros.cut.ac.cy	CUT	×	Preliminary

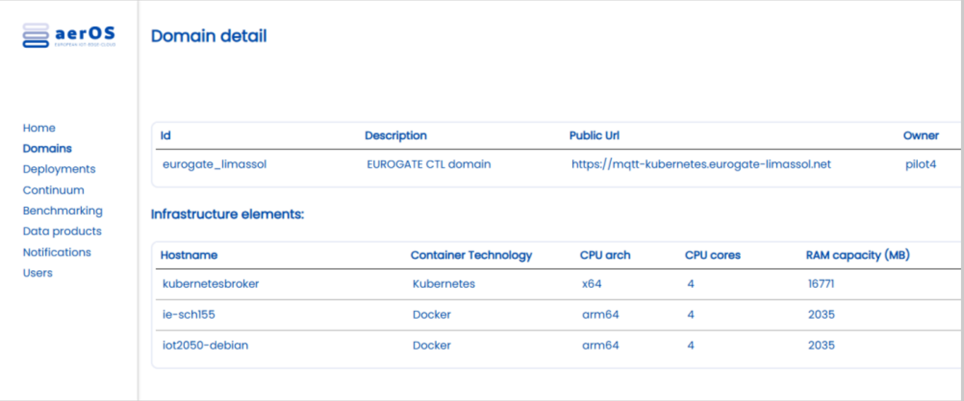


Domain detail

Id	Description	Public Url	Owner
pilot4_domain1	Pilot4 domain1	https://aeros-pilot4.prodevelopataws.com	pilot4

Infrastructure elements:

Hostname	Container Technology	CPU arch	CPU cores	RAM capacity (MB)
ip-172-31-3-63	Kubernetes	x64	4	16602

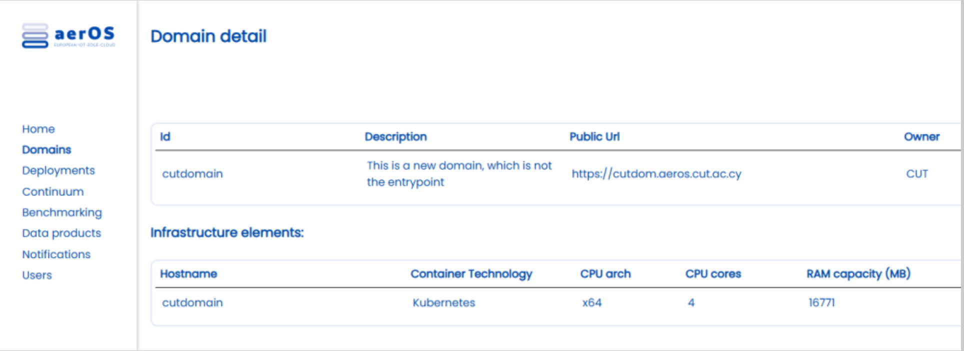


Domain detail

Id	Description	Public Url	Owner
eurogate_limassol	EUROGATE CTL domain	https://mqtt-kubernetes.eurogate-limassol.net	pilot4

Infrastructure elements:

Hostname	Container Technology	CPU arch	CPU cores	RAM capacity (MB)
kubernetesbroker	Kubernetes	x64	4	16771
ie-sch155	Docker	arm64	4	2035
iot2050-debian	Docker	arm64	4	2035



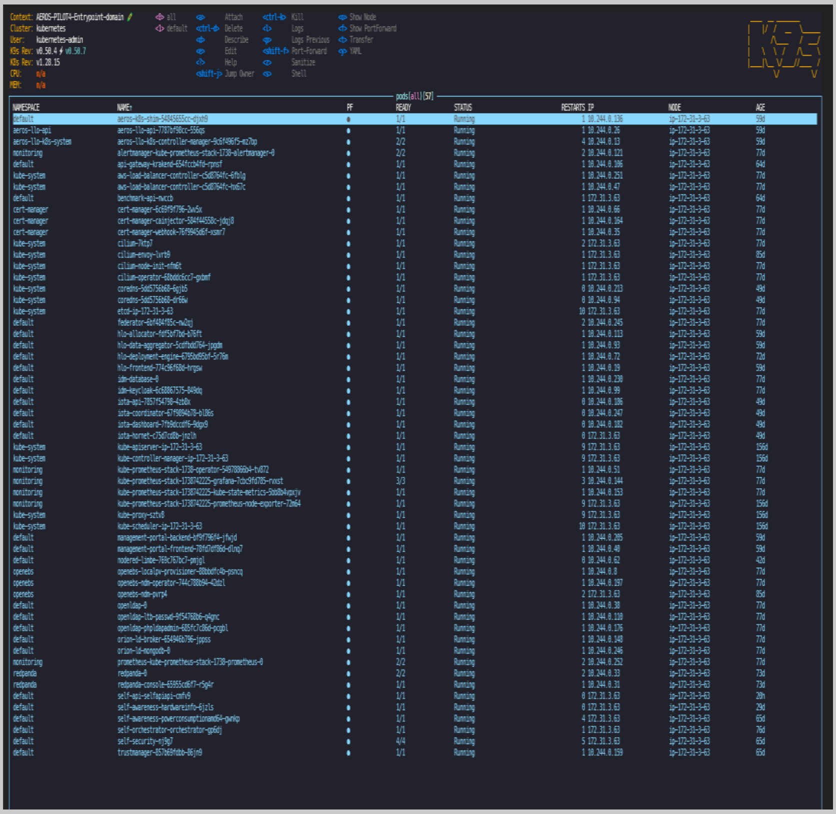
Domain detail

Id	Description	Public Url	Owner
cutdomain	This is a new domain, which is not the entrypoint	https://cutdomain.aeros.cut.ac.cy	CUT

Infrastructure elements:

Hostname	Container Technology	CPU arch	CPU cores	RAM capacity (MB)
cutdomain	Kubernetes	x64	4	16771

The first domain of Pilot 4 is a cloudAWS IE on kubeadm, domain 2 is a mix of k3s + Docker (5 IEs on k3s + 2 IEs on Docker) and domain 3 is an IE on CUT with kubeadm. The following image shows the execution of the self-* modules in the Pilot 4 infrastructure:



```
Context: AEROS-PILOT4-Entrypoint-domain       <0> all       <a>       Attach    <ctrl-k>  Kill           <o> Show Node
Cluster: kubernetes                           <1> default   <ctrl-d>  Delete    <l>       Logs           <f> Show PortForward
User:    kubernetes-admin                                   <d>       Describe  <p>       Logs Previous  <t> Transfer
K9s Rev: v0.50.4 ⚡ v0.50.7                                  <e>       Edit      <shift-f> Port-Forward   <y> YAML
K8s Rev: v1.28.15                                           <?>       Help      <z>       Sanitize
CPU:     n/a                                                <shift-j> Jump Owner <s>      Shell
MEM:     n/a

                                                        pods(all)[57]
NAMESPACE             NAME↑                                                       PF  READY  STATUS   RESTARTS IP            NODE             AGE
default               aeros-k8s-shim-54845655cc-djxh9                             ●   1/1    Running         1 10.244.0.136  ip-172-31-3-63   59d
aeros-llo-api         aeros-llo-api-7787bf98cc-556qs                              ●   1/1    Running         1 10.244.0.26   ip-172-31-3-63   59d
aeros-llo-k8s-system  aeros-llo-k8s-controller-manager-9c6f496f5-mz7bp            ●   2/2    Running         4 10.244.0.13   ip-172-31-3-63   59d
monitoring            alertmanager-kube-prometheus-stack-1738-alertmanager-0      ●   2/2    Running         2 10.244.0.121  ip-172-31-3-63   77d
default               api-gateway-krakend-654fccb4fd-rpnsf                        ●   1/1    Running         1 10.244.0.106  ip-172-31-3-63   64d
kube-system           aws-load-balancer-controller-c5d8764fc-6fblg                ●   1/1    Running         1 10.244.0.251  ip-172-31-3-63   77d
kube-system           aws-load-balancer-controller-c5d8764fc-hx67c                ●   1/1    Running         1 10.244.0.47   ip-172-31-3-63   77d
default               benchmark-api-nwccb                                         ●   1/1    Running         1 172.31.3.63   ip-172-31-3-63   64d
cert-manager          cert-manager-6c69f9f796-2wv5x                               ●   1/1    Running         1 10.244.0.66   ip-172-31-3-63   77d
cert-manager          cert-manager-cainjector-584f44558c-jdqj8                    ●   1/1    Running         1 10.244.0.164  ip-172-31-3-63   77d
cert-manager          cert-manager-webhook-76f9945d6f-xsmr7                       ●   1/1    Running         1 10.244.0.35   ip-172-31-3-63   77d
kube-system           cilium-7ktp7                                                ●   1/1    Running         2 172.31.3.63   ip-172-31-3-63   77d
kube-system           cilium-envoy-lvrb9                                          ●   1/1    Running         1 172.31.3.63   ip-172-31-3-63   85d
kube-system           cilium-node-init-nfm6t                                      ●   1/1    Running         1 172.31.3.63   ip-172-31-3-63   77d
kube-system           cilium-operator-68bddc6cc7-gxbmf                            ●   1/1    Running         1 172.31.3.63   ip-172-31-3-63   77d
kube-system           coredns-5dd5756b68-6gjb5                                    ●   1/1    Running         0 10.244.0.213  ip-172-31-3-63   49d
kube-system           coredns-5dd5756b68-dr66w                                    ●   1/1    Running         0 10.244.0.94   ip-172-31-3-63   49d
kube-system           etcd-ip-172-31-3-63                                         ●   1/1    Running        10 172.31.3.63   ip-172-31-3-63   77d
default               federator-6bf484f85c-nw2qj                                  ●   1/1    Running         2 10.244.0.245  ip-172-31-3-63   77d
default               hlo-allocator-fdf5bf7bd-b76ft                               ●   1/1    Running         1 10.244.0.113  ip-172-31-3-63   59d
default               hlo-data-aggregator-5cdfbdd764-jpgdm                        ●   1/1    Running         1 10.244.0.93   ip-172-31-3-63   59d
default               hlo-deployment-engine-6795bd95bf-5r76m                      ●   1/1    Running         1 10.244.0.72   ip-172-31-3-63   72d
default               hlo-frontend-774c96f68d-hrgsw                               ●   1/1    Running         1 10.244.0.19   ip-172-31-3-63   59d
default               idm-database-0                                              ●   1/1    Running         1 10.244.0.230  ip-172-31-3-63   77d
default               idm-keycloak-6c68867575-849dq                               ●   1/1    Running         1 10.244.0.99   ip-172-31-3-63   77d
default               iota-api-7857f54798-4zb8x                                   ●   1/1    Running         0 10.244.0.106  ip-172-31-3-63   49d
default               iota-coordinator-67f9894b78-bl86s                           ●   1/1    Running         0 10.244.0.247  ip-172-31-3-63   49d
default               iota-dashboard-7fb9dccdf6-9dgx9                             ●   1/1    Running         0 10.244.0.182  ip-172-31-3-63   49d
default               iota-hornet-c75d7cd8b-jnzlh                                 ●   1/1    Running         0 172.31.3.63   ip-172-31-3-63   49d
kube-system           kube-apiserver-ip-172-31-3-63                               ●   1/1    Running         9 172.31.3.63   ip-172-31-3-63   156d
kube-system           kube-controller-manager-ip-172-31-3-63                      ●   1/1    Running         9 172.31.3.63   ip-172-31-3-63   156d
monitoring            kube-prometheus-stack-1738-operator-549788866b4-tv872       ●   1/1    Running         1 10.244.0.51   ip-172-31-3-63   77d
monitoring            kube-prometheus-stack-1738742225-grafana-7cbc9fd785-rvxst   ●   3/3    Running         3 10.244.0.144  ip-172-31-3-63   77d
monitoring            kube-prometheus-stack-1738742225-kube-state-metrics-5bb8b4vpxjv ● 1/1  Running         1 10.244.0.153  ip-172-31-3-63   77d
monitoring            kube-prometheus-stack-1738742225-prometheus-node-exporter-72m64 ● 1/1  Running         9 172.31.3.63   ip-172-31-3-63   156d
kube-system           kube-proxy-sztv8                                            ●   1/1    Running         9 172.31.3.63   ip-172-31-3-63   156d
kube-system           kube-scheduler-ip-172-31-3-63                               ●   1/1    Running        10 172.31.3.63   ip-172-31-3-63   156d
default               management-portal-backend-bf9f796f4-jfwjd                   ●   1/1    Running         1 10.244.0.205  ip-172-31-3-63   59d
default               management-portal-frontend-78fd7df86d-dlnq7                 ●   1/1    Running         1 10.244.0.40   ip-172-31-3-63   59d
default               nodered-limbe-769c767bc7-pmjgl                              ●   1/1    Running         0 10.244.0.62   ip-172-31-3-63   42d
openebs               openebs-localpv-provisioner-88bbdfc4b-psncq                 ●   1/1    Running         1 10.244.0.8    ip-172-31-3-63   77d
openebs               openebs-ndm-operator-744c788b94-42dzl                       ●   1/1    Running         1 10.244.0.197  ip-172-31-3-63   77d
openebs               openebs-ndm-pvrp4                                           ●   1/1    Running         2 172.31.3.63   ip-172-31-3-63   85d
default               openldap-0                                                  ●   1/1    Running         1 10.244.0.38   ip-172-31-3-63   77d
default               openldap-ltb-passwd-9f54768b6-q4gnc                         ●   1/1    Running         1 10.244.0.110  ip-172-31-3-63   77d
default               openldap-phpldapadmin-685fc7c86d-pcgbl                      ●   1/1    Running         1 10.244.0.176  ip-172-31-3-63   77d
default               orion-ld-broker-654946b796-jppss                            ●   1/1    Running         1 10.244.0.148  ip-172-31-3-63   77d
default               orion-ld-mongodb-0                                          ●   1/1    Running         1 10.244.0.246  ip-172-31-3-63   77d
monitoring            prometheus-kube-prometheus-stack-1738-prometheus-0          ●   2/2    Running         2 10.244.0.252  ip-172-31-3-63   77d
redpanda              redpanda-0                                                  ●   2/2    Running         2 10.244.0.33   ip-172-31-3-63   73d
redpanda              redpanda-console-65955cd6f7-r5g4r                           ●   1/1    Running         1 10.244.0.31   ip-172-31-3-63   73d
default               self-api-selfapiapi-cmfv9                                   ●   1/1    Running         0 172.31.3.63   ip-172-31-3-63   20h
default               self-awareness-hardwareinfo-6jzls                           ●   1/1    Running         0 172.31.3.63   ip-172-31-3-63   29d
default               self-awareness-powerconsumptionamd64-gwnkp                  ●   1/1    Running         4 172.31.3.63   ip-172-31-3-63   65d
default               self-orchestrator-orchestrator-gp6dj                        ●   1/1    Running         1 172.31.3.63   ip-172-31-3-63   76d
default               self-security-nj9g7                                         ●   4/4    Running         5 172.31.3.63   ip-172-31-3-63   65d
default               trustmanager-857b69fdbb-86jn9                               ●   1/1    Running         1 10.244.0.159  ip-172-31-3-63   65d
```

KPI 1.5.3 Number of metrics monitored from IEs

Table 56: KPI 1.5.3 Number of metrics monitored from IEs

<table>
<tr><th>KPI ID number and partner resp.</th><th colspan="3">KPI 1.5.3</th></tr>
<tr><td>KPI Name</td><td colspan="3">Number of metrics monitored from IEs</td></tr>
<tr><td>Description</td><td colspan="3">The amount of different information that the self-* modules are able to obtain on the characteristics, specifications, current performance and health status of the nodes where they run</td></tr>
<tr><td>Motivation</td><td colspan="3">The more metrics obtained, the more accurate will be the health indices of the nodes of aerOS</td></tr>
<tr><td>Target value</td><td colspan="3">15 attributes</td></tr>
<tr><td>Prerequisites</td><td colspan="3">Self-awareness module running on one IE of study</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">Self-awareness module (T3.5)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">Each self-awareness submodule can measure a certain amount of data extracted from the node where it is running. By analysing the specifications of two modules (hardware metrics and energy consumption) it is possible to know how much information and metrics they can extract from each IE. The hardware info sub-module is able to obtain information about the CPU (number of cores, max. frequency, architecture and current usage), the RAM memory (total amount, available and current usage), the storage (type, total amount, available and current usage), the network (speed up, speed down, traffic up, traffic down and lost packages) the operating system, the hostname, the internal IP address, the MAC address of the IE, theLow-Level Orchestrator, the infrastructure element tier or if it is able to run real-time services. The energy consumption submodule is able to obtain information on the current and average energy consumption of the node</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>20 attributes</i></td><td><i>28 attributes</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">At the beginning of July (1 July to 4 July) tests will be carried out to determine the amount of metrics and information that the self-awareness sub-modules are capable of obtaining. To do this, it will be installed on a node (regardless of its characteristics) and all the metrics and information will be obtained from the IE. Subsequently, it will be compared with the specification of the module and it will be verified that it is indeed capable of obtaining all the requested information. Finally, the amount of information and metrics obtained will be counted.
In order to obtain the values generated by the self-awareness, a query is realised by the name of the IE entity to the Orion-LD Context Broker of the corresponding domain, obtaining the following result:</td></tr>
</table>

```
{
  "id": "urn:ngsi-ld:InfrastructureElement:Cluster:1284319d9e1d",
  "type": "InfrastructureElement",
  "domain": "urn:ngsi-ld:Domain:Cluster",
  "hostname": "continuum-cluster-master",
  "containerTechnology": "Kubernetes",
  "internalIpAddress": "192.168.250.236",
  "macAddress": "12:84:31:9d:9e:1d",
  "lowLevelOrchestrator": "urn:ngsi-ld:LowLevelOrchestrator:Cluster:Kubernetes",
  "cpuCores": 4,
  “cpuFreqMax”: 2100,
  "currentCpuUsage": 13,
  “gpu”: false,
  “gpuMemory”: -1,
  "ramCapacity": 33649,
  "availableRam": 30119,
  "currentRamUsage": 3530,
  "currentRamUsagePct": 11,
  “diskType”: “HDD”,
  "diskCapacity": 82111,
  "availableDisk": 22684,
  "currentDiskUsage": 55642,
  "currentDiskUsagePct": 71,
  “netSpeedUp”: 163,
  “netSpeedDown”: 45,
  “netTrafficUp”: 0.18,
  “netTrafficDown”: 0.01,
  “netLostPackages”: 0,
  "avgPowerConsumption": 9,
  "currentPowerConsumption": 5,
  “powerSource”: “urn:ngsi-ld:none”,
  “energyEfficiencyRatio”: -1,
  "realTimeCapable": false,
  "trustScore": -1,
  "cpuArchitecture": "urn:ngsi-ld:CpuArchitecture:x64",
  "operatingSystem": "urn:ngsi-ld:OperatingSystem:Linux",
  "infrastructureElementTier": "urn:ngsi-ld:InfrastructureElementTier:Cloud",
  "infrastructureElementStatus": "urn:ngsi-ld:InfrastructureElementStatus:Ready",
  "location": [0, 0]
}
```

Of all these values, the following are obtained by self-awareness:

- domain.

	• hostname. • internalIpAddress. • macAddress. • lowLevelOrchestrator. • cpuCores. • cpuFreqMax. • currentCpuUsage. • ramCapacity. • availableRam. • currentRamUsage. • currentRamUsagePct. • diskType. • diskCapacity. • availableDisk. • currentDiskUsage. • currentDiskUsagePct. • netSpeedUp. • netSpeedDown. • netTrafficUp. • netTrafficDown. • netLostPackages. • avgPowerConsumption. • currentPowerConsumption. • realTimeCapable. • cpuArchitecture. • operatingSystem. • infrastructureElementTier. For a total of 28 attributes. As can be seen, the number of attributes that self-awareness is capable of obtaining has increased by 40 %, from 20 to 28 attributes.

KPI 1.5.4 Number of avoided service downgrade experience cases

Table 57: KPI 1.5.4 Number of avoided service downgrade experience cases

KPI ID number and partner resp.	**KPI 1.5.4**
KPI Name	**Number of avoided service downgrade experience cases**
Description	All the different types of scenarios in which a continuum IE is prevented from not being able to respond to the requests made to it, either to obtain information or to request it to execute a certain workload. In other words, the trait of aerOS of reacting in advance (e.g., reorchestrating services that were running in the IE, or deactivating from being eligible for new services, or horizontally scaling replicas) so that the IE is still functional and operative in the mid-term.
Motivation	Reducing the number of situations in which a IE in the continuum stops responding to requests for information or becomes inoperative increases overall user satisfaction and provides a better image of a robust, reliable and fault-tolerant system

Target value	5 demonstrable scenarios		
Prerequisites	Self-awareness and self-orchestration modules are functional. The KPI-1.5.5 has been demonstrated as VALID after measurements in M24, as the evaluation of KPI-1.5.4 depends to some extent of such KPI.		
aerOS components (task)	T3.3 HLO and LLOs, T3.5 self-awareness and self-orchestration.		
Evaluation means	The goal of the KPI is to demonstrate that aerOS can reduce the downgrade experience. In order to do that, the proposal is to make a comparison of a before/after (B/A) scenario: • First, the "Before" scenario is constructed. Here, the team aims at "profiling" which is the impact of several type of services in the continuum, enable to forecast at which point (and with which numbers, a IE would either collapse or downgrade the service experience). • Second, the "After" scenario will depart from the same type of services, and, after judiciously selecting specific thresholds, the self-orchestration mechanism will be put in place in the IEs of the continuum. There, by using the same evaluation means of KPI-1.5.5., the team will reflect whether these techniques avoiding the occurrence of the forecasted down situations. The methods used, and the assumptions taken, were: • Metrics: Trend forecasting of CPU usage metrics. • Assumption: Less current CPU usage per cores on a machine means it has more compute power available for the application. Therefore, this is a solid metric to consider when "degradation" is more likely to occur in a specific node (IE). • Methodology: To accurately determine if the task is going into a potential downgrade or is just experiencing a sudden spike in intensity, the task is monitored for a certain amount of time (2 min) before judging if there is a downgrade scenario or not. • After this time passes, a Neural Prophet regression model forecasts metrics for a certain amount of time in the future (1 min), and the predicted value is provided in the evaluation. This mechanism, in the runtime functioning of aerOS Meta-OS, would connect with the self-orchestration's reallocation trigger to offload to the HLO the picking of alternative best fit for the task.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***0 demonstrable scenarios (0%)***	***4 demonstrable scenarios (80%)***
Outcome elaboration (M38)	The scenarios were designed to test several models of user load, as well as set up to utilize different resilience mechanisms present in the aerOS self-* capabilities. Namely, we utilize self-orchestration and self-scaling as the services to prevent service downgrade. Self-orchestration evicts services at risk of degradation according to a rules-based system and allocates them in a different Infrastructure Element (IE) across the network.		

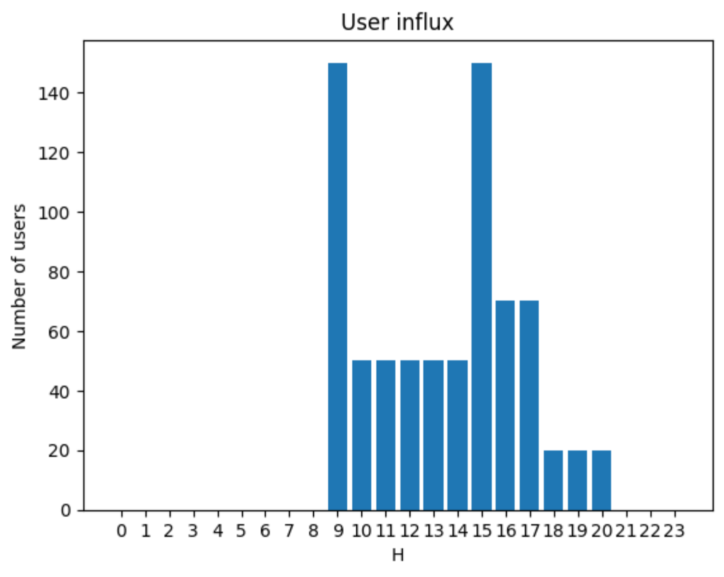
Self-scaling predicts resource usage across registered services and utilizes this prediction to anticipate high demand, scaling the number of replicas of a service.

For modelling, we utilize across all scenarios a sample service, in this case a job queue server, that is the service for which we want to avoid downgrade and simulate load with several cronjobs that are designed to request a variable workload of the server randomly, in predefined patterns intended to resemble real users. For example, some scenarios have a somewhat constant workload during the day, but experience a spike of usage at 9am, where the simulated 'workers' come online and start requesting jobs all at once. Simulated workers request jobs every minute across a fixed schedule, since they are modelled with cron jobs.

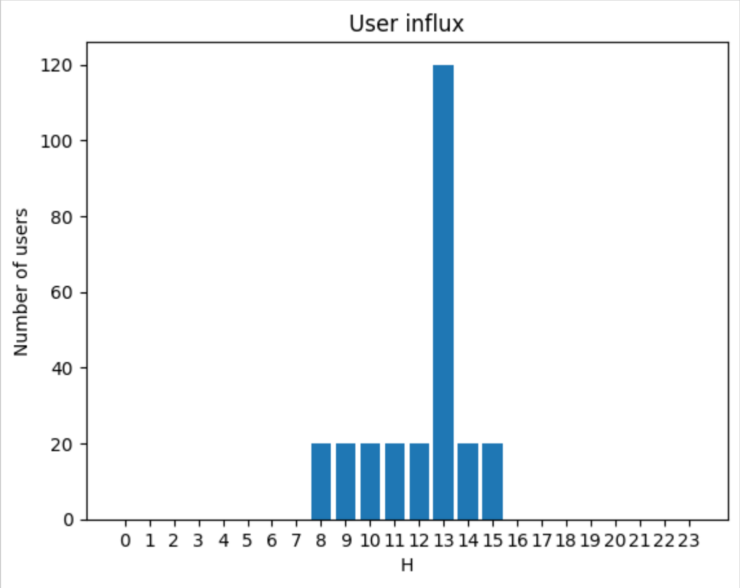
It is also important to note that while the user influx is completely predictable, the computer resources used are completely random, since the simulated job's difficulty varies randomly.

Two workload profiles are utilized across all scenarios: **high demand**, and **sudden spikes**.

High demand models work usage between 9am and 8pm. To resemble usual patterns for job servers, this model experiences highest demand at 9am and 3pm, where workers are supposed to come online and start making requests for the service. The workers are either 'morning' workers – active from 9am to 5pm – or evening workers, active from 3pm to 8pm. In the graph below illustrating activity across the day, we can see a high influx of users of each type at the 'start of their shift', as well as constant activity throughout it. Furthermore, there is an overlap between both shift types from 3pm to 5pm.



Sudden burst profiles are much simpler. In these cases, the activity is kept constant throughout the workday from 9am to 5pm and experiences a sudden spike at 1pm.



- **Self-orchestration with constant high demand**

For this scenario, we utilize the high demand work profile and deploy the job server in a predetermined IE, in which we also register a rule for self-orchestration indicating a redeployment if global CPU usage in the IE exceeds 75%. Since self-orchestration is a purely reactive system, we expect it to suddenly redeploy the service at the start of the work period to a different IE, after which the service will stay stable in its assigned IE. Finally, we expect another redeployment at the second peak of activity, when the evening users come online.

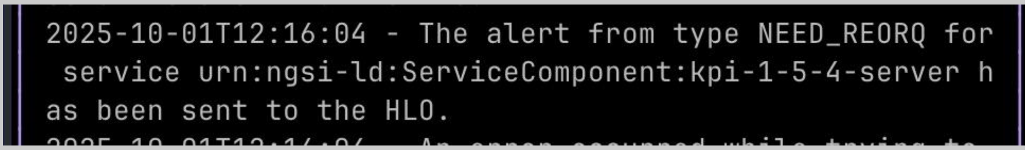
At the beginning of the user load, which starts with a peak, the service is immediately redeployed to a different IE in the pool. After the move, the service stays stable in the new IE, since all jobs come in at once during the peak and enough were processed before the reallocation such that the new IE is not overloaded by the remaining requests. The final reallocation takes place at the second peak, where the situation previously described repeats itself. After this last reallocation, the service remains stable until the end of the simulation period

```
2025-09-26T09:15:28 - The alert from type NEED_REORQ for service urn:ngsi-ld:ServiceComponent:kpi-1-5-4-server has been
 sent to the HLO.
```

- **Self-orchestration with sudden bursts in activity**

In this case, the sudden burst profile is used, and like in the previous scenario, we deploy the server in a single predetermined IE, registering a self-orchestration rule to redeploy if global CPU usage exceeds 75%. In this case, the expectation is for a single redeployment to take place, at the usage spike.

Indeed, this scenario is exactly replicated. The service stays stable in its assigned IE until the spike is reached. After reallocation, since we used a high threshold of CPU usage, enough requests were able to be processed such that there is no need for further redeployment after the move.



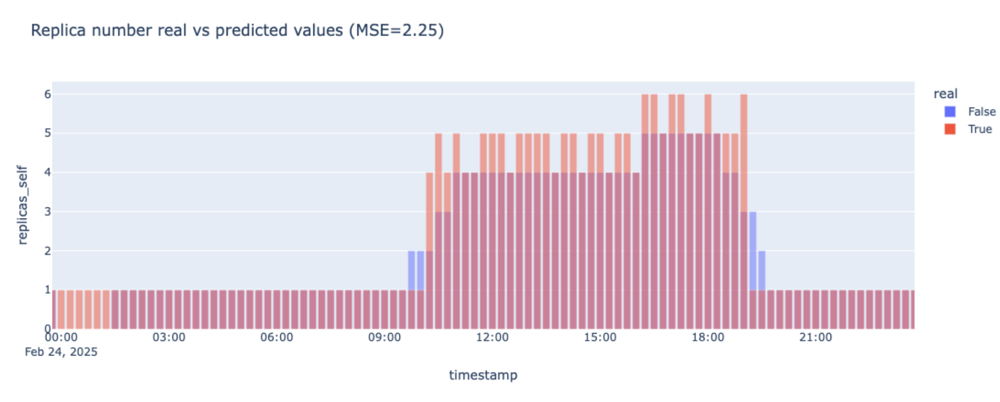
```
2025-10-01T12:16:04 - The alert from type NEED_REORQ for
 service urn:ngsi-ld:ServiceComponent:kpi-1-5-4-server h
as been sent to the HLO.
```

- **Self-scaling with constant high demand**

In this scenario we aim to demonstrate the predictive ability of self-scaling to scale service replicas and avoid degradation by increasing resources. For this, we deploy our test service in a predetermined IE and register the service with the self-scaling manager. Using the high demand profile, we expect the service not to need redeployment and instead be able to meet demand by spinning up service replicas.

Since self-scaling is a predictive system, we leave the service running for the period required by self-scaling, in this case 2 days. Self-scaling then adjusts the number of replicas based on its prediction every 15 minutes. Note that jobs are submitted every minute by the simulated users, meaning that since the workload is randomized there could be scenarios where this randomness results in a higher workload and therefore self-scaling underestimates the number of replicas.

After executing the system for 24h, the graph underneath illustrates the number of real replicas required by the service, in red, compared to the predicted number of replicas, in blue.



Remember that replicas are reevaluated every 15 minutes, meaning that if the resource usage dictates one more replica, even for a small amount of time in that 15min band, the ‘true’ or required number of replicas increases. Overall, we see that self-scaling does a good but not perfect job of avoiding service degradation, where the maximum error is of 2 replicas, which occurs in a single 15min slice and would constitute service degradation.

- **Self-orchestration with sudden burst among a pool of Infrastructure Elements**

This scenario is completely equivalent to scenario 2, except in this case we also intend to test integration with other aerOS systems, namely the semi-automatic deployment mode, where we can specify a set of IEs where a service can be allocated. The expectation then is the same as that scenario, where a reallocation is expected at the peak of the work profile. The difference is that there are only two IEs in the specified pool, and we expect redeployments to be between the two IEs. Given that there is expected to be a single redeployment in the profile, at the end of a cycle the service will be allocated in the only other IE in the pool. This exhibits a case where a continuum operator wishes to allocate services with more control than fully automatic deployments provide, for example to ensure critical services remain in on-premise nodes.

Replicating scenario 2, the service is immediately reallocated to the other IE in the semi-automatic deployment pool on the activity burst and stays stable until the end.

```
2025-10-06T12:01:02 - The alert from type NEED_REORQ for service urn:ngsi-ld:ServiceComponent:
kpi-1-5-4-server has been sent to the HLO.
```

KPI 1.5.5 % of reorchestration requests issued by decentralized IEs

Table 58: KPI 1.5.5 % of reorchestration requests issued by decentralized IE

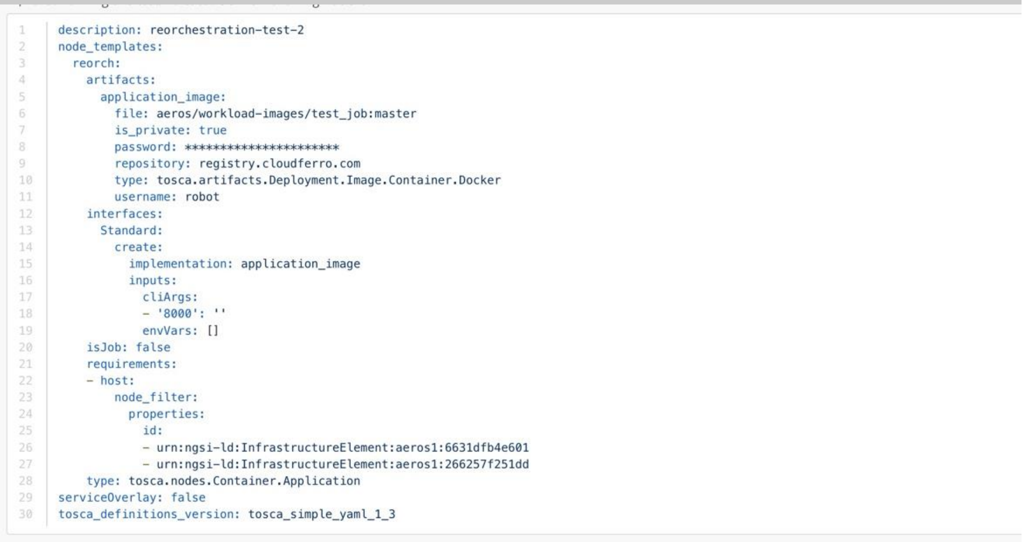
KPI ID number and partner resp.	**KPI 1.5.5**
KPI Name	**% of reorchestration requests issued by decentralized IEs**
Description	Number of requests coming from decentralised IEs in the computing continuum to the main aerOS reorchestration systems based on their current or future workload to avoid failures in running services or system overloads that may generate unwanted situations
Motivation	A number of decentralised reorchestration requests provide insight into the actual performance and processing capacity of the IEs in the computing continuum
Target value	25%
Prerequisites	Self-awareness, self-diagnose, self-realtimeness, self-optimisation and adaptation, and self-orchestrator modules running on the continuum of study. First, on a continuum formed by 3 domains (MVP), and, later, in the continuum corresponding to the selected pilot.
aerOS components (task)	Self-awareness, self-diagnose, self-realtimeness, self-optimisation and adaptation, and self-orchestrator modules (T3.5).
Evaluation means	The structure of measuring this KPI will be tackled in two different stages. - First (to be covered in D5.5), a simulated scenario will be created departing from one running continuum. There, some services will be run, and a situation will be artificially generated to demonstrate that the reorchestration functionality is operative and that it indeed supports the KPI-1.5.4 in which the overload of an IE is reduced thanks to compensation in another part of the continuum. - Second (to be covered in D5.6), a running scenario will be observed during a certain timeframe (1 month, closer to the final date of aerOS). This running scenario will exist within one out of the 5 pilots of the project. The specific dates, pilot and timing will be described later in D5.6. The evaluation means here can be decomposed in two different methods: - Continuous observation of the services that are running in a continuum. Reporting if a service was originally allocated to a specific IE and then it ends up running in another IE (and providing evidence). - Checking the IOTA registries. As it has been designed, every time that the self-orchestration request is triggered an IOTA message will be immutably registered through the DLT. Therefore, checking the IOTA registries and

	the total amount of services deployed in a certain timeframe, this percentage will be extracted.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)		***1/3 service component is successfully re-orchestrated (33%).***	***The service is successfully redeployed to a different InfrastructureElement following the IE's forced overload***

Outcome elaboration (M38)

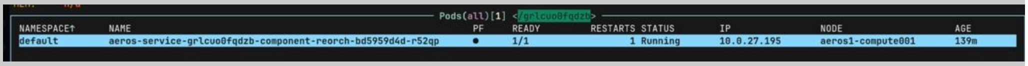
The test is carried out in the infrastructure from Pilot 2 Green Edge Processing. A processing service performing cloud mask computing for Earth Observation imaging is deployed. The service retrieves imagery from Sentinel-2, performs the computing at the edge nodes, and then saves results to a cloud-hosted bucket.

Below is the deployment specification for the processing service, as well as the pool of nodes it can be deployed in, by using semi-automatic deployment.



```
description: reorchestration-test-2
node_templates:
  reorch:
    artifacts:
      application_image:
        file: aeros/workload-images/test_job:master
        is_private: true
        password: **********************
        repository: registry.cloudferro.com
        type: tosca.artifacts.Deployment.Image.Container.Docker
        username: robot
    interfaces:
      Standard:
        create:
          implementation: application_image
          inputs:
            cliArgs:
            - '8000': ''
            envVars: []
    isJob: false
    requirements:
    - host:
        node_filter:
          properties:
            id:
            - urn:ngsi-ld:InfrastructureElement:aeros1:6631dfb4e601
            - urn:ngsi-ld:InfrastructureElement:aeros1:266257f251dd
    type: tosca.nodes.Container.Application
serviceOverlay: false
tosca_definitions_version: tosca_simple_yaml_1_3
```

The InfrastructureElement that has the service assigned by the HLO from among thepool is IE *6631d*, which correspond to node *aeros1-compute001*



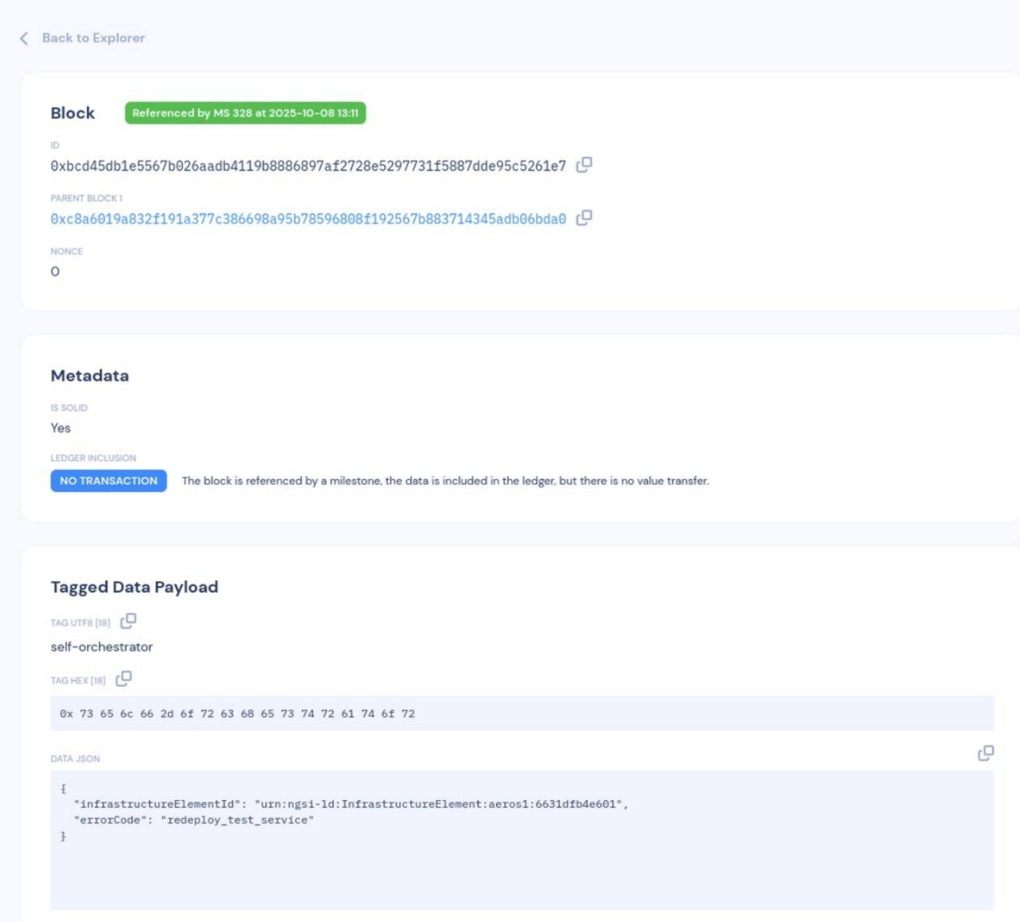
```
                                              Pods(all)[1] </grlcuo0fqdzb>
 NAMESPACE↑   NAME                                                      PF   READY        RESTARTS STATUS    IP            NODE               AGE
 default      aeros-service-grlcuo0fqdzb-component-reorch-bd5959d4d-r52qp  ●    1/1                 1 Running   10.0.27.195   aeros1-compute001  139m
```

This IE then has a rule registered with its self-orchestrator ordering a redeployment if resource usage exceeds a certain percentage. Additionally, the test service is intentionally modified to exceed normal usage and thus force degradation and the service's redeployment.

```
curl --location 'http://10.16.4.124:8001/rules' \
--header 'Content-Type: application/json' \
--data '{
    "name": "CF_PILOT_2_KPI_155",
    "conditions": {
        "any": [
            {
            "fact": "urn:ngsi-ld:InfrastructureElement:aeros1:6631dfb4e601:currentCpuUsage",
            "operator": "greaterThanInclusive",
            "value": 30
            }
        ]
    },
    "event": {
        "type": "NEED_REORQ",
        "params": {
            "message": "urn:ngsi-ld:Service:v8hpqdjamh2z:Component:reorch"
        }
    }
}
'
```

Self-orchestrator will then reallocate the processing job to the other IE in the pool once the rule triggers and registers a message through IOTA.

The IOTA message, tagged ‘self-orchestrator’, indicates the call to the associated EAT function to handle reallocation is made.



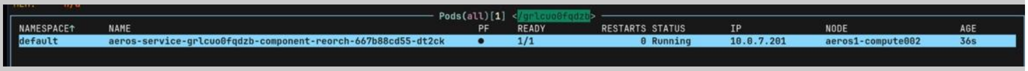
Block
Referenced by MS 328 at 2025-10-08 13:11
ID
0xbcd45db1e5567b026aadb4119b8886897af2728e5297731f5887dde95c5261e7
PARENT BLOCK 1
0xc8a6019a832f191a377c386698a95b78596808f192567b883714345adb06bda0
NONCE
0

Metadata
IS SOLID
Yes
LEDGER INCLUSION
NO TRANSACTION
The block is referenced by a milestone, the data is included in the ledger, but there is no value transfer.

Tagged Data Payload
TAG UTF8 [18]
self-orchestrator
TAG HEX [18]
0x 73 65 6c 66 2d 6f 72 63 68 65 73 74 72 61 74 6f 72
DATA JSON

```
{
  "infrastructureElementId": "urn:ngsi-ld:InfrastructureElement:aeros1:6631dfb4e601",
  "errorCode": "redeploy_test_service"
}
```

Finally, we can see the service reallocated to the other node in the semi-automatic pool, IE *22625*, which corresponds to the node *aeros1-compute002*

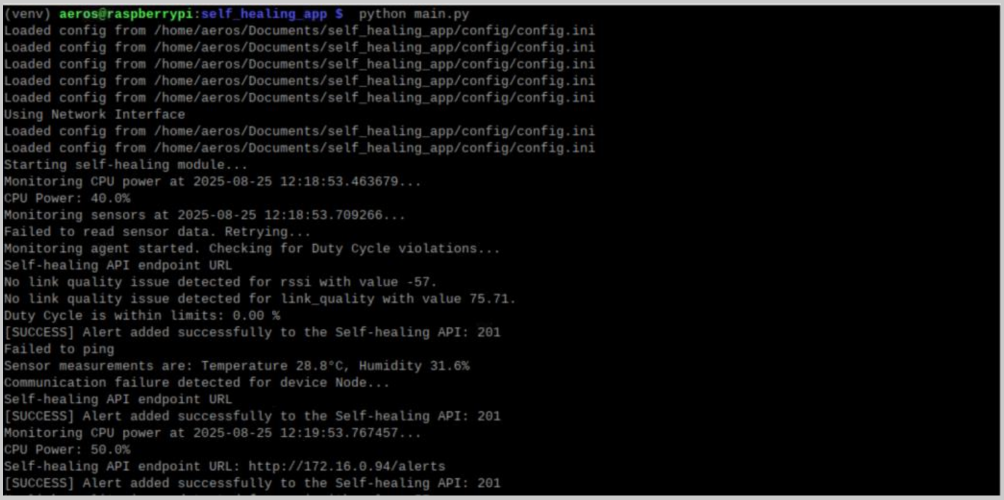


```
                                                          Pods(all)[1] </grlcuo0fqdzb>
NAMESPACE↑   NAME                                                    PF   READY    RESTARTS STATUS    IP           NODE                AGE
default      aeros-service-grlcuo0fqdzb-component-reorch-667b88cd55-dt2ck   ●    1/1             0 Running   10.0.7.201   aeros1-compute002   36s
```

KPI 1.5.6 # of IoT healing scenarios covered

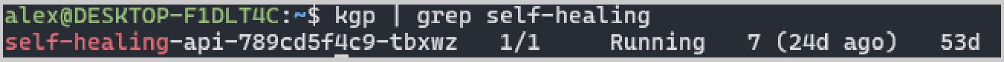
Table 59: KPI 1.5.6 # of IoT healing scenarios covered

KPI ID number and partner resp.	**KPI 1.5.6**		
KPI Name	**# of IoT healing scenarios covered**		
Description	The KPI aims at measuring the potential situations that the self-healing procedures can take effect.		
Motivation	Self-healing crystalizes the capability of autonomously recovering affected parts of the system both at the hardware and software level caused by failures or abnormal states. Self-healing can also restart the system to pre-established routines scheduling, if necessary.		
Target value	5		
Prerequisites	IE’s hardware setup is complete. Definition and implementation of scenarios are ready to be tested on IE hardware.		
aerOS components (task)	Self-healing (T3.5)		
Evaluation means	The importance of the self-healing functionality needs to be shown with specific scenarios. So far, the following “healing scenarios” have been identified: Sensor Failure, Device Power Alert, Network Protocol Violation, Link Quality Issues, Communication Failure Indication (no messages received by IE). In the first phase, tests ARE completed locally in FOGUS lab, running the defined scenarios on IE's hardware. In the second phase tests will be completed in the different Pilots of the project.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0 (no self-healing capabilities)***	***3/5 (60%)***	***5/5 (100%)***
Outcome elaboration (M38)	All five defined scenarios of the self-healing module have been fully implemented and validated under different conditions to analyse system behaviour and detect any possible failures or abnormal states.		

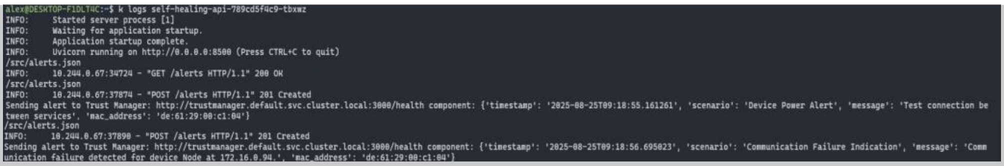


```
(venv) aeros@raspberrypi:self_healing_app $  python main.py
Loaded config from /home/aeros/Documents/self_healing_app/config/config.ini
Loaded config from /home/aeros/Documents/self_healing_app/config/config.ini
Loaded config from /home/aeros/Documents/self_healing_app/config/config.ini
Loaded config from /home/aeros/Documents/self_healing_app/config/config.ini
Loaded config from /home/aeros/Documents/self_healing_app/config/config.ini
Using Network Interface
Loaded config from /home/aeros/Documents/self_healing_app/config/config.ini
Loaded config from /home/aeros/Documents/self_healing_app/config/config.ini
Starting self-healing module...
Monitoring CPU power at 2025-08-25 12:18:53.463679...
CPU Power: 40.0%
Monitoring sensors at 2025-08-25 12:18:53.709266...
Failed to read sensor data. Retrying...
Monitoring agent started. Checking for Duty Cycle violations...
Self-healing API endpoint URL
No link quality issue detected for rssi with value -57.
No link quality issue detected for link_quality with value 75.71.
Duty Cycle is within limits: 0.00 %
[SUCCESS] Alert added successfully to the Self-healing API: 201
Failed to ping
Sensor measurements are: Temperature 28.8°C, Humidity 31.6%
Communication failure detected for device Node...
Self-healing API endpoint URL
[SUCCESS] Alert added successfully to the Self-healing API: 201
Monitoring CPU power at 2025-08-25 12:19:53.767457...
CPU Power: 50.0%
Self-healing API endpoint URL: http://172.16.0.94/alerts
[SUCCESS] Alert added successfully to the Self-healing API: 201
```

The module was successfully deployed in the Pilot 5 infrastructure, where its functionality was tested and confirmed.



```
alex@DESKTOP-F1DLT4C:~$ kgp | grep self-healing
self-healing-api-789cd5f4c9-tbxwz   1/1     Running   7 (24d ago)   53d
```



```
alex@DESKTOP-F1DLT4C:~$ k logs self-healing-api-789cd5f4c9-tbxwz
INFO:     Started server process [1]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO:     Uvicorn running on http://0.0.0.0:8500 (Press CTRL+C to quit)
/src/alerts.json
INFO:     10.244.0.67:34724 - "GET /alerts HTTP/1.1" 200 OK
/src/alerts.json
INFO:     10.244.0.67:37874 - "POST /alerts HTTP/1.1" 201 Created
Sending alert to Trust Manager: http://trustmanager.default.svc.cluster.local:3000/health component: {'timestamp': '2025-08-25T09:18:55.161261', 'scenario': 'Device Power Alert', 'message': 'Test connection be
tween services', 'mac_address': 'de:61:29:00:c1:04'}
/src/alerts.json
INFO:     10.244.0.67:37890 - "POST /alerts HTTP/1.1" 201 Created
Sending alert to Trust Manager: http://trustmanager.default.svc.cluster.local:3000/health component: {'timestamp': '2025-08-25T09:18:56.695023', 'scenario': 'Communication Failure Indication', 'message': 'Comm
unication failure detected for device Node at 172.16.0.94.', 'mac_address': 'de:61:29:00:c1:04'}
```

Any failure or abnormal state that is detected by the algorithm of the Self-healing module generates an alert message in a structured JSON format, as shown below. Those alert messages are available through the API that the Self-healing module exposes, and this endpoint is accessible from other IE components (e.g. Self-API).

```
{
  "alerts": [
    {
      "timestamp": "2025-08-25T09:18:55.161261",
      "scenario": "Device Power Alert",
      "message": "Test connection between services",
      "mac_address": "de:61:29:00:c1:04"
    },
    {
      "timestamp": "2025-08-25T09:18:56.695023",
      "scenario": "Communication Failure Indication",
      "message": "Communication failure detected for device Node at 172.16.0.94.",
      "mac_address": "de:61:29:00:c1:04"
    },
    {
      "timestamp": "2025-08-25T09:19:54.301221",
      "scenario": "Device Power Alert",
      "message": "Test connection between services",
      "mac_address": "de:61:29:00:c1:04"
    },
```

In addition, the Self-healing module also interacts with the Trust Manager component. It posts the related JSON alert to the Trust Manager API, when a failure or abnormal state is detected. These alerts contribute to the calculation algorithm of the Trust Score.

KPI 1.5.7 % of intrusion detected by the self-security

Table 60: KPI 1.5.7 % of intrusion detected by the self-security

KPI ID number and partner resp.	**KPI 1.5.7**
KPI Name	**% of intrusion detected by the self-security**
Description	Indicates the ability to detect cybersecurity intrusions that have been made to the IE.
Motivation	Measuring the percentage of intrusions that the self-security component has been able to detect allows the performance of the self-security component to be measured.
Target value	>90% intrusions
Prerequisites	Have the self-security component installed and running in IE
aerOS components (task)	Self-security (T3.5)

Evaluation means	In order to analyse this KPI, 3 different attacks will be launched on the IE where self-security is installed and the ability of this component to detect will be tested. With this, the detection rate of the attacks will be calculated. Currently, the component is configured to detect "port scanning", "denial of service (DoS)" attacks and “Brute Force” attacks. Currently, the component is configured to detect "port scanning", "denial of service (DoS)" and brute force attacks. By the selection of very modular tools in aerOS, this lis can be extended in the future as other needs arise.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***0 (0%)***	***14/14 (100%)***
Outcome elaboration (M38)	In order to test the component in the pilots that are using self-security, an script that is in charge of launching these three attacks has been generated:		

```
#!/bin/bash
# Check if exactly one argument (IP) is passed
if [ $# -ne 1 ]; then
echo "Usage: $0 <TARGET_IP>"
exit 1
fi


IP_TARGET=$1
echo "The target IP is: $IP_TARGET"


echo "Sending simulated HTTP POST requests..."
for i in {1..15}; do
curl -X POST \
-d "client_id=my-client" \
-d "username=test-user" \
-d "password=test-pass" \
-d "grant_type=password" \
http://$IP_TARGET:30383/auth/realms/master/protocol/openid-connect/token
done
```

```
echo "Running basic nmap scan..."

nmap $IP_TARGET -Pn -A -T4



echo "Sending controlled SYN packets with hping3..."

sudo hping3 --syn -p 8010 -c 100 $IP_TARGET

echo "Testing completed."
```

Self-security has been installed in the following pilots: Pilot 1.1, Pilot 2, Pilot 3, Pilot 4 and Pilot 5.

Below are the detection results for each of these:

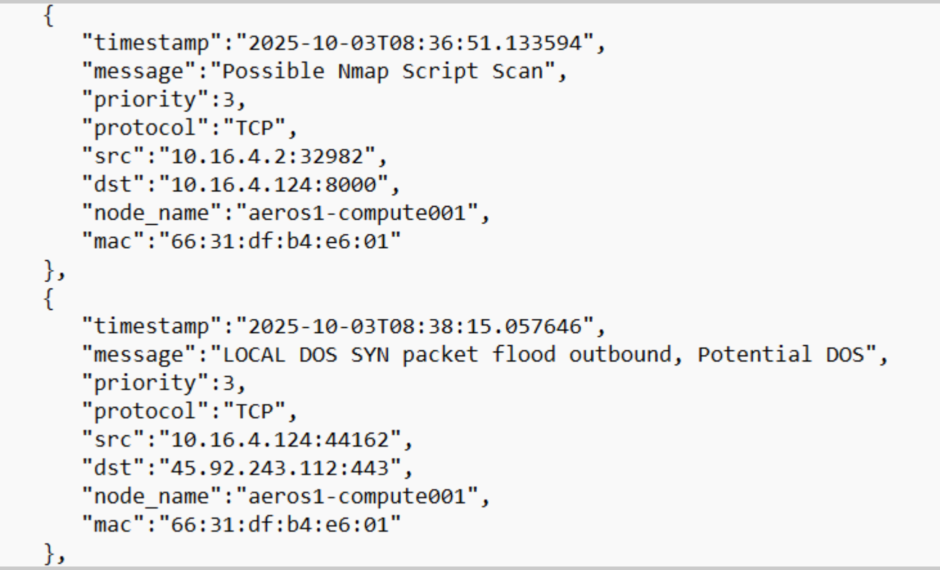
Pilot 1.1

In this pilot, self-security was able to detect the three attacks:

```
  {
    "timestamp": "2025-10-02T11:31:16.687966",
    "message": "Keycloak Brute Force Detected",
    "priority": 2,
    "protocol": "TCP",
    "src": "176.12.87.170:56850",
    "dst": "192.168.15.48:32380",
    "node_name": "servaep1ma01",
    "mac": "00:50:56:b5:09:c7"
  },
  {
    "timestamp": "2025-10-02T11:06:33.624997",
    "message": "Possible Nmap Script Scan",
    "priority": 3,
    "protocol": "TCP",
    "src": "192.168.15.49:32816",
    "dst": "192.168.15.48:8010",
    "node_name": "servaep1ma01",
    "mac": "00:50:56:b5:09:c7"
  },
  {
    "timestamp": "2025-10-02T11:06:33.574301",
    "message": "SYN Flood detected via hping3",
    "priority": 2,
    "protocol": "TCP",
    "src": "192.168.15.49:32816",
    "dst": "192.168.15.48:8010",
    "node_name": "servaep1ma01",
    "mac": "00:50:56:b5:09:c7"
  },
```

Pilot 2

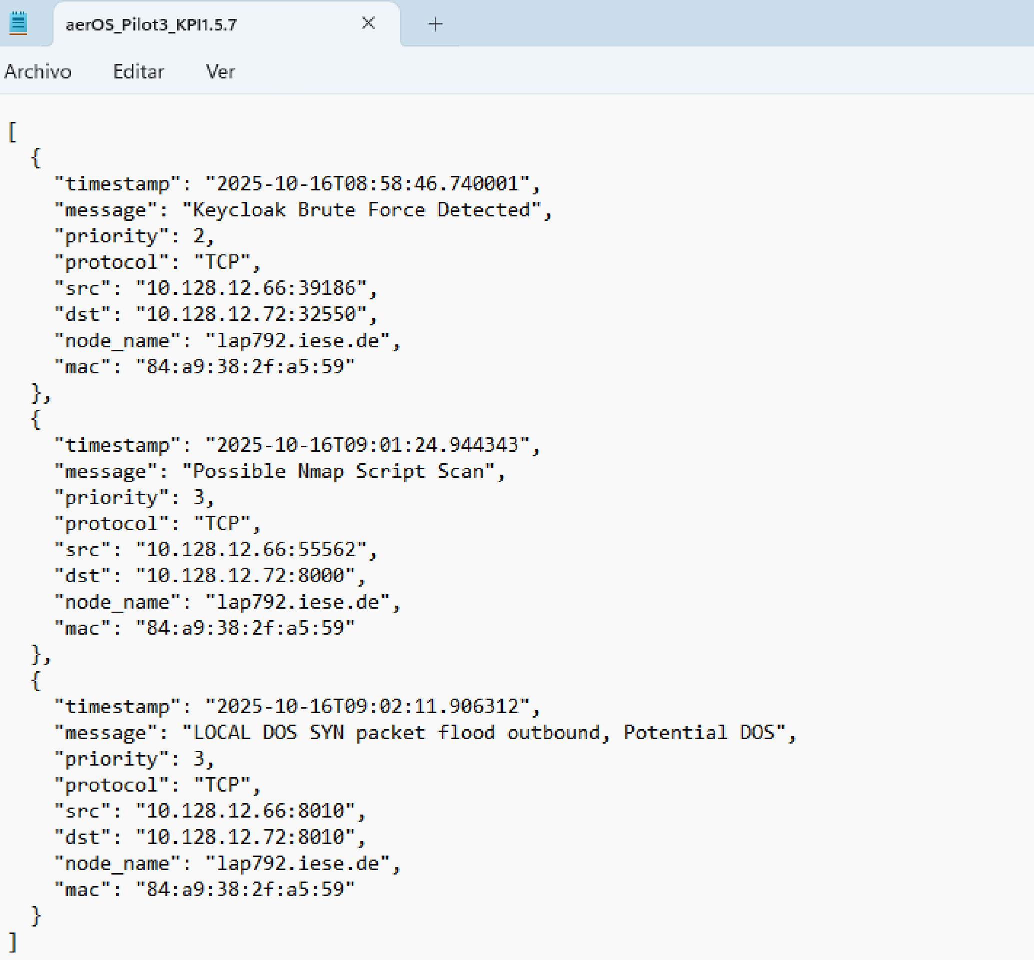
In this pilot, the Keycloak instance is not accessible from the network, this is installed in a cloud that is not accessible. For this cause, in this pilot only 2 attacks were implemented and detected:



```
  {
     "timestamp":"2025-10-03T08:36:51.133594",
     "message":"Possible Nmap Script Scan",
     "priority":3,
     "protocol":"TCP",
     "src":"10.16.4.2:32982",
     "dst":"10.16.4.124:8000",
     "node_name":"aeros1-compute001",
     "mac":"66:31:df:b4:e6:01"
  },
  {
     "timestamp":"2025-10-03T08:38:15.057646",
     "message":"LOCAL DOS SYN packet flood outbound, Potential DOS",
     "priority":3,
     "protocol":"TCP",
     "src":"10.16.4.124:44162",
     "dst":"45.92.243.112:443",
     "node_name":"aeros1-compute001",
     "mac":"66:31:df:b4:e6:01"
  },
```

Pilot 3

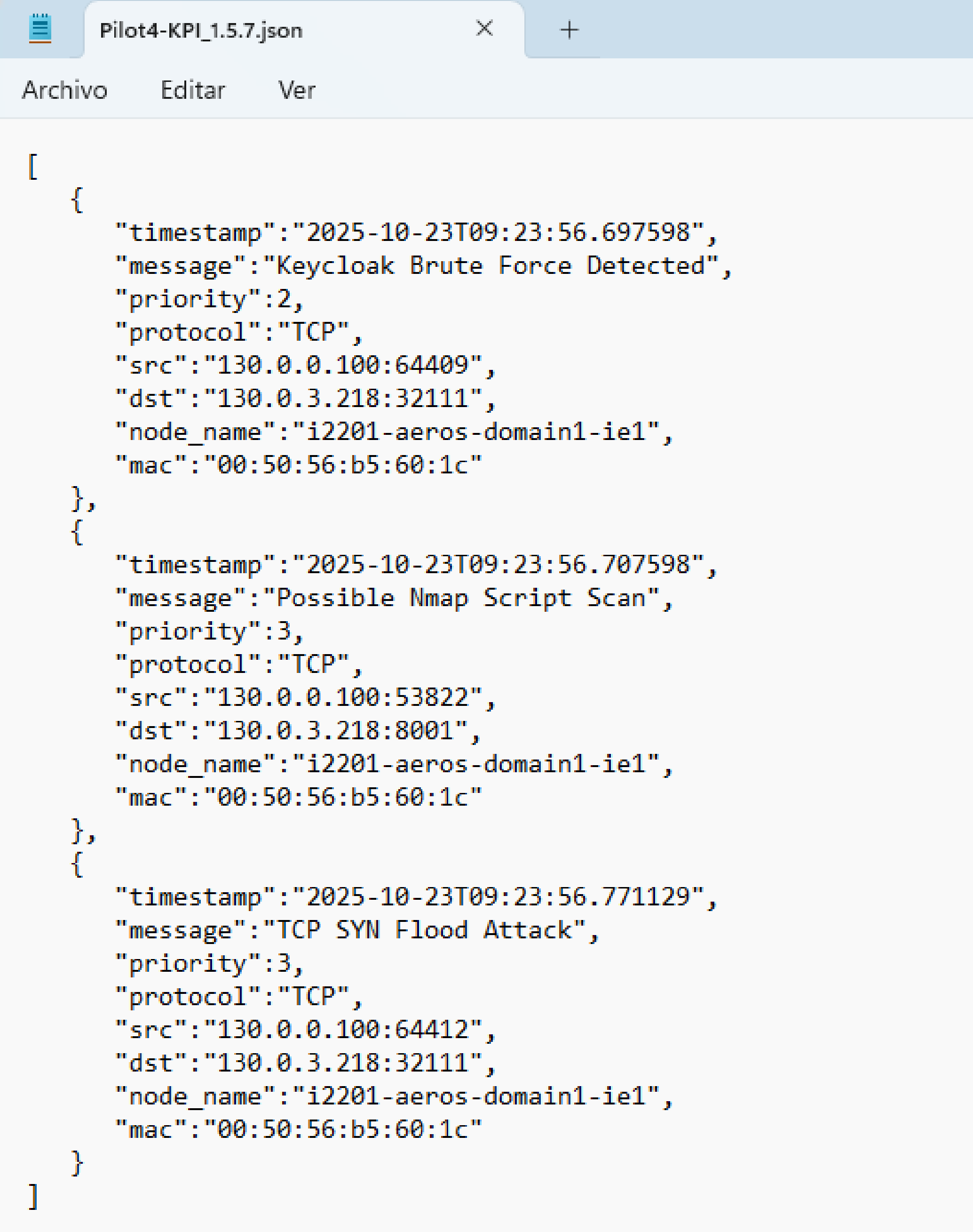
In this pilot, self-security was able to detect the three attacks:



```
[
  {
    "timestamp": "2025-10-16T08:58:46.740001",
    "message": "Keycloak Brute Force Detected",
    "priority": 2,
    "protocol": "TCP",
    "src": "10.128.12.66:39186",
    "dst": "10.128.12.72:32550",
    "node_name": "lap792.iese.de",
    "mac": "84:a9:38:2f:a5:59"
  },
  {
    "timestamp": "2025-10-16T09:01:24.944343",
    "message": "Possible Nmap Script Scan",
    "priority": 3,
    "protocol": "TCP",
    "src": "10.128.12.66:55562",
    "dst": "10.128.12.72:8000",
    "node_name": "lap792.iese.de",
    "mac": "84:a9:38:2f:a5:59"
  },
  {
    "timestamp": "2025-10-16T09:02:11.906312",
    "message": "LOCAL DOS SYN packet flood outbound, Potential DOS",
    "priority": 3,
    "protocol": "TCP",
    "src": "10.128.12.66:8010",
    "dst": "10.128.12.72:8010",
    "node_name": "lap792.iese.de",
    "mac": "84:a9:38:2f:a5:59"
  }
]
```

Pilot 4

In this pilot, self-security was able to detect the three attacks:



```
[
   {
      "timestamp":"2025-10-23T09:23:56.697598",
      "message":"Keycloak Brute Force Detected",
      "priority":2,
      "protocol":"TCP",
      "src":"130.0.0.100:64409",
      "dst":"130.0.3.218:32111",
      "node_name":"i2201-aeros-domain1-ie1",
      "mac":"00:50:56:b5:60:1c"
   },
   {
      "timestamp":"2025-10-23T09:23:56.707598",
      "message":"Possible Nmap Script Scan",
      "priority":3,
      "protocol":"TCP",
      "src":"130.0.0.100:53822",
      "dst":"130.0.3.218:8001",
      "node_name":"i2201-aeros-domain1-ie1",
      "mac":"00:50:56:b5:60:1c"
   },
   {
      "timestamp":"2025-10-23T09:23:56.771129",
      "message":"TCP SYN Flood Attack",
      "priority":3,
      "protocol":"TCP",
      "src":"130.0.0.100:64412",
      "dst":"130.0.3.218:32111",
      "node_name":"i2201-aeros-domain1-ie1",
      "mac":"00:50:56:b5:60:1c"
   }
]
```

Pilot 5

In this pilot, self-security was able to detect the three attacks:

```
[
  {
    "timestamp": "2025-10-02T11:28:26.856308",
    "message": "Keycloak Brute Force Detected",
    "priority": 2,
    "protocol": "TCP",
    "src": "172.16.0.65:43394",
    "dst": "172.16.0.64:30032",
    "node_name": "node01",
    "mac": "00:0c:29:cb:df:53"
  },
  {
    "timestamp": "2025-10-02T11:30:09.657096",
    "message": "Possible Nmap Script Scan",
    "priority": 3,
    "protocol": "TCP",
    "src": "172.16.0.65:55328",
    "dst": "172.16.0.64:8010",
    "node_name": "node01",
    "mac": "00:0c:29:cb:df:53"
  },

  {
    "timestamp": "2025-10-02T11:30:10.127149",
    "message": "SYN Flood detected via hping3",
    "priority": 2,
    "protocol": "TCP",
    "src": "172.16.0.65:55460",
    "dst": "172.16.0.64:8010",
    "node_name": "node01",
    "mac": "00:0c:29:cb:df:53"
  },
```

aerOS decentralised AI

KPI 1.6.1 Realising decentralized AI/ML with scalability comparable to centralized approach (KVI-4.1)

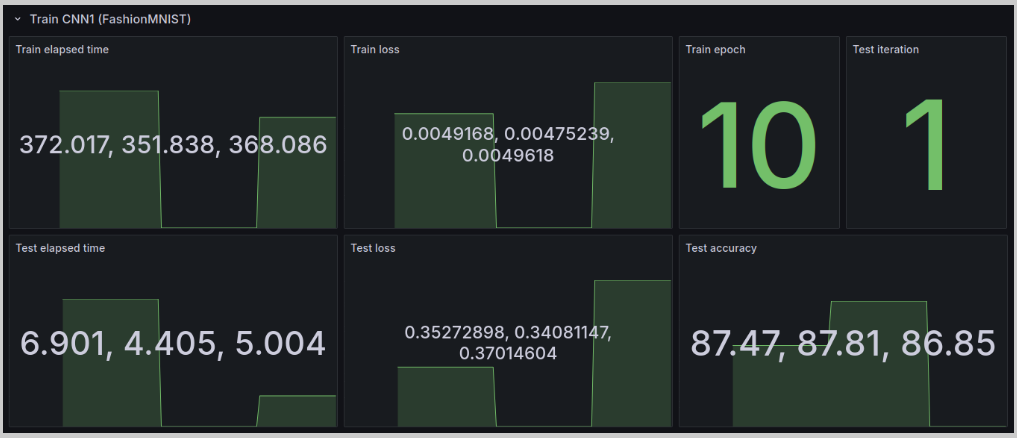
Table 61: KPI 1.6.1 Realising decentralized AI/ML with scalability comparable to centralized approach (KVI-4.1)

KPI ID number and partner resp.	**KPI 1.6.1**
KPI Name	**Realising decentralized AI/ML with scalability comparable to centralized approach (KVI-4.1)**

Description	Scalability is the ability of AI algorithms, data, models, and infrastructure to operate at the size, speed, and complexity required. aerOS should operate with or be validated with at least three applications of decentralized AI.		
Motivation	Decentralized AI/ML should not negatively influence operations of AI-based system compared with a centralized AI/ML.		
Target value	>=3 applications		
Prerequisites	aerOS deployment ready with final (or close to final) version of base components and aux AI components ready for evaluation		
aerOS components (task)	AI Task Controller, AI Local Executor (T4.3)		
Evaluation means	Three decentralized AI applications will be identified in and their scalability will be evaluated with proper tests or justification. One application will be based on experiments on decentralized vs centralized model training, the other two will be based on model inference in a decentralized and centralized approach. For model training evaluation metrics will cover time of training, trained model performance. For model inference metrics will cover: inference time and resource utilization (memory, CPU). Plan for a model training-based application: • Run model training at least 3 times in a centralized approach (can be outside aerOS) and measure evaluation metrics • Prepare a model to be trained with a decentralized approach on several aerOS IEs • Run at least 3 times the training process using IEs selected by aerOS and each time measuring evaluation metrics • Results from decentralized tests will be averaged to be compared with averaged results for a centralized approach Plan for a model inference-based application: • Run at least 3 times a set number of inferences on a model over a selected period of time in a centralized approach (can be outside aerOS deployment) and each time measure evhttps://nextcloud.aeros-project.eu/apps/files/files/291863?dir=%2FaerOS%2F2%20-%20Work%20Packages%2FWP5%20-%20aerOS%20integration%2C%20use%20cases%20deployment%20and%20validation%2FDeliverables%2FD5.6%2FOLD%20FILES&openfile=trueal uation metrics • Run at least 3 times a set number of inferences on a model over a selected period of time using IEs selected by aerOS and each time measure evaluation metrics • Results from decentralized tests will be averaged to be compared with averaged results for a centralized approach		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**

<table>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>N/A</i></td><td><i>100% achieved: 3 applications (2 inference, 1 training)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">According to the stated evaluation means, three different machine learning applications were tested upon aerOS continuum. One application for comparing the decentralized and centralized training with a CNN model trained on the FashionMNIST dataset. Two applications for testing the inference behavior in centralized and decentralized environments a CNN and MLP models evaluated on the MNIST dataset. We have measured and compared the evaluation metrics.
During the experiments, models were first trained and then tested to evaluate their performance. We collected train and test loss values from each run, as well as test accuracy. Additionally, the training and testing procedures were benchmarked to gather the elapsed time (in seconds). Finally, we measured the resource (RAM and CPU) usage during the experiments.
In both scenarios, the experiments were run on GNU/Linux-based OS, on x64 CPUs (14 cores for local setup, 4 cores for aerOS setup), and in both cases with 16GB of RAM. All resource consumption metrics were measured and recorded with k8s tools (i.e., Prometheus, Grafana, Kepler).
First, the models were ran outside of aerOS. inside a local cluster. The results of the resource usage during the experiments are as follows.

First, these are the results of the CNN (2 hidden layers) training on the FashionMNIST dataset. The model was trained in 10 epochs. After the last epoch it was tested (once). The consecutive number are the results from the three runs.</td></tr>
</table>



Train CNN1 (FashionMNIST)

Train elapsed time: 372.017, 351.838, 368.086

Train loss: 0.0049168, 0.00475239, 0.0049618

Train epoch: 10

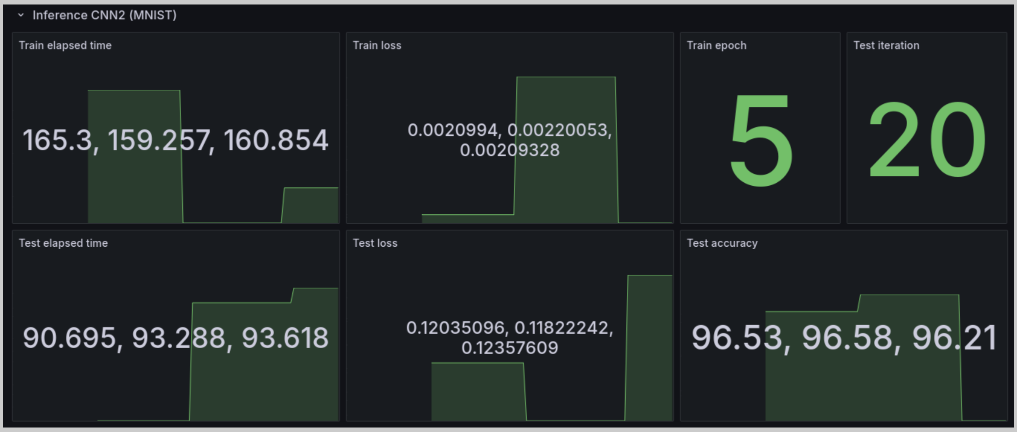
Test iteration: 1

Test elapsed time: 6.901, 4.405, 5.004

Test loss: 0.35272898, 0.34081147, 0.37014604

Test accuracy: 87.47, 87.81, 86.85

Next, a different CNN model (with 2 hidden layers) trained on MNIST. This time the model was trained in 5 epochs. However, it was tested 20 times to get a more precise average behavior (time) during inference.



Inference CNN2 (MNIST)

Train elapsed time: 165.3, 159.257, 160.854

Train loss: 0.0020994, 0.00220053, 0.00209328

Train epoch: 5

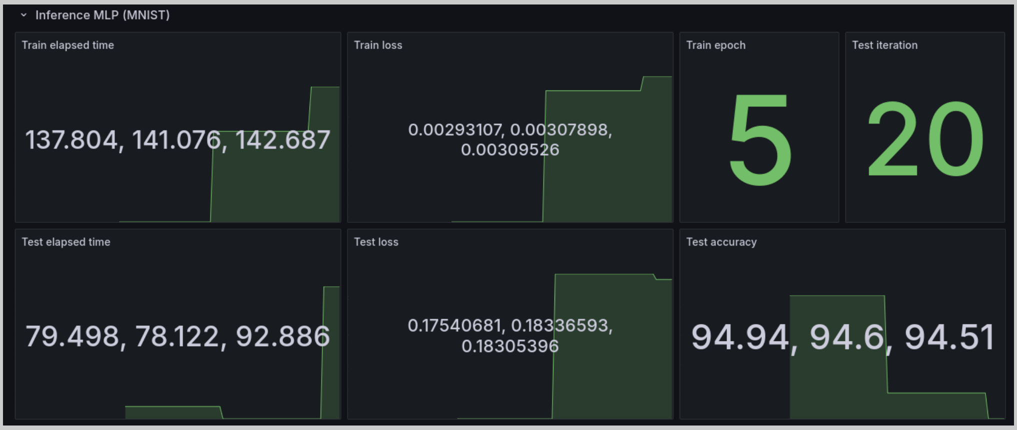
Test iteration: 20

Test elapsed time: 90.695, 93.288, 93.618

Test loss: 0.12035096, 0.11822242, 0.12357609

Test accuracy: 96.53, 96.58, 96.21

Finally, the MLP model (with 3 hidden layers) results. The experiments setup was the same as for the previous CNN model.



Inference MLP (MNIST)

Train elapsed time: 137.804, 141.076, 142.687

Train loss: 0.00293107, 0.00307898, 0.00309526

Train epoch: 5

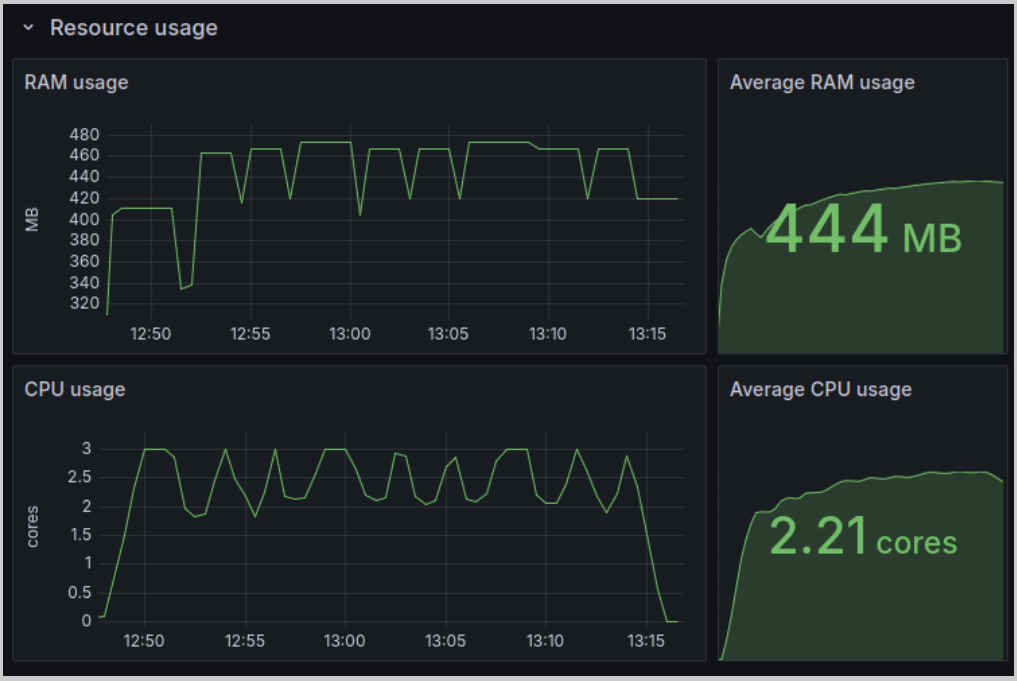
Test iteration: 20

Test elapsed time: 79.498, 78.122, 92.886

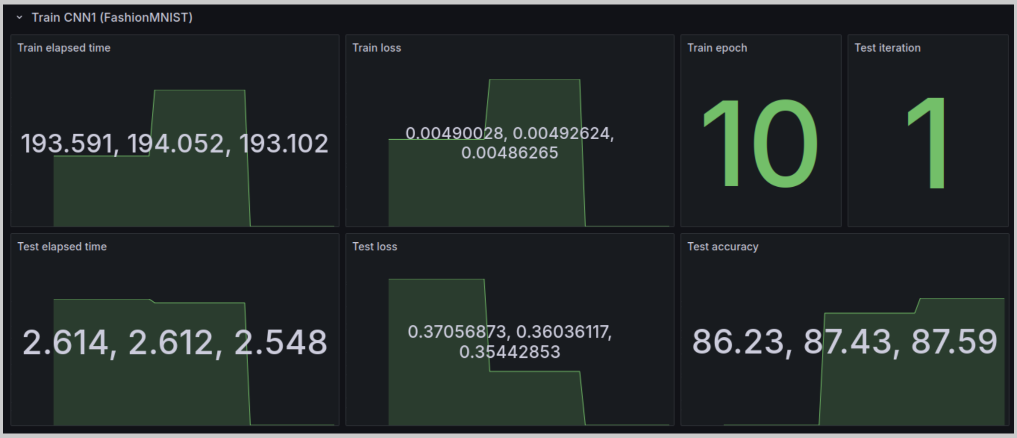
Test loss: 0.17540681, 0.18336593, 0.18305396

Test accuracy: 94.94, 94.6, 94.51

Next, the experiments were run in aerOS cluster. The setup for the experiments was the same as for the local experiments. The results are as follows. One can see that the average CPU and RAM usage was slightly lower, in favor of aerOS.



Next, the CNN trained on FashionMNIST had a very similar quality results (as expected), however the elapsed time needed for training (and inference) was better (about 2 times faster).



Train elapsed time: 193.591, 194.052, 193.102

Train loss: 0.00490028, 0.00492624, 0.00486265

Train epoch: 10

Test iteration: 1

Test elapsed time: 2.614, 2.612, 2.548

Test loss: 0.37056873, 0.36036117, 0.35442853

Test accuracy: 86.23, 87.43, 87.59

The inference-focused tests on the CNN model and the MNIST dataset showed that the quality remains unchanged however the inference is faster.

And here, similarly to the previous CNN model, the MLP time results are again better in favor of aerOS.

Therefore, the conclusion is that using aerOS for the AI experiments had no negative impact on the operation of the models. Furthermore, their speed and resource usage was improved in aerOS. The code of the experiments is available here: https://gitlab.aeros-project.eu/wp4/t4.3/ai-experiments/-/tree/dev

Moreover, the model training was also tested using Federated Learning (FL) components. To compare the performance between centralized and decentralized training approaches, experiments were carried out on two different infrastructures:

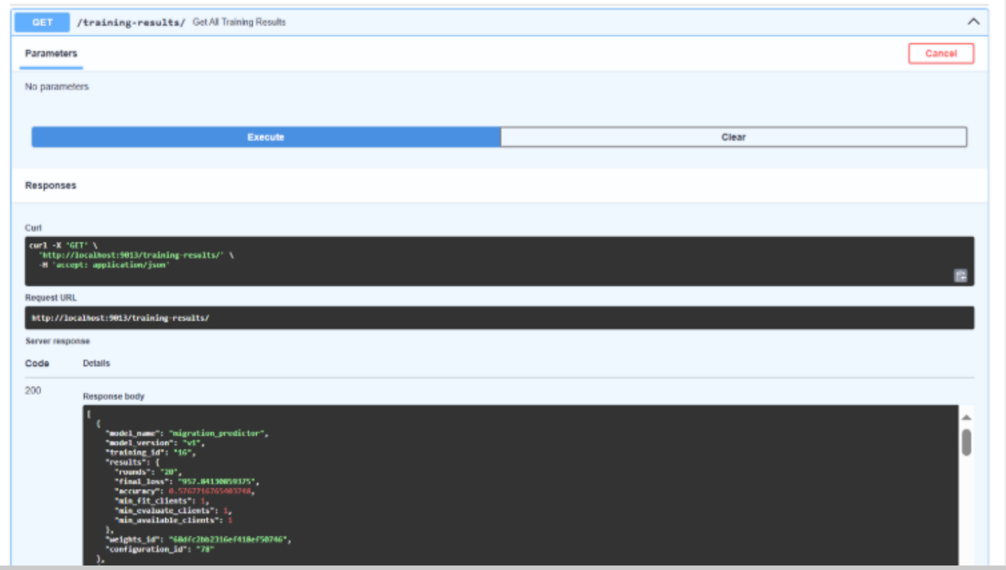
1. Single Local Virtual Machine (Centralized setup): **CPU: Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz (2 cores, 4 processors), RAM: 16 GB** Services were deployed and run as images within **amd64** Linux Docker containers, so that a consistent software environment is ensured across conducted experiments.
2. aerOS Continuum (Decentralized setup): **Deployed semi-automatically on two identical nodes (IEs) within the CloudFerro domain with the parameter, CPU: x64 Architecture, 4 cores, RAM: 15.6 GB**

Training was performed simultaneously on two clients (Local Operations), with the Federated Averaging (FedAvg) algorithm used to aggregate model updates across the clients. The trained model was a simple neural network with a single hidden layer. This lightweight design was intentionally chosen, as the primary goal was to evaluate the performance of training within the aerOS continuum, rather than to maximize model accuracy.

The training dataset was generated semi-synthetically using the Extended Green Cloud Simulator (EGCS), a simulation model of CloudFerro green edge infrastructure. It represents resource utilization during computational task execution throughout the entire year and combines: (1) real weather condition data monitored by Electrum in the year 2024, (2) simulated resource utilization, (3) simulated resource demands of computational tasks generated based on real tasks processed in CloudFerro infrastructure. The training objective was to learn an effective task migration strategy — specifically, to identify when factors such as CPU usage and weather conditions should trigger task migration. The ultimate aim was to maximize the utilization of servers powered by green energy.

The dataset was composed of 10156 observations in total and was divided into 4 different subsets, each corresponding to an individual season of the year: (1) 2539 observations, (2) 2750 observations, (3) 2727 observations, (4) 2143 observations. Such division was made in order to separate the patterns in the data that could depend on individual weather characteristics that may differ between the seasons. All of the subsets were subsequently randomly divided in half, as such representing the client's local data used for training. The experiments were run for each data subset separately, resulting in running 4 training experiments on each infrastructure.

The results of the experiments were stored using the FL repository (i.e. service belonging to FL components collection) and collected using its Swagger API as indicated on the following figure:

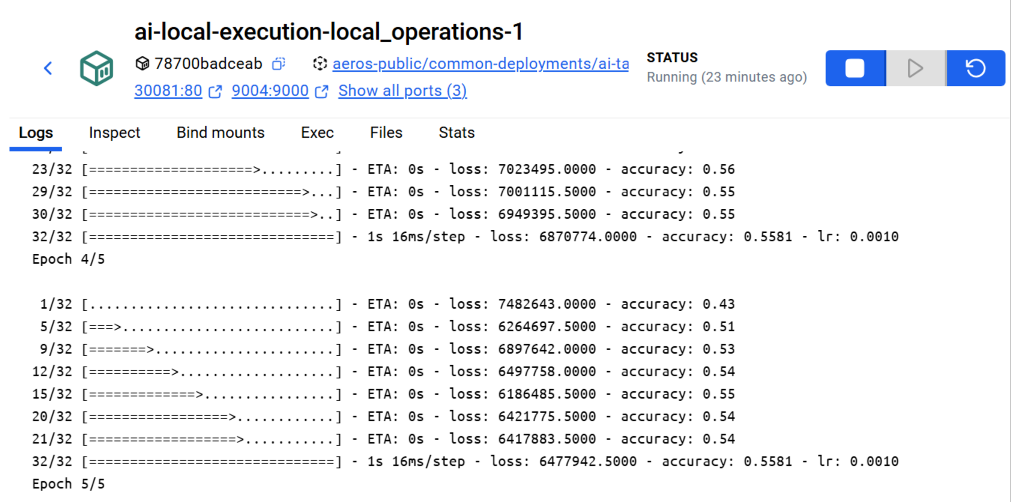


```
curl -X 'GET' \
  'http://localhost:9013/training-results/' \
  -H 'accept: application/json'
```

```
http://localhost:9013/training-results/
```

```
[
  {
    "model_name": "migration_predictor",
    "model_version": "v1",
    "training_id": "16",
    "results": {
      "rounds": "20",
      "final_loss": "957.841308593375",
      "accuracy": 0.5767716765403748,
      "min_fit_clients": 1,
      "min_evaluate_clients": 1,
      "min_available_clients": 1
    },
    "weights_id": "68dfc2bb2316ef418ef50746",
    "configuration_id": "78"
  },
```

Moreover, the training time was obtained by processing logs coming from individual client's containers



```
23/32 [====================>.........] - ETA: 0s - loss: 7023495.0000 - accuracy: 0.56
29/32 [==========================>...] - ETA: 0s - loss: 7001115.5000 - accuracy: 0.55
30/32 [===========================>..] - ETA: 0s - loss: 6949395.5000 - accuracy: 0.55
32/32 [==============================] - 1s 16ms/step - loss: 6870774.0000 - accuracy: 0.5581 - lr: 0.0010
Epoch 4/5

 1/32 [..............................] - ETA: 0s - loss: 7482643.0000 - accuracy: 0.43
 5/32 [===>..........................] - ETA: 0s - loss: 6264697.5000 - accuracy: 0.51
 9/32 [=======>......................] - ETA: 0s - loss: 6897642.0000 - accuracy: 0.53
12/32 [==========>...................] - ETA: 0s - loss: 6497758.0000 - accuracy: 0.54
15/32 [=============>................] - ETA: 0s - loss: 6186485.5000 - accuracy: 0.55
20/32 [=================>............] - ETA: 0s - loss: 6421775.5000 - accuracy: 0.54
21/32 [==================>...........] - ETA: 0s - loss: 6417883.5000 - accuracy: 0.54
32/32 [==============================] - 1s 16ms/step - loss: 6477942.5000 - accuracy: 0.5581 - lr: 0.0010
Epoch 5/5
```

The results for all experiments were then averaged in order to compare centralized and decentralized approaches. In particular, the following outcomes were obtained:

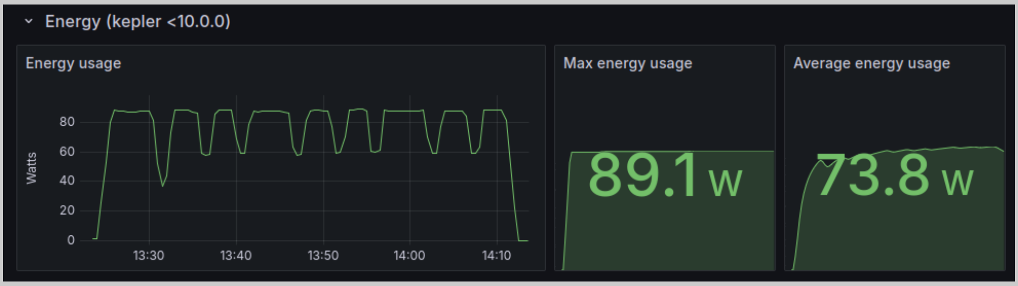
1. Centralized: **57.2% (Accuracy), 116.2 s (Training Time)**
2. Decentralized: **54.2% (Accuracy), 102.8 s (Training Time)**

The average results showcase that the decentralized approach yields comparable results to the centralized one. Detailed results of the experiments with the applied experiments setup can be found in https://gitlab.aeros-project.eu/wp4/t4.3/fl-experiments.

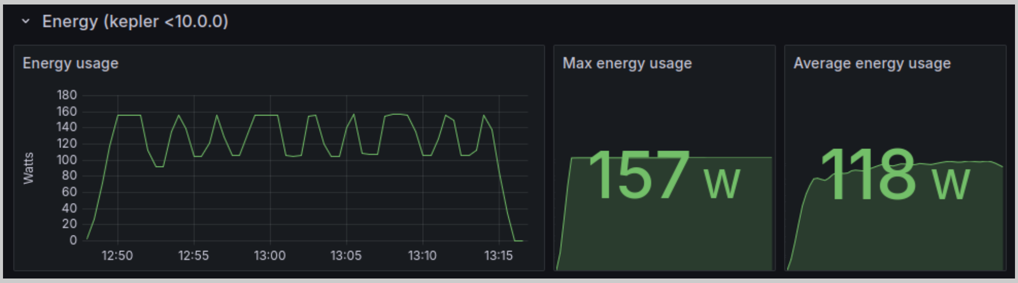
KPI 1.6.2 Energy consumption reduction due to moving AI from cloud to the edge (KVI-4.2)

Table 62: KPI 1.6.2 Energy consumption reduction due to moving AI from cloud to the edge (KVI-4.2)

KPI ID number and partner resp.	**KPI 1.6.2**
KPI Name	**Energy consumption reduction due to moving AI from cloud to the edge (KVI-4.2)**
Description	Energy consumption should be decreased for AI being run closer to the edge, possibly on local data and with frugal adjustment.
Motivation	aerOS aims to establish what are the benefits and trade-off resulting from moving AI closed to the edge.
Target value	> 50% (on average on tested scenarios)
Prerequisites	aerOS deployment ready with final (or close to final) version of base components and aux AI components ready for evaluation in development/integration environment.
aerOS components (task)	AI Local Executor, AI Task Controller, frugal techniques (T4.3)
Evaluation means	Experiments will be conducted to measure energy consumption when running model inference on elements with different processing capabilities – cloud vs edge. For edge

	deployments frugal techniques will likely be applied. During analysis in M24-M36 a proper evaluation metrics will be selected. Evaluation plan: - Running at least 3 times a set of inferences over a period of time over a model deployed in the cloud where data needs to be sent from edge IEs to the cloud; each time measure evaluation metrics - Running at least 3 times a set of inferences over a period of time over a model deployed on edge IEs with local access to data; each time measure evaluation metrics - Running at least 3 times a set of inferences over a period of time over a frugal model (model from previous points after application of quantization/pruning) deployed on edge IEs with local access to data; each time measure evaluation metrics - Compared averaged results from three above options - The energy consumption of the running processes is foreseen to be established from information gathered using Kepler (a k8s monitoring tool).		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***From research literature:*** - ***BERT LM*** - ***experiment used 8 V100 GPUs for 36 hours and used a total of 37 kWh.*** - ***Three sizes of DenseNets on MNIST lasted between 20 and 25 minutes and consumed between 20 and 38Wh*** - ***Energy reduction achieved with proposed methods for Microsoft Azure cloud compute platform was less than 27%.*** - ***Energy consumption on centralized vs distributed approach decreased, on average, less than 10%.***	***N/A***	***15-35%***
Outcome elaboration (M38)	The final experiments included not only model inference but also **model training**, as the training process is more resource-intensive and computationally demanding, making it a better indicator of energy consumption. The experiments followed the same scenarios defined in KPI 1.6.1. These are the results of the experiments from the previous KPI (CNN/MLP, MNIST/FashionMNIST) running outside of aerOS. 		

And these are the results obtained in aerOS.



One can notice that the power usage in aerOS is higher. However, it's important to note that the time required to finish the experiments was different. For aerOS it was 29 minutes, and outside of aerOS it was 51 minutes. Therefore, the energy consumption was as follows.

Outside aerOS: **~225 kJ**

Inside aerOS: **~205 kJ**

Therefore the energy consumption during the experiments was reduced by **~9%** in aerOS.

Next, what follows are the FL experiments. The experiments were carried out on two different infrastructures:

2. Single Local Virtual Machine (Centralized setup): **CPU: Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz (2 cores, 4 processors), RAM: 16 GB** Services were deployed and run as images within **amd64** Linux Docker containers, so that a consistent software environment is ensured across conducted experiments.
3. aerOS Continuum (Decentralized setup): **Deployed semi-automatically on two identical nodes (IEs) within the CloudFerro domain with the parameter, CPU: x64 Architecture, 4 cores, RAM: 15.6 GB**

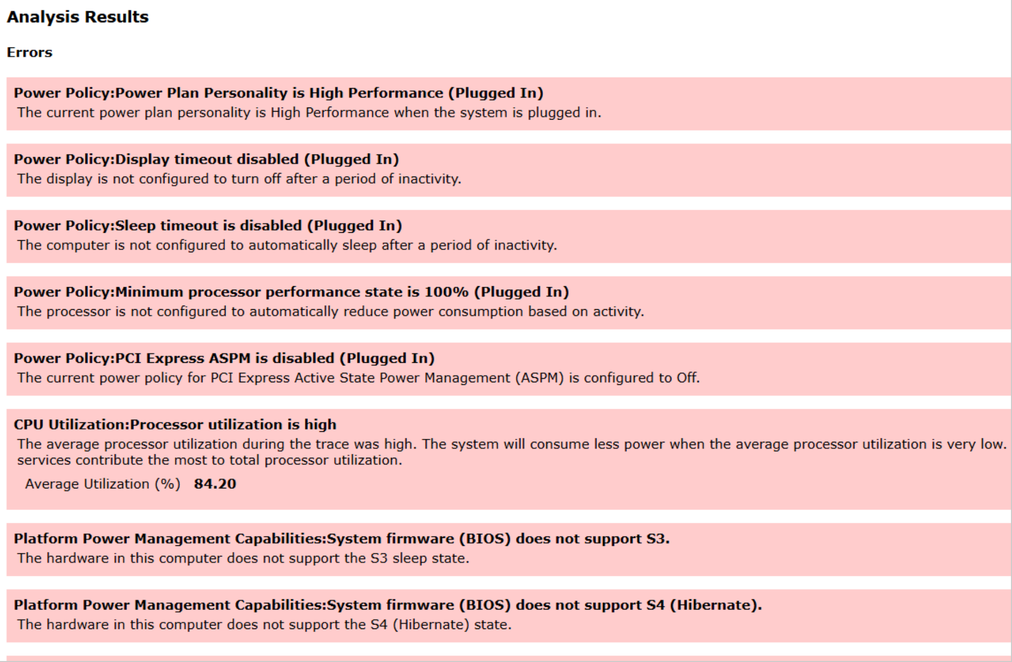
Training was performed simultaneously on two clients (Local Operations), with the Federated Averaging (FedAvg) algorithm used to aggregate model updates across the clients. The trained model was a simple neural network with a single hidden layer.

The training dataset was generated semi-synthetically using the Extended Green Cloud Simulator (EGCS), a simulation model of CloudFerro green edge infrastructure. It represents resource utilization during computational task execution throughout the entire year and combines: (1) real weather condition data monitored by Electrum in the year 2024, (2) simulated resource utilization, (3) simulated resource demands of computational tasks generated based on real tasks processed in CloudFerro infrastructure. The training objective was to learn an effective task migration strategy — specifically, to identify when factors such as CPU usage and weather conditions should trigger task migration. The ultimate aim was to maximize the utilization of servers powered by green energy.

The dataset was composed of 10156 observations in total and was divided into 4 different subsets, each corresponding to an individual season of the year: (1) 2539 observations, (2) 2750 observations, (3) 2727 observations, (4) 2143 observations. Such division was made in order to separate the patterns in the data that could depend on individual weather characteristics that may differ between the seasons. All of the subsets were subsequently randomly divided in half, as such representing the client's local data used for training. The experiments were run for each data subset separately, resulting in running 4 training experiments on each infrastructure.

The results of the experiments were stored using the FL repository (i.e. service belonging to FL components collection) and collected using its Swagger API.

To measure energy consumption on the local machine, the Windows *powercfg* utility was used to generate an energy report, as presented in the following figures.



Analysis Results

Errors

Power Policy:Power Plan Personality is High Performance (Plugged In)
The current power plan personality is High Performance when the system is plugged in.

Power Policy:Display timeout disabled (Plugged In)
The display is not configured to turn off after a period of inactivity.

Power Policy:Sleep timeout is disabled (Plugged In)
The computer is not configured to automatically sleep after a period of inactivity.

Power Policy:Minimum processor performance state is 100% (Plugged In)
The processor is not configured to automatically reduce power consumption based on activity.

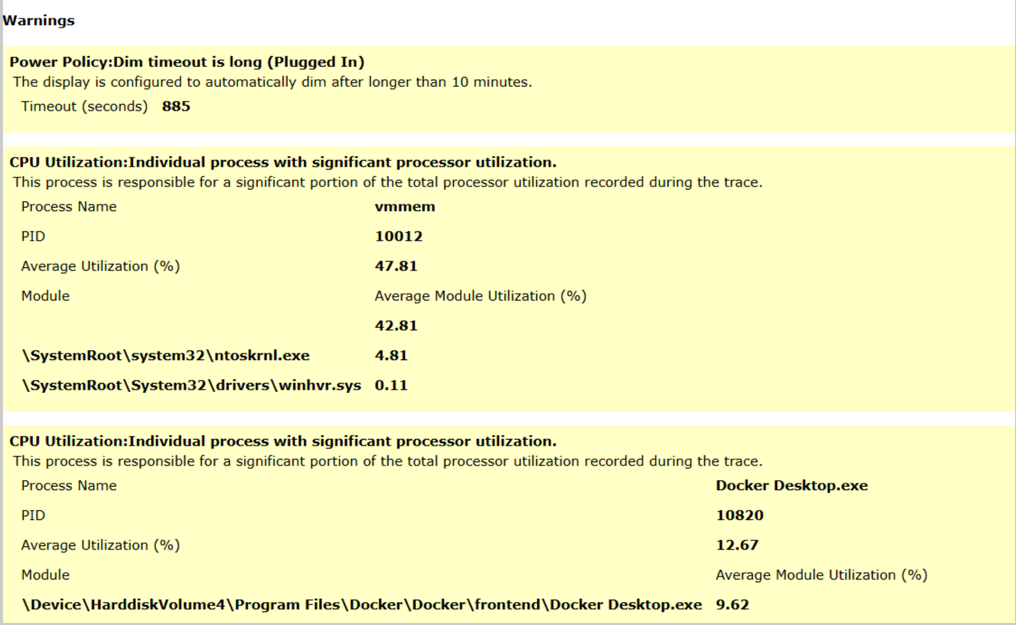
Power Policy:PCI Express ASPM is disabled (Plugged In)
The current power policy for PCI Express Active State Power Management (ASPM) is configured to Off.

CPU Utilization:Processor utilization is high
The average processor utilization during the trace was high. The system will consume less power when the average processor utilization is very low. services contribute the most to total processor utilization.

Average Utilization (%)	**84.20**

Platform Power Management Capabilities:System firmware (BIOS) does not support S3.
The hardware in this computer does not support the S3 sleep state.

Platform Power Management Capabilities:System firmware (BIOS) does not support S4 (Hibernate).
The hardware in this computer does not support the S4 (Hibernate) state.



Warnings

Power Policy:Dim timeout is long (Plugged In)
The display is configured to automatically dim after longer than 10 minutes.

Timeout (seconds)	**885**

CPU Utilization:Individual process with significant processor utilization.
This process is responsible for a significant portion of the total processor utilization recorded during the trace.

Process Name	**vmmem**
PID	**10012**
Average Utilization (%)	**47.81**
Module	Average Module Utilization (%)
	42.81
\SystemRoot\system32\ntoskrnl.exe	**4.81**
\SystemRoot\System32\drivers\winhvr.sys	**0.11**

CPU Utilization:Individual process with significant processor utilization.
This process is responsible for a significant portion of the total processor utilization recorded during the trace.

Process Name	**Docker Desktop.exe**
PID	**10820**
Average Utilization (%)	**12.67**
Module	Average Module Utilization (%)
\Device\HarddiskVolume4\Program Files\Docker\Docker\frontend\Docker Desktop.exe	**9.62**

A custom script was then developed to extract relevant resource utilization metrics from this report, such as: (1) CPU utilization, (2) timer resolution requests, and (3) counts of devices preventing sleep states.

These metrics were subsequently used to estimate the system's energy consumption. The script used in these computations is provided along with the detailed experimental results in https://gitlab.aeros-project.eu/wp4/t4.3/fl-experiments.

For the aerOS continuum setup, **Kepler** (Kubernetes-based Efficient Power Level Exporter) was used to monitor and measure energy consumption at the node level for each client. The final energy consumption results were then averaged for each of the conducted experiments. In particular, the following results have been obtained:

1. **Subset 1: 44064J** (Local machine), **29035J** (aerOS), ~**35%** reduction
2. **Subset 2: 45744J** (Local machine), **39304J** (aerOS), **~15%** reduction
3. **Subset 3: 38016J** (Local machine), **36261J** (aerOS), **~5%** reduction
4. **Subset 4: 42048J** (Local machine), **28981J** (aerOS), **~32%** reduction

Observably, the highest reduction was obtained for the experiments conducted on Subset 1 and Subset 2, which may be attributed to smaller size of these dataset and shorter time of training. Specifically, in those cases the the resources were occupied for a shorter time with running services, which maintenance contributed the most to the energy consumption on local machine. While minor variations may still arise from hardware differences, using containerized execution in Linux-based environments for both experiments supports the argument that the observed reductions in energy primarily reflect the effects of decentralized execution, resource distribution, and workload scheduling, rather than OS-level or software discrepancies. It is also important to stress that although the achieved reduction is below estimated threshold of 50%, the conducted experiments did not cover the reduction of energy associated with data transfers that are a typical in cloud systems.

To roughly estimate the potential reduction that can come up from omitting the necessity of data transfers, a preliminary experiment was run, where the entire dataset used for training was transferred using *scp* command between two VMs (with parameters equivalent to those of the centralized test). During the experiment, the CPU and memory usage were measured and then used to estimate average power consumption. Throughout the data transfer, it the average power usage was equal to **23.72W** with total energy reaching **14877.29J.** When scaled proportionally to the sizes of the sub-datasets, this value represents approximately **8%** of the total energy consumed in the local setup.

Although this estimate should be interpreted with caution (given the limitations in measurement accuracy) it nevertheless suggests that a decentralized approach could yield up to **8% energy savings** by eliminating the need for data transfer.

The original target value of above 50% was very ambitious since energy consumption depends on various factors (model architecture, data location, resources).

From research literature referenced in the KPI baseline description, results from other studies reported energy consumption reduction of less than 27% and less than 10%.

Moreover:

• In https://doi.org/10.1016/j.iot.2023.100930 - *Power consumption reduction for IoT devices thanks to Edge-AI: Application to human activity recognition* – reported achieved energy consumption reduction was up to 21%.

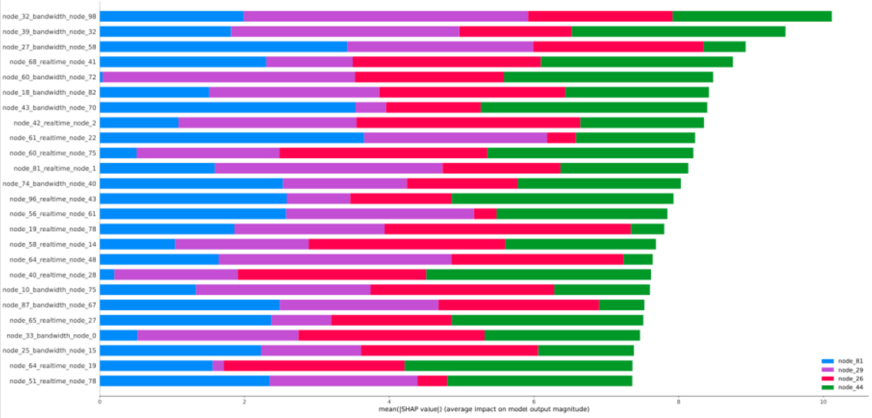
• In article *New tools are available to help reduce the energy that AI models devour* (https:// news.mit.edu/2023/new-tools-available-reduce-energy-that-ai-models-devour-1005) - up to 80% reduction in energy consumption can be obtained by training the model with proper early stopping criterion (note: this result is very promising however it is not related to moving AI from cloud to edge). The largest contributor to emissions is reported to be model inference and it can be addressed by proper hardware selection. Northwestern University reported that they created an optimiser than

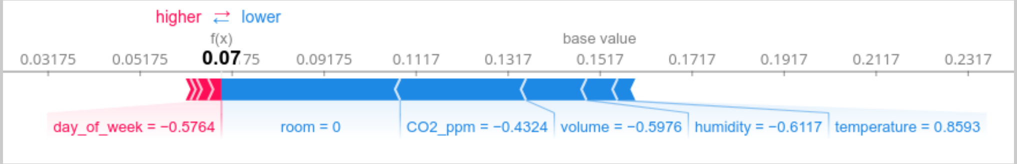
	matches model with hardware and resulting energy consumption reduction was by 10-20% without significant drop in quality of service. • In https://arxiv.org/abs/2206.05229 - *Measuring the Carbon Intensity of AI in Cloud Instances* - paper is about CO2 emission, however energy consumption values fo different models (LM, NLP, computer graphics) are also given along with experiments on emission reduction. Emission reduction formula uses energy consumption, and achieved emission reduction results were about 27%. Even though methods described in the article are different that aerOS approach, the result can be meaningful to our estimations. • In https://ieeexplore.ieee.org/document/8668812/ - The study showcase that the energy consumption can be reduced from 14% up to 25% by fully shifting to distributed edge architectures. In such cases, the total consumption heavily depends on the network energy consumption.

KPI 1.6.3 Validation of comprehensive support, by aerOS, for distributed frugal AI components with explainability (KVI-4.3)

Table 63: KPI 1.6.3 Validation of comprehensive support, by aerOS, for distributed frugal AI components with explainability (KVI-4.3)

KPI ID number and partner resp.	**KPI 1.6.3**
KPI Name	**Validation of comprehensive support, by aerOS, for distributed frugal AI components with explainability (KVI-4.3)**
Description	Identification of applications (within use cases/scenarios) in which distributed frugal AI components potentially supported by explainability should be applicable.
Motivation	aerOS will operate on heterogenous Infrastructure Elements with both internal decision-making and potentially AI-related services. This heterogeneity may require for application of frugal techniques to enable effective operations and interpretability/explainability to provide additional information about the decision making.
Target value	>= 2 frugal applications, >= 2 XAI applications
Prerequisites	Advanced or finalized design of aerOS internal operations and architectures for the pilots
aerOS components (task)	AI Local Executor, AI Task Controller, frugality techniques, explainability techniques (T4.3)
Evaluation means	An in-depth review of all the aerOS scenarios and continuum was addressed. Every AI service that is trained with: (1) datasets sizes smaller by min 30% of the estimated full dataset, (2) using resources with limited capacities requiring application of frugal techniques, will be considered as a frugal application. A survey was conducted with different end-users from those aerOS scenarios that claim that are making use of explainable AI. If the feedback obtained is higher than 50%, it will be considered as an acceptable XAI application.

<table>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>1 XAI application in progress, 1 identified to be done – 40%</i>
<i>0 frugal applications (but still under analysis)</i>
<i>Total = 20%</i></td><td><i>2 XAI applications (internal use case and in pilot 5)</i>
<i>2 frugal applications (both in pilot 5)</i>
<i>Total = 100%</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">XAI approaches have been successfully applied in two different scenarios. The first scenario involved the aerOS internal resource allocator (HLO) that used a reinforcement learning approach in its decision-making process. Our proposed solution for explaining the HLO behaviour was to apply SHAP (SHapley Additive exPlanations) method to understand what input features to the HLO were the most important for each decision it took. The solution was deployed as a function in aerOS using the Embedded Analytics Tool (EAT) that provided an environment for the function to run and a tool to visualize the results of the function (i.e., Grafana). The code of the explainer is available here: https://gitlab.aeros-project.eu/wp4/t4.3/explainability-service/-/tree/main/hlo-explainer-faas
An example of the visualization, where each row represents an input feature and its magnitude of importance for a particular decision:

Next, using the gathered experience, we determined another useful application of the XAI in aerOS. The use case was observed for the Pilot 5, where a tree-based machine learning algorithm (i.e., xgboost) was used for predicting energy consumption. Our approach for explaining the model outcomes was to use SHAP-based explainer. The solution was deployed like before, as a function inside EAT in the pilot's cluster. The code of the explainer is available here: https://gitlab.aeros-project.eu/pilots/pilot-5/forecasting-health-energy/-/tree/dev/src/app/xai/openfaas
The visualization uses a force plot, where each input feature's impact on the expected output value (either increasing it or lowering it), leading to the final ML model output. Here is an example taken from Pilot 5:</td></tr>
</table>



The application of XAI was measued by number of use cases addressed. The survey was not conducted as an evaluation method because in case of the mentioned use cases there was a close collaboration with target reciepients of XAI output. This was a small and very targeted group so survey would not have provided any additional insight. Survey would be applicable in case of XAI directed at more heterogenous end user which was not the case in analyzed use cases.

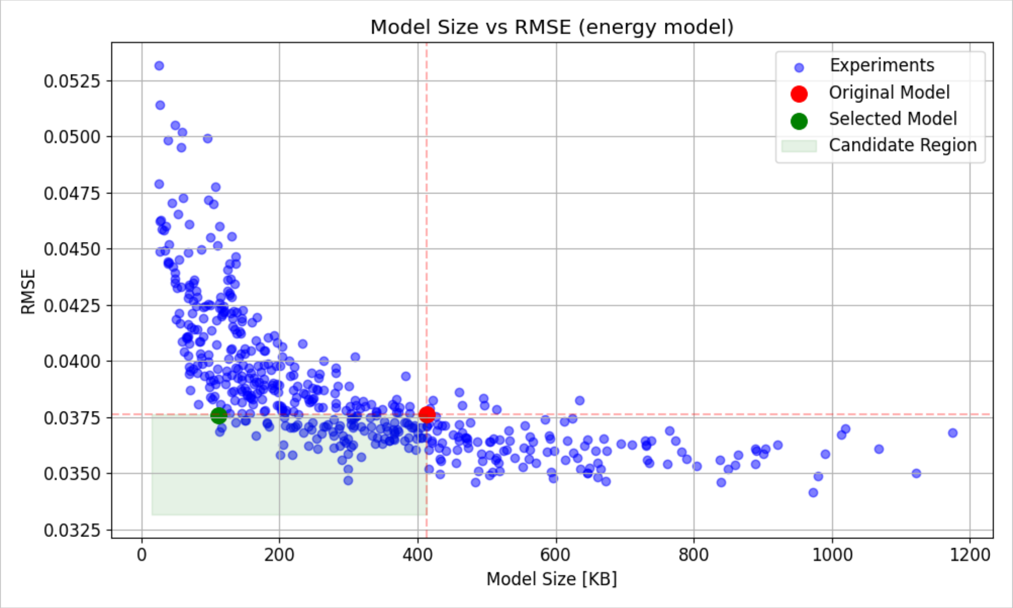
The frugal approach was applied in Pilot 5 in two different use case scenarios. The first scenario involves the machine learning algorithm for predicting the energy consumption. The technique applied to make the models more efficient is architecture search trying to find a model that is smaller yet with similar quality of the original one. The other scenario involves minifying tree-based machine learning models used in the pilot for forecasting environmental parameters. The approach was the same as before. The experiments code is available here:

https://gitlab.aeros-project.eu/pilots/pilot-5/forecasting-health-energy/-/tree/dev/src/app

During the architecture search the candidate models were constrained to be at least as good as the original ones (measured in RMSE and R-squared). All the selected frugal models were not only at least as good as the original ones but slightly better. The results of size measurements are as follows.

Model	**Original size [KB]**	**Reduced size [KB]**
Energy consumption	412	111
CO_2	783	115
PM_1	764	61
$PM_{2.5}$	757	32
PM_{10}	764	32
Temperature	761	55
Humidity	786	169

An example of a visualization of the energy model behaviour during the architecture search. Each blue dot is a test metric result (RMSE in this case) for a model trained with a set of hyperparameters. The green region contains models that are smaller and better than the original one. One can see that there are multiple candidates, however as per our objective we chose the smallest one that was at least as good as the original one.



KPI 1.6.4 Delivery of cookbook/good practices manual for explainable frugal AI near the edge (KVI-4.4)

Table 64: KPI 1.6.4 Delivery of cookbook/good practices manual for explainable frugal AI near the edge (KVI-4.4)

KPI ID number and partner resp.	**KPI 1.6.4**		
KPI Name	**Delivery of cookbook/good practices manual for explainable frugal AI near the edge (KVI-4.4)**		
Description	Description of good practices based on target value of application that would guide IoT developer to select the best approach.		
Motivation	Application of explainability and frugal techniques needs to be customized to a scenario being considered. Therefore, what is required is a set of good practices that would guide the IoT developer in selecting the best approach.		
Target value	>= 3 (pilot-specific)		
Prerequisites	Available results of evaluation of AI-based applications in the pilots and results of evaluation of aerOS deployment in the pilots. Results coming from all pilots in which AI was utilized, the most insightful will be selected for good practices preparation.		
aerOS components (task)	Explainability techniques, frugality techniques (T4.3)		
Evaluation means	Number of guidelines formulated, where a guideline is understood as a set of rules or remarks related to a specific feature or use case.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**

Measured value (% achieved)	***0***	***N/A***	***3 (100%)***
Outcome elaboration (M38)	The good practices formulated are based on the most insightful applications of frugal and/or explainable AI in aerOS deployments. AI was used in both internal scenarios (AI being part of aerOS components) and external scenarios (AI being part of a business use cases executed using aerOS deployment) that both can serve as reference guiding scenarios. Practical implementation of researched techniques exposed potential advantages but also challenges related to the efficient deployment and integration with other techniques. Use cases that were studied are: explainability in service allocation (HLO, grey-box allocator) and in energy consumption prediction (Pilot 5), frugality in AI-based self-optimization (aerOS self-optimization component) as an example scenario of potential anomaly-detection task deployed in aerOS, application of frugality to AI models in Pilot 5. The guidelines were described in a section of the aerOS Read The Docs (https://docs.aeros-project.eu/en/latest/ai_analytics/frugal/cookbook/index.html)		

KPI 1.6.5 Decentralized frugal AI techniques available

Table 65: KPI 1.6.5 Decentralized frugal AI techniques available

KPI ID number and partner resp.	**KPI 1.6.5**		
KPI Name	**Decentralized frugal AI techniques available**		
Description	Techniques applied to provide frugality to AI in aerOS or aerOS-based deployments where AI operations in restricted conditions need to be supported.		
Motivation	aerOS will operate on heterogenous Infrastructure Elements with both internal decision-making and potentially AI-related services. This heterogeneity may require for application of frugal techniques to enable effective operations.		
Target value	>= 3 techniques		
Prerequisites	N/A		
aerOS components (task)	Frugality techniques (T4.3)		
Evaluation means	Number of frugality techniques that have been evaluated for their effectiveness and for which applicability to aerOS scenarios was studied.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***2 techniques (50%)***	***3 techniques (100%)***

Outcome elaboration (M38)

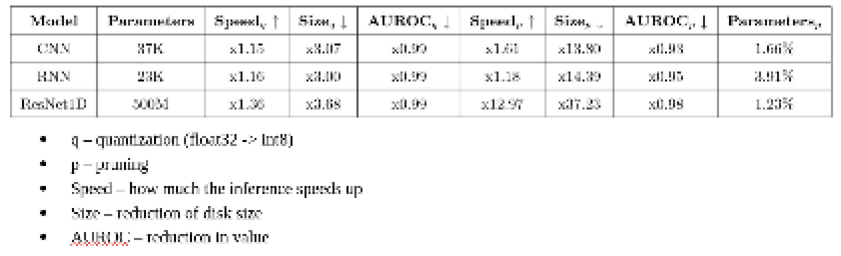
Three techniques have been implemented and experimented with: quantization, pruning and knowledge distillation.

Quantization was used to reduce the precision of model weights from 32-bit floating-point down to 8-bit integer representation, to decrease model size and speed up inference (while maintain reasonable results). Pruning was used to remove less important weights from a model to improve computational efficiency, without significantly impacting model quality. Knowledge distillation was applied to transfer knowledge from a larger (more complex) model to a smaller model during pruning.

In order to evaluate the behaviour of the mentioned methods experiments were conducted on a dataset with characteristics like datasets in aerOS deployment scenarios (i.e. monitoring readings with the aim of anomaly detection). The tests were run against CNN, RNN, and ResNet architectures.

Code is available in the GitLab repository https://gitlab.aeros-project.eu/wp4/t4.3/model-reduction-service.

Following is an extract from the AI model reduction experiments (knowledge distillation results is included in the pruning results):



Model	Parameters	Speed$_q$ ↑	Size$_q$ ↓	AUROC$_q$ ↓	Speed$_p$ ↑	Size$_p$ ↓	AUROC$_p$ ↓	Parameters$_p$
CNN	87K	x1.15	x3.07	x0.99	x1.61	x13.80	x0.98	1.66%
RNN	23K	x1.16	x3.00	x0.99	x1.18	x14.39	x0.95	3.91%
ResNet1D	300M	x1.26	x3.68	x0.99	x12.97	x37.23	x0.98	1.23%

- q – quantization (float32 -> int8)
- p – pruning
- Speed – how much the inference speeds up
- Size – reduction of disk size
- AUROC – reduction in value

Results were summarized in publication "Neural Network Compression for Resource Constrained Environments" (submitted for publication in Informatica journal).

KPI 1.6.6 AI explainability techniques available

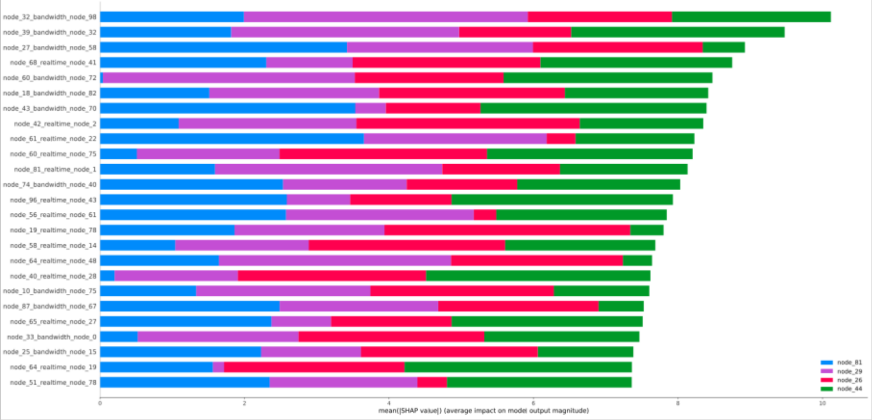
Table 66: KPI 1.6.6 AI explainability techniques available

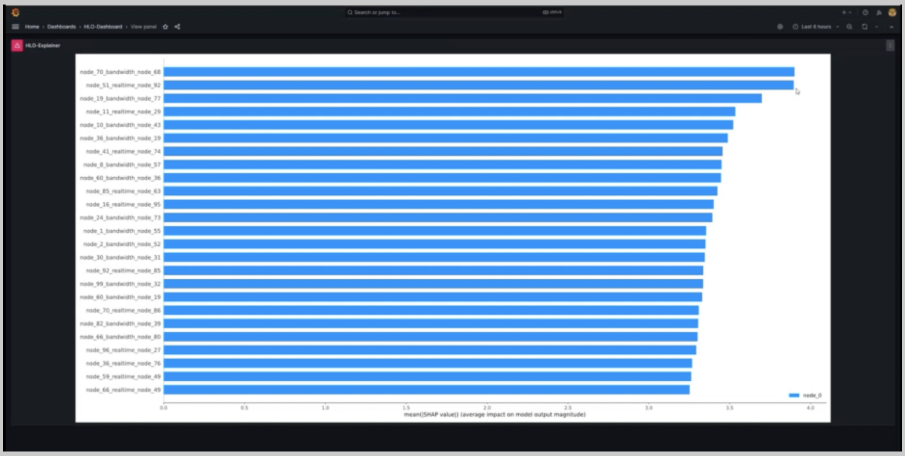
KPI ID number and partner resp.	**KPI 1.6.6**
KPI Name	**AI explainability techniques available**
Description	Techniques applied to provide interpretability/explainability in aerOS or aerOS-based deployments where AI interpretability/explainability can support the operations of the systems by enabling understanding of intelligent automatized decision-making.
Motivation	System with automatic intelligent decision-making should provide means to monitor and verify its behaviour.
Target value	>=2 techniques
Prerequisites	N/A
aerOS components (task)	Explainability techniques (T4.3)

<table>
<tr><td>Evaluation means</td><td colspan="3">Number of explainability techniques that have been evaluated for their effectiveness and for which applicability to aerOS scenarios was studied.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>1 technique – Shapley values for RL (50%)</i></td><td><i>2 techniques (100%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">
The first technique based on Shapley values was proposed for service allocation (internal aerOS use case). The method iteratively explains each decision made by the service allocator. The explainer works following a FaaS approach, which is an architectural pattern that can be used in other use cases.

The explainer is available in the GitLab repository: https://gitlab.aeros-project.eu/wp4/t4.3/explainability-service

This technique was tested as part of aerOS allocation process, e.g. in MVP2 demo. The following figures show examples of explanation of a single decision made by HLO that is shown in Embedded Analytics Tool.





The second technique was investigated as part of proposed alternative service allocation mechanism based on a grey-box model approach. Gray-box modelling is an architecture-based XAI technique where the model is designed to provide compartmentalized intermediate steps that can be validated separate from the overall model. This design results in explainability through tiered predictions, where because the inputs to further steps in the algorithm or model are separate and trained/designed for a specific purpose,
</td></tr>
</table>

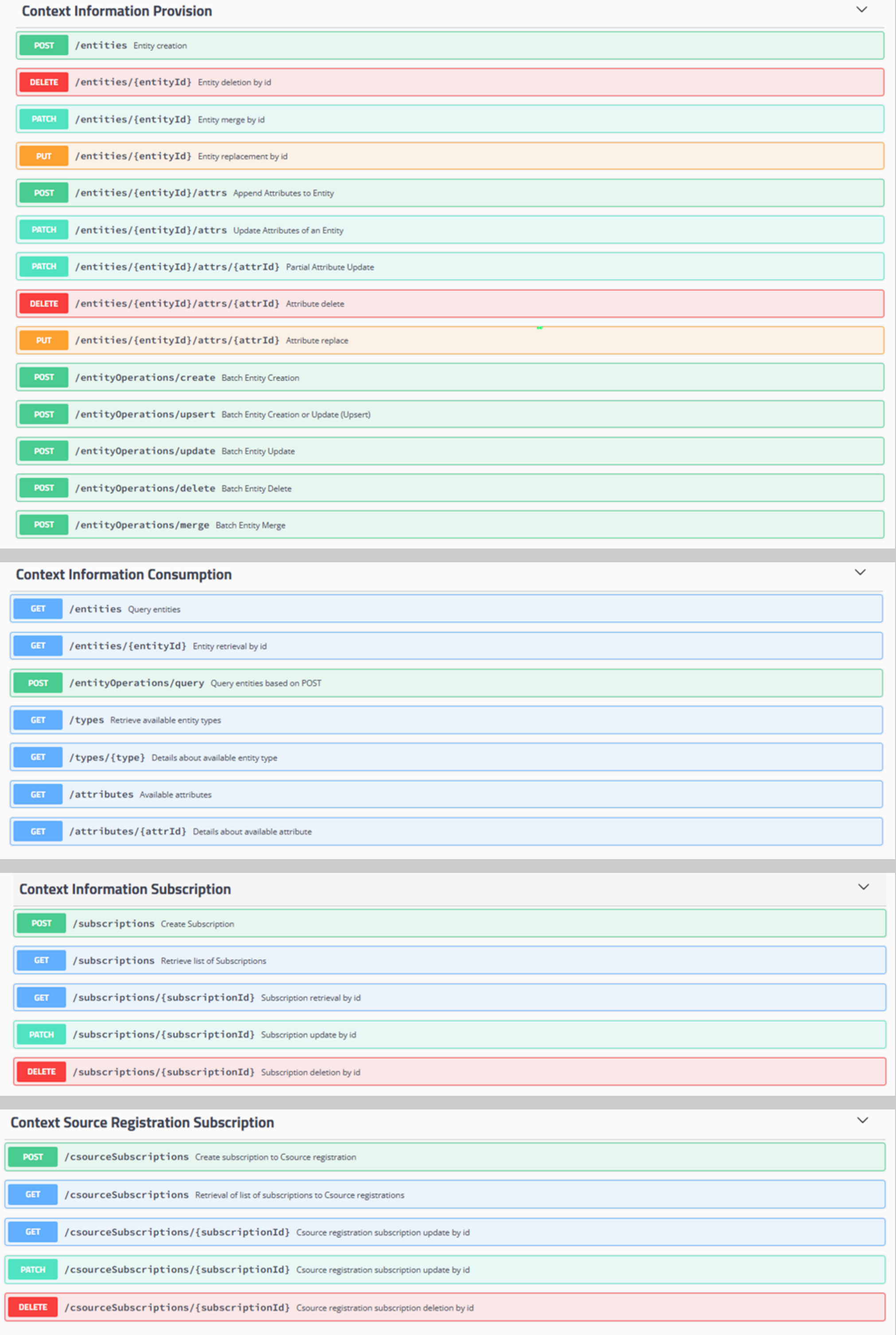
	model outputs can be traced and tracked back to which intermediate component made the prediction. aerOS components, such as self-scaling utilize this approach, generating independent predictions of future usage of several system resources. These predictions are then combined to produce an estimated number of replicas a Component may need to provide reliable service. The respective code is available at: https://gitlab.aeros-project.eu/wp3/t3.5/self-scaling Research published in: J. J. García et al., "Towards More Explainable and Traceable AI: Gray-boxed Design in a Case of Microservice Allocation," 2024 International Conference on INnovations in Intelligent SysTems and Applications (INISTA), Craiova, Romania, 2024, pp. 1-6, doi: 10.1109/INISTA62901.2024.10683838.

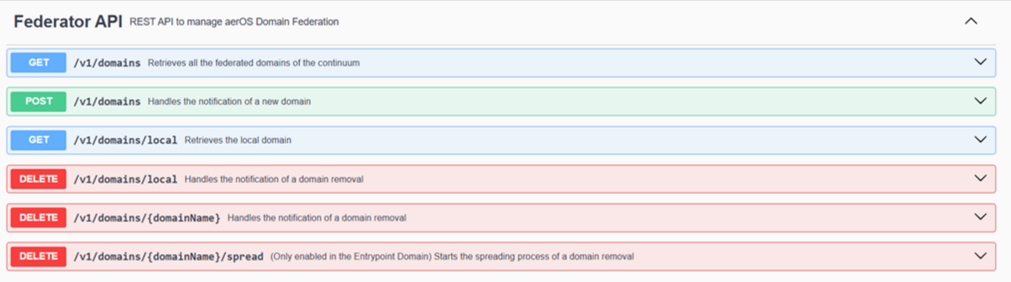
aerOS common API

KPI 1.7.1 % of aerOS core services exposed through OpenAPI

Table 67: KPI 1.7.1 % of aerOS core services exposed through OpenAPI

<table>
<tr><th>KPI ID number and partner resp.</th><th colspan="3">KPI 1.7.1</th></tr>
<tr><td>KPI Name</td><td colspan="3">% of aerOS core services exposed through OpenAPI</td></tr>
<tr><td>Description</td><td colspan="3">This KPI measures the proportion of aerOS's core services that are accessible through an OpenAPI. The goal is to ensure that a significant part of the system's functionality is available via well-defined and standardised interfaces.</td></tr>
<tr><td>Motivation</td><td colspan="3">Exposing services through OpenAPIs facilitates integration with other systems, encourages developer engagement, and supports a modular, scalable architecture. It enables third-party developers to easily connect with and build upon the aerOS platform, fostering innovation and expanding the system's capabilities.</td></tr>
<tr><td>Target value</td><td colspan="3">>50%</td></tr>
<tr><td>Prerequisites</td><td colspan="3">The exposed APIs of each aerOS component must be provided and can be reachable and interactive, providing the expected results.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">HLO (T3.3), Context Broker (T4.2), and Data Fabric (T4.2)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">The evaluation will involve identifying the total number of core aerOS services, involved in the project. At least 50% of these services must expose their components via Open API. The OpenAPI endpoints will be documented through screenshots for verification.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>0</i></td><td><i>25% of aerOS core services (50%)</i></td><td><i>88% of aerOS core services (176% of accomplishment)</i></td></tr>
</table>

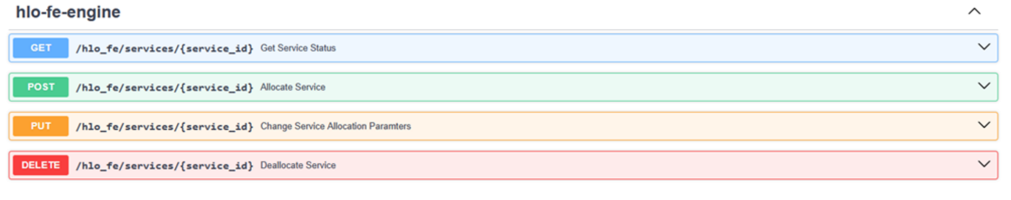
Outcome elaboration (M38)	The core services identified in aerOS are: HLO, Orion-LD Context Broker, IdM Keycloak, OpenLDAP, KrankedD API Gateway, Data Fabric, Self-* modules, Management Portal. Context Broker API:  **Context Information Provision** POST `/entities` Entity creation DELETE `/entities/{entityId}` Entity deletion by id PATCH `/entities/{entityId}` Entity merge by id PUT `/entities/{entityId}` Entity replacement by id POST `/entities/{entityId}/attrs` Append Attributes to Entity PATCH `/entities/{entityId}/attrs` Update Attributes of an Entity PATCH `/entities/{entityId}/attrs/{attrId}` Partial Attribute Update DELETE `/entities/{entityId}/attrs/{attrId}` Attribute delete PUT `/entities/{entityId}/attrs/{attrId}` Attribute replace POST `/entityOperations/create` Batch Entity Creation POST `/entityOperations/upsert` Batch Entity Creation or Update (Upsert) POST `/entityOperations/update` Batch Entity Update POST `/entityOperations/delete` Batch Entity Delete POST `/entityOperations/merge` Batch Entity Merge **Context Information Consumption** GET `/entities` Query entities GET `/entities/{entityId}` Entity retrieval by id POST `/entityOperations/query` Query entities based on POST GET `/types` Retrieve available entity types GET `/types/{type}` Details about available entity type GET `/attributes` Available attributes GET `/attributes/{attrId}` Details about available attribute **Context Information Subscription** POST `/subscriptions` Create Subscription GET `/subscriptions` Retrieve list of Subscriptions GET `/subscriptions/{subscriptionId}` Subscription retrieval by id PATCH `/subscriptions/{subscriptionId}` Subscription update by id DELETE `/subscriptions/{subscriptionId}` Subscription deletion by id **Context Source Registration Subscription** POST `/csourceSubscriptions` Create subscription to Csource registration GET `/csourceSubscriptions` Retrieval of list of subscriptions to Csource registrations GET `/csourceSubscriptions/{subscriptionId}` Csource registration subscription update by id PATCH `/csourceSubscriptions/{subscriptionId}` Csource registration subscription update by id DELETE `/csourceSubscriptions/{subscriptionId}` Csource registration subscription deletion by id Federator API:



Federator API REST API to manage aerOS Domain Federation

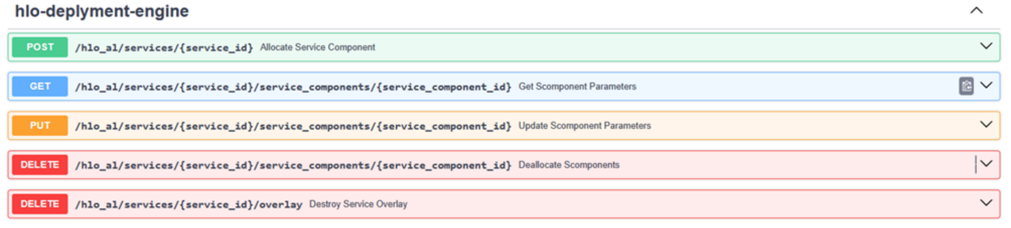
GET /v1/domains Retrieves all the federated domains of the continuum
POST /v1/domains Handles the notification of a new domain
GET /v1/domains/local Retrieves the local domain
DELETE /v1/domains/local Handles the notification of a domain removal
DELETE /v1/domains/{domainName} Handles the notification of a domain removal
DELETE /v1/domains/{domainName}/spread (Only enabled in the Entrypoint Domain) Starts the spreading process of a domain removal

HLO API:



hlo-fe-engine

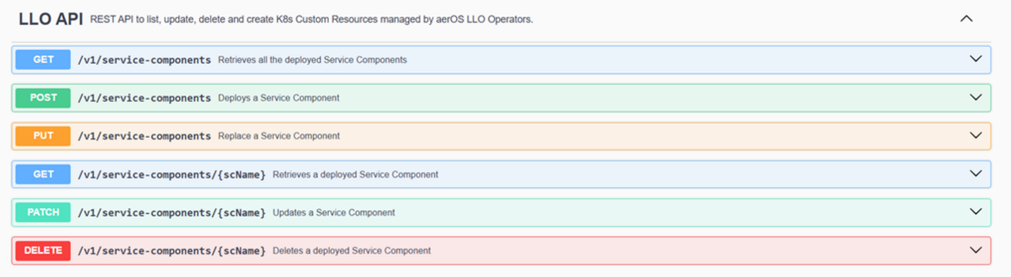
GET /hlo_fe/services/{service_id} Get Service Status
POST /hlo_fe/services/{service_id} Allocate Service
PUT /hlo_fe/services/{service_id} Change Service Allocation Paramters
DELETE /hlo_fe/services/{service_id} Deallocate Service



hlo-deplyment-engine

POST /hlo_al/services/{service_id} Allocate Service Component
GET /hlo_al/services/{service_id}/service_components/{service_component_id} Get Scomponent Parameters
PUT /hlo_al/services/{service_id}/service_components/{service_component_id} Update Scomponent Parameters
DELETE /hlo_al/services/{service_id}/service_components/{service_component_id} Deallocate Scomponents
DELETE /hlo_al/services/{service_id}/overlay Destroy Service Overlay

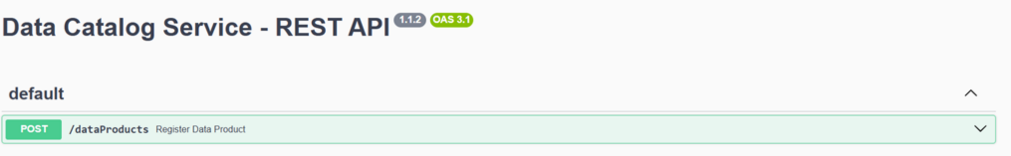
LLO API:



LLO API REST API to list, update, delete and create K8s Custom Resources managed by aerOS LLO Operators.

GET /v1/service-components Retrieves all the deployed Service Components
POST /v1/service-components Deploys a Service Component
PUT /v1/service-components Replace a Service Component
GET /v1/service-components/{scName} Retrieves a deployed Service Component
PATCH /v1/service-components/{scName} Updates a Service Component
DELETE /v1/service-components/{scName} Deletes a deployed Service Component

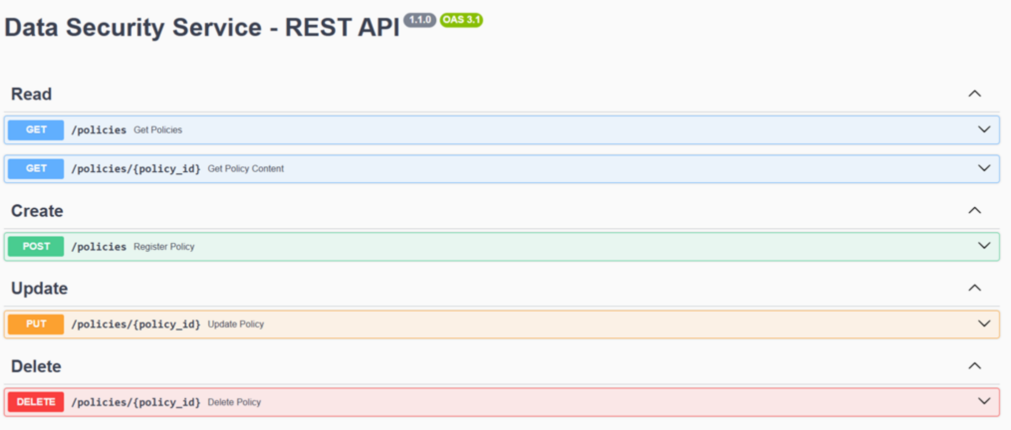
Data Fabric API:



Data Catalog Service - REST API 1.1.2 OAS 3.1

default

POST /dataProducts Register Data Product



Data Security Service - REST API 1.1.0 OAS 3.1

Read

GET /policies Get Policies
GET /policies/{policy_id} Get Policy Content

Create

POST /policies Register Policy

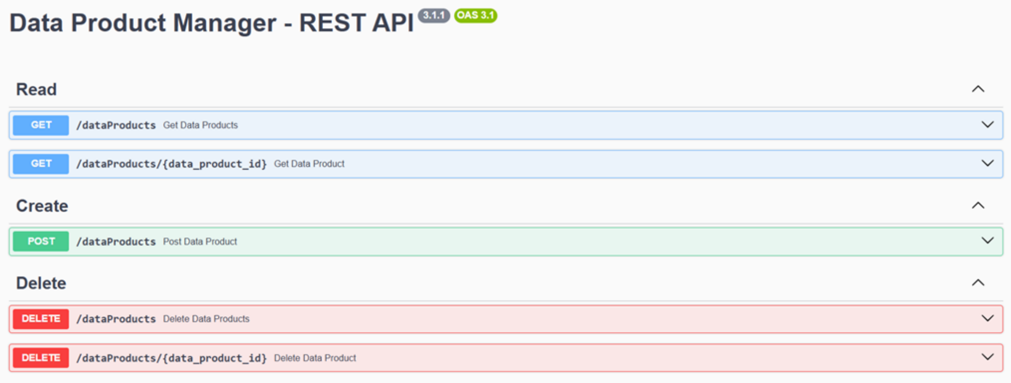
Update

PUT /policies/{policy_id} Update Policy

Delete

DELETE /policies/{policy_id} Delete Policy

DPM API:



Data Product Manager - REST API 3.1.1 OAS 3.1

Read

GET `/dataProducts` Get Data Products

GET `/dataProducts/{data_product_id}` Get Data Product

Create

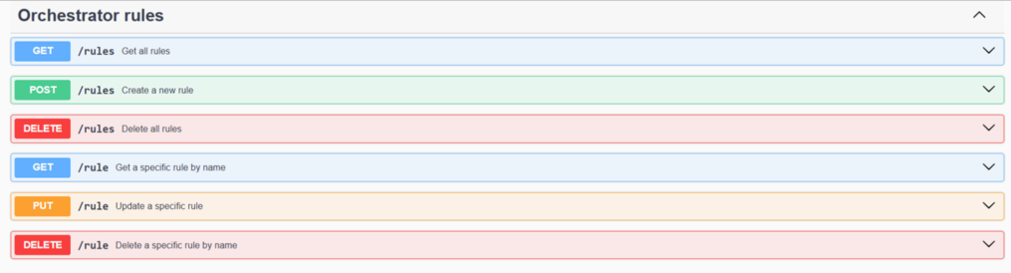
POST `/dataProducts` Post Data Product

Delete

DELETE `/dataProducts` Delete Data Products

DELETE `/dataProducts/{data_product_id}` Delete Data Product

Self-Capabilities API:



Orchestrator rules

GET `/rules` Get all rules

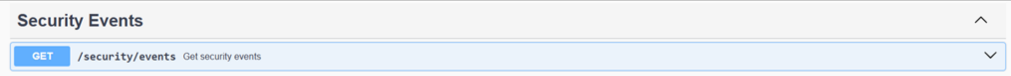
POST `/rules` Create a new rule

DELETE `/rules` Delete all rules

GET `/rule` Get a specific rule by name

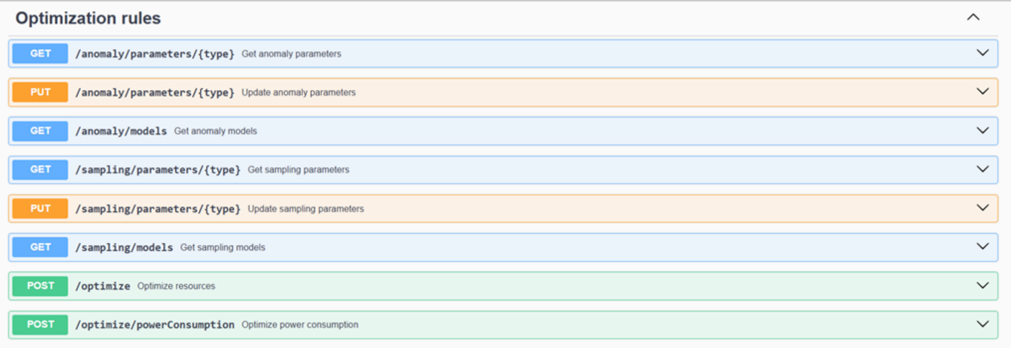
PUT `/rule` Update a specific rule

DELETE `/rule` Delete a specific rule by name



Security Events

GET `/security/events` Get security events



Optimization rules

GET `/anomaly/parameters/{type}` Get anomaly parameters

PUT `/anomaly/parameters/{type}` Update anomaly parameters

GET `/anomaly/models` Get anomaly models

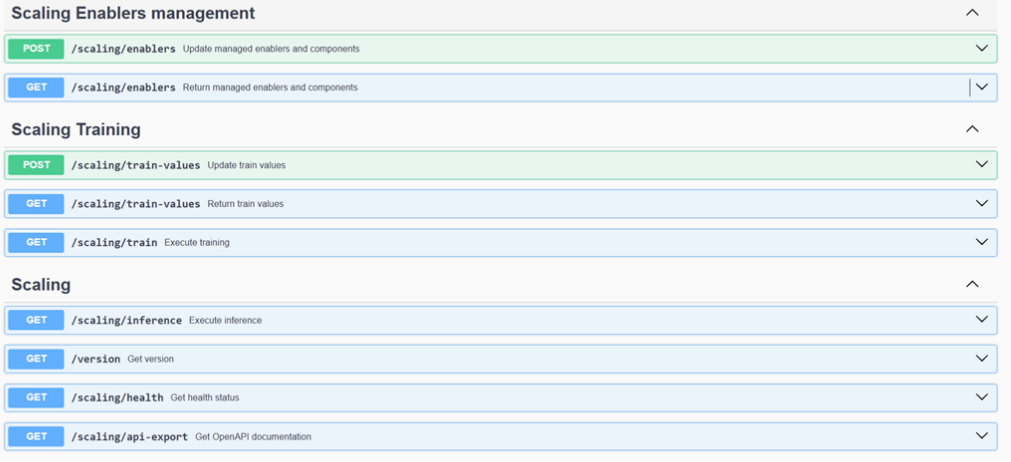
GET `/sampling/parameters/{type}` Get sampling parameters

PUT `/sampling/parameters/{type}` Update sampling parameters

GET `/sampling/models` Get sampling models

POST `/optimize` Optimize resources

POST `/optimize/powerConsumption` Optimize power consumption



Scaling Enablers management

POST `/scaling/enablers` Update managed enablers and components

GET `/scaling/enablers` Return managed enablers and components

Scaling Training

POST `/scaling/train-values` Update train values

GET `/scaling/train-values` Return train values

GET `/scaling/train` Execute training

Scaling

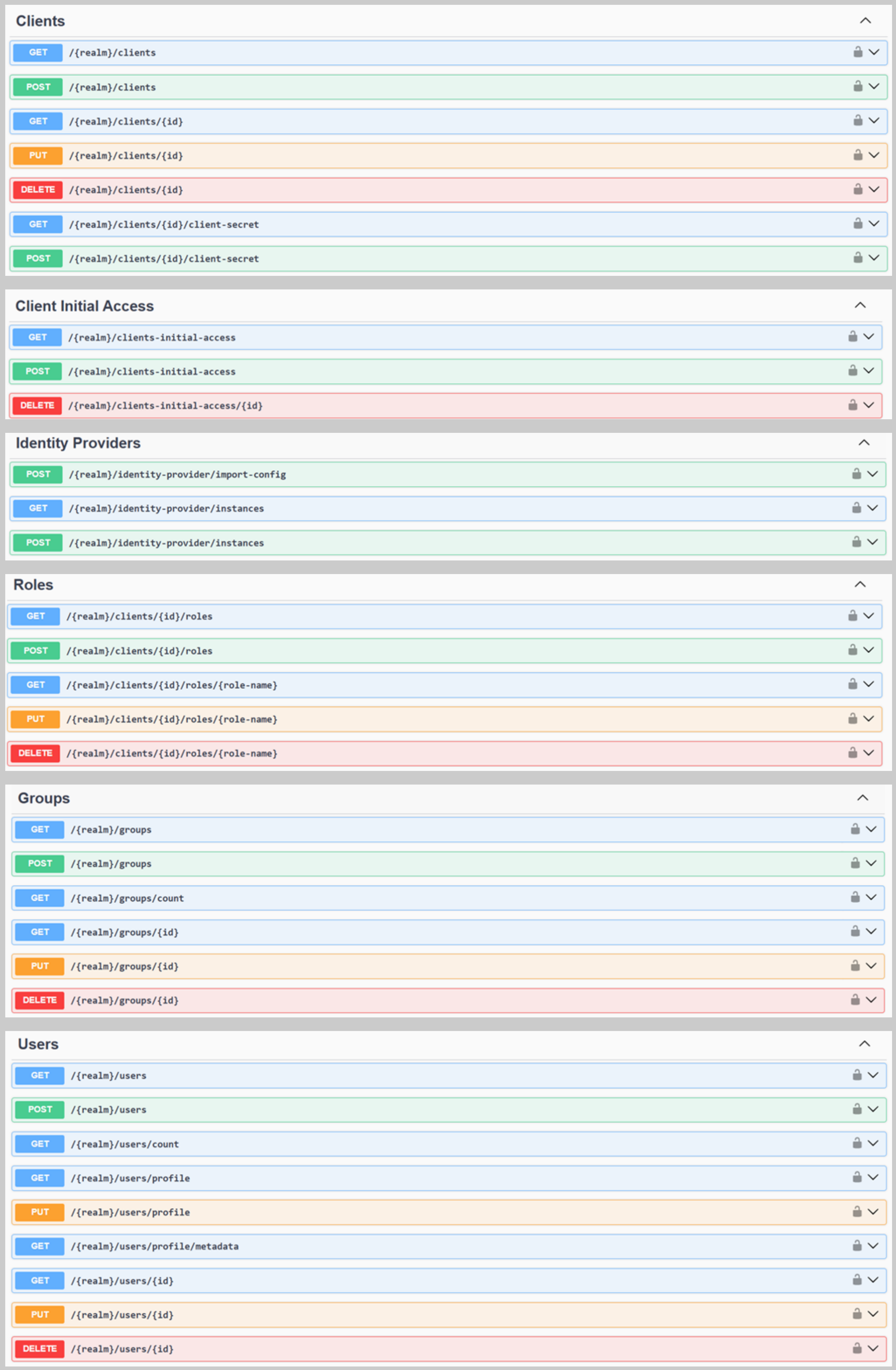
GET `/scaling/inference` Execute inference

GET `/version` Get version

GET `/scaling/health` Get health status

GET `/scaling/api-export` Get OpenAPI documentation

IdM API:



Clients

GET /{realm}/clients
POST /{realm}/clients
GET /{realm}/clients/{id}
PUT /{realm}/clients/{id}
DELETE /{realm}/clients/{id}
GET /{realm}/clients/{id}/client-secret
POST /{realm}/clients/{id}/client-secret

Client Initial Access

GET /{realm}/clients-initial-access
POST /{realm}/clients-initial-access
DELETE /{realm}/clients-initial-access/{id}

Identity Providers

POST /{realm}/identity-provider/import-config
GET /{realm}/identity-provider/instances
POST /{realm}/identity-provider/instances

Roles

GET /{realm}/clients/{id}/roles
POST /{realm}/clients/{id}/roles
GET /{realm}/clients/{id}/roles/{role-name}
PUT /{realm}/clients/{id}/roles/{role-name}
DELETE /{realm}/clients/{id}/roles/{role-name}

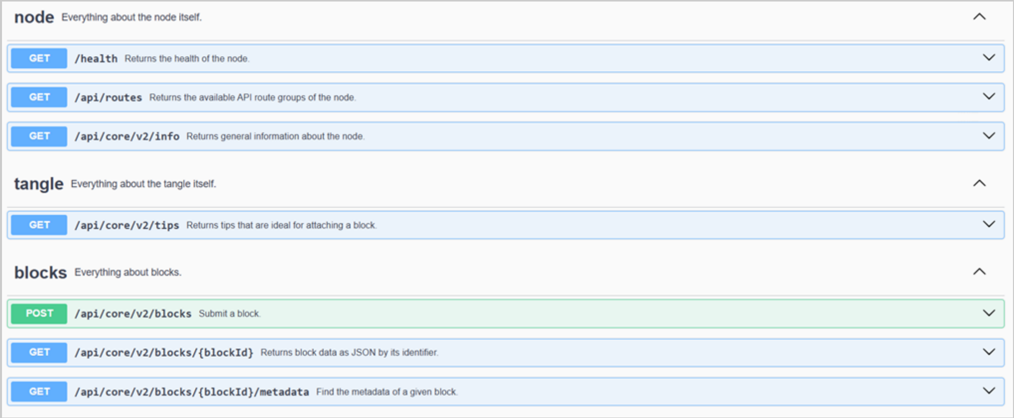
Groups

GET /{realm}/groups
POST /{realm}/groups
GET /{realm}/groups/count
GET /{realm}/groups/{id}
PUT /{realm}/groups/{id}
DELETE /{realm}/groups/{id}

Users

GET /{realm}/users
POST /{realm}/users
GET /{realm}/users/count
GET /{realm}/users/profile
PUT /{realm}/users/profile
GET /{realm}/users/profile/metadata
GET /{realm}/users/{id}
PUT /{realm}/users/{id}
DELETE /{realm}/users/{id}

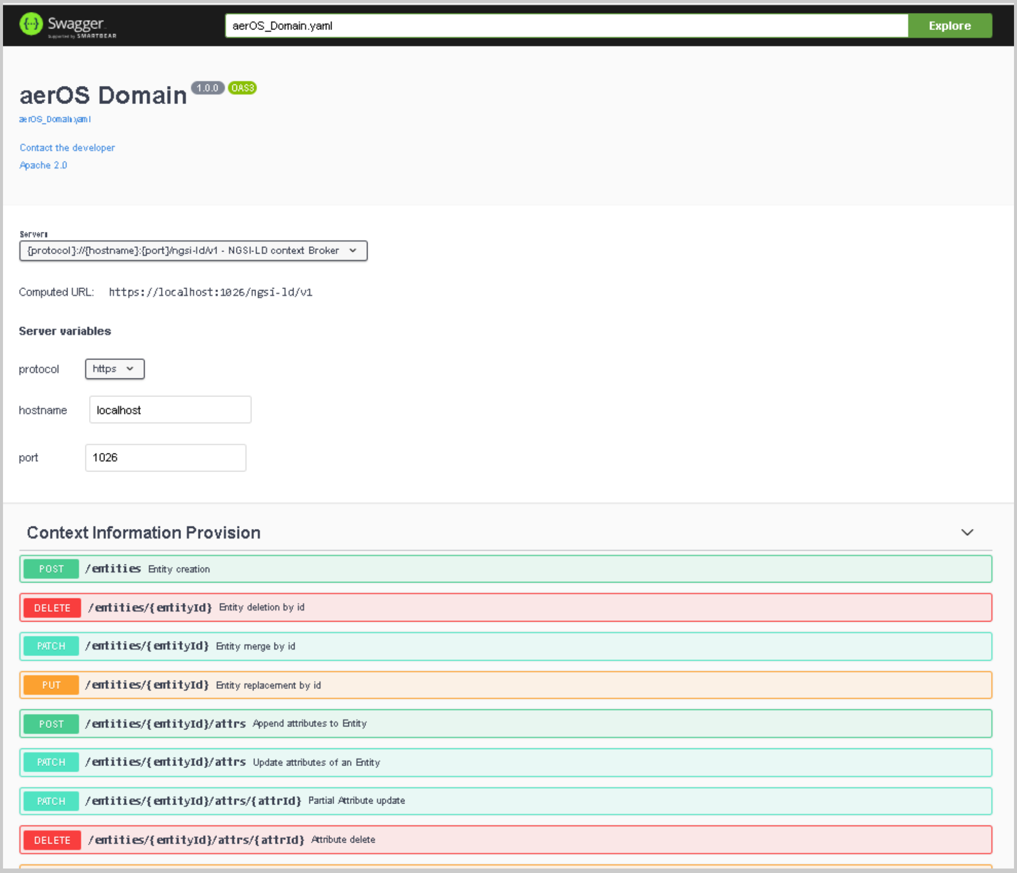
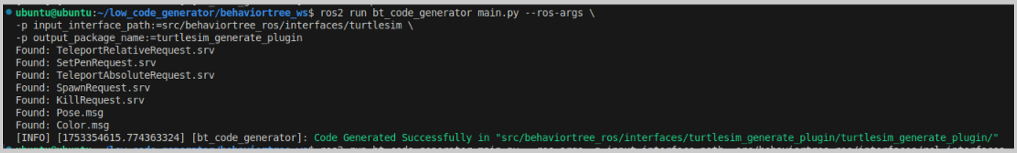
IOTA API:

	 **node** Everything about the node itself. GET `/health` Returns the health of the node. GET `/api/routes` Returns the available API route groups of the node. GET `/api/core/v2/info` Returns general information about the node. **tangle** Everything about the tangle itself. GET `/api/core/v2/tips` Returns tips that are ideal for attaching a block. **blocks** Everything about blocks. POST `/api/core/v2/blocks` Submit a block. GET `/api/core/v2/blocks/{blockId}` Returns block data as JSON by its identifier. GET `/api/core/v2/blocks/{blockId}/metadata` Find the metadata of a given block.

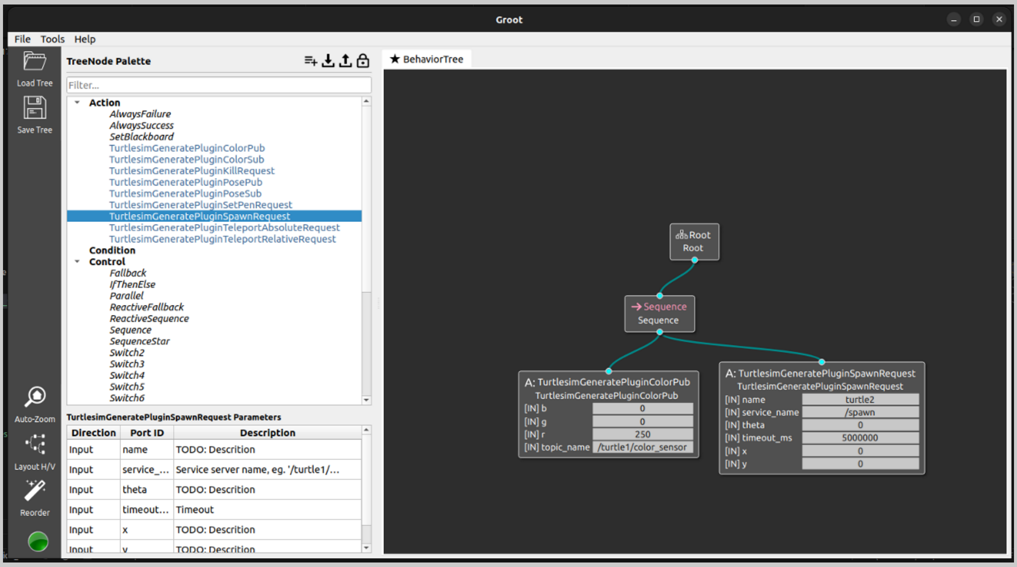
KPI 1.7.2 OpenAPI UIs for documenting APIs and generating code

Table 68: KPI 1.7.2 OpenAPI UIs for documenting APIs and generating code

<table>
<tr><td>KPI ID number and partner resp.</td><td colspan="3">KPI 1.7.2</td></tr>
<tr><td>KPI Name</td><td colspan="3">OpenAPI UIs for documenting APIs and generating code</td></tr>
<tr><td>Description</td><td colspan="3">These UIs will make it easier for developers to understand and work with the aerOS APIs. By providing clear documentation and tools for code generation, developers can more efficiently integrate their services with aerOS, reducing development time and potential errors</td></tr>
<tr><td>Motivation</td><td colspan="3">The motivation behind this KPI is to enhance developer experience and productivity by providing comprehensive and accessible documentation of the aerOS APIs. With well-documented APIs and integrated code generation tools, developers can quickly grasp the functionality and implementation details of aerOS services. This leads to faster integration, fewer development errors, and a more streamlined development process. Ultimately, it supports the goal of creating a robust, developer-friendly ecosystem around aerOS.</td></tr>
<tr><td>Target value</td><td colspan="3">2</td></tr>
<tr><td>Prerequisites</td><td colspan="3">The aerOS domain OpenAPI must be provided</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">HLO (T3.3), Context Broker (T4.2), and Data Fabric (T4.2).</td></tr>
<tr><td>Evaluation means</td><td colspan="3">The evaluation of this KPI will involve several steps. First, it will require the identification of the main aerOS exposed APIs that will be documented using OpenAPI UIs. The success of these UIs will be measured by their completeness and usability. Evidence of successful implementation will be provided through access to the UIs, user guides, and examples of generated code.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
</table>

Measured value (% achieved)	***0***	***1 (50%)***	***2 (100%)***
Outcome elaboration (M38)	First, one OpenAPI UI has been successfully implemented, providing comprehensive documentation for the aerOS components (KPI 1.7.1). Below is a screenshot of the Swagger page for the exposed services, showcasing the OpenAPI documentation:  aerOS_Domain.yaml aerOS Domain aerOS_Domain.yaml Contact the developer Apache 2.0 Servers {protocol}://{hostname}:{port}/ngsi-ld/v1 - NGSI-LD context Broker Computed URL: `https://localhost:1026/ngsi-ld/v1` Server variables protocol: https hostname: localhost port: 1026 Context Information Provision POST `/entities` Entity creation DELETE `/entities/{entityId}` Entity deletion by id PATCH `/entities/{entityId}` Entity merge by id PUT `/entities/{entityId}` Entity replacement by id POST `/entities/{entityId}/attrs` Append attributes to Entity PATCH `/entities/{entityId}/attrs` Update attributes of an Entity PATCH `/entities/{entityId}/attrs/{attrId}` Partial Attribute update DELETE `/entities/{entityId}/attrs/{attrId}` Attribute delete Second, based on the API documentation a low code generator has been created. This is capable to generate Node-RED, behavior tree or scratch blocks based on the API definitions: 			

```
ubuntu@ubuntu:~/low_code_generator/behaviortree_ws$ ros2 run bt_code_generator main.py --ros-args \
 -p input_interface_path:=src/behaviortree_ros/interfaces/turtlesim \
 -p output_package_name:=turtlesim_generate_plugin
 Found: TeleportRelativeRequest.srv
 Found: SetPenRequest.srv
 Found: TeleportAbsoluteRequest.srv
 Found: SpawnRequest.srv
 Found: KillRequest.srv
 Found: Pose.msg
 Found: Color.msg
 [INFO] [1753354615.774363324] [bt_code_generator]: Code Generated Successfully in "src/behaviortree_ros/interfaces/turtlesim_generate_plugin/turtlesim_generate_plugin/"
```

TurtlesimGeneratePluginSpawnRequest Parameters

Direction	Port ID	Description
Input	name	TODO: Descrition
Input	service_...	Service server name, eg. '/turtle1/...
Input	theta	TODO: Descrition
Input	timeout...	Timeout
Input	x	TODO: Descrition
Input	y	TODO: Descrition

A: TurtlesimGeneratePluginColorPub
TurtlesimGeneratePluginColorPub

[IN] b	0
[IN] g	0
[IN] r	250
[IN] topic_name	/turtle1/color_sensor

A: TurtlesimGeneratePluginSpawnRequest
TurtlesimGeneratePluginSpawnRequest

[IN] name	turtle2
[IN] service_name	/spawn
[IN] theta	0
[IN] timeout_ms	5000000
[IN] x	0
[IN] y	0

KPI 1.7.3 Create Protocol Buffers definition for intra-orchestration module communication

Table 69: KPI 1.7.3 Create Protocol Buffers definition for intra-orchestration module communication

KPI ID number and partner resp.	**KPI 1.7.3**
KPI Name	**Create Protocol Buffers definition for intra-orchestration module communication**
Description	This KPI aims to develop three Protocol Buffers definitions to facilitate communication between different modules within the aerOS orchestration layer. Protocol Buffers is a method of serializing structured data
Motivation	Using Protocol Buffers enhances the efficiency and reliability of data interchange between modules. This approach ensures consistent, lightweight, and backward-compatible communication, crucial for maintaining the robustness and scalability of the orchestration layer.
Target value	3
Prerequisites	Identification and documentation of all modules within the aerOS orchestration layer that require Kakfa communication.
aerOS components (task)	HLO (T3.3)
Evaluation means	The evaluation will involve identifying and documenting the different modules within the aerOS orchestration layer that require Kafka communication. The team developed Protocol Buffers definitions for each identified communication pathway. The definitions are documented and it is confirmed that they are consistent and lightweight. Evidence of successful implementation will be provided below through code repositories and integration examples.

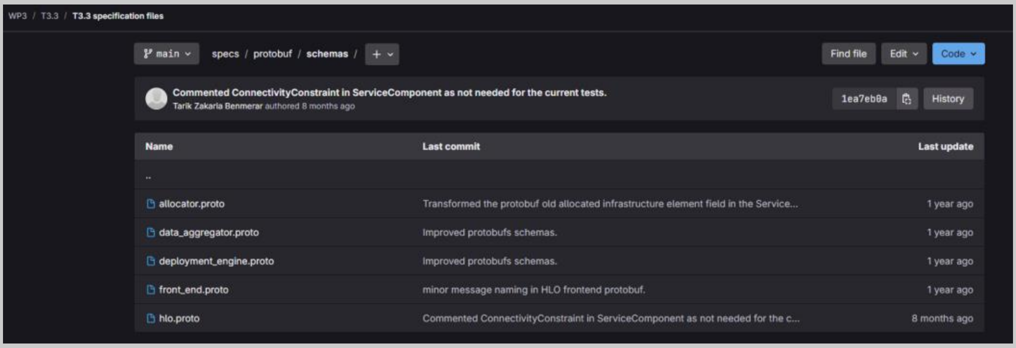
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*N/A*	*4 (133%)*	*5 (166%)*

Outcome elaboration (M38)

Five Protocol Buffers definitions have been successfully identified and documented, exceeding the original target. These definitions facilitate communication among the following components:

- HLO Allocator
- HLO Data Aggregator
- HLO Deployment Engine
- HLO Front End

Five Protocol Buffers definitions have been developed, tested, and integrated, ensuring they are consistent, lightweight, and backward compatible. Below is a screenshot of the code repository showcasing the developed Protocol Buffers definitions.



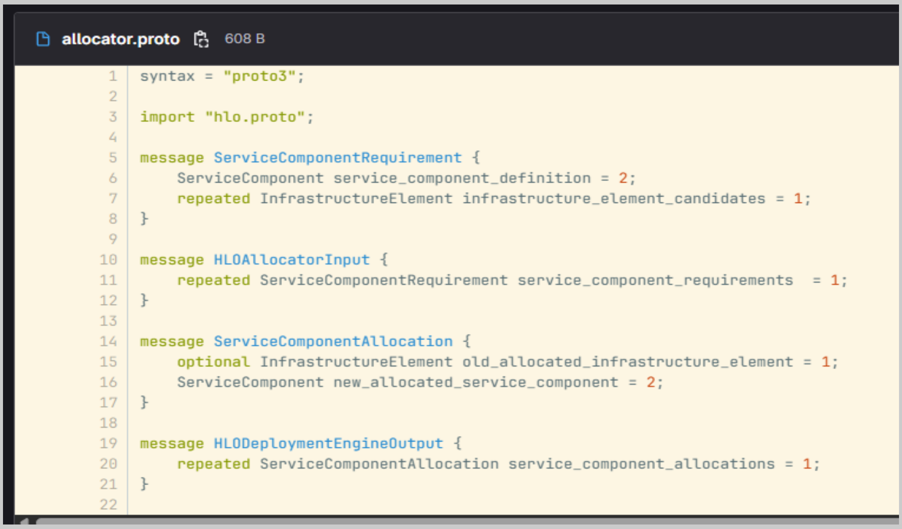
WP3 / T3.3 / T3.3 specification files

main / specs / protobuf / schemas /

Commented ConnectivityConstraint in ServiceComponent as not needed for the current tests.
Tarik Zakaria Benmerar authored 8 months ago

1ea7eb0a

Name	Last commit	Last update
..		
allocator.proto	Transformed the protobuf old allocated infrastructure element field in the Service...	1 year ago
data_aggregator.proto	Improved protobufs schemas.	1 year ago
deployment_engine.proto	Improved protobufs schemas.	1 year ago
front_end.proto	minor message naming in HLO frontend protobuf.	1 year ago
hlo.proto	Commented ConnectivityConstraint in ServiceComponent as not needed for the c...	8 months ago

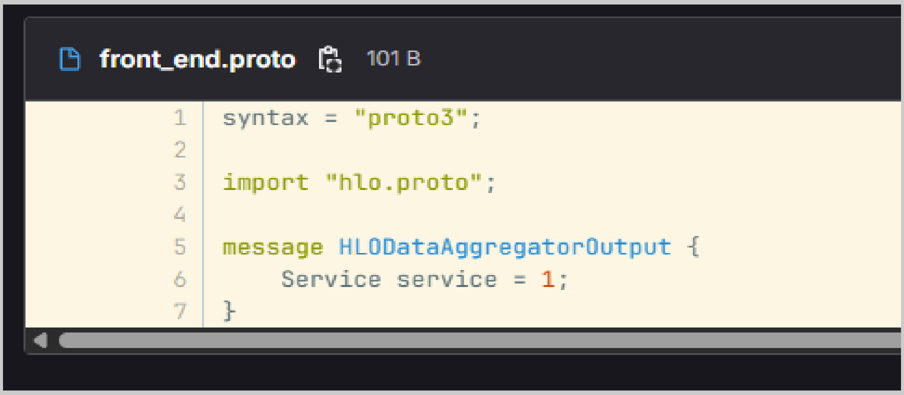


allocator.proto 608 B

```
syntax = "proto3";

import "hlo.proto";

message ServiceComponentRequirement {
    ServiceComponent service_component_definition = 2;
    repeated InfrastructureElement infrastructure_element_candidates = 1;
}

message HLOAllocatorInput {
    repeated ServiceComponentRequirement service_component_requirements  = 1;
}

message ServiceComponentAllocation {
    optional InfrastructureElement old_allocated_infrastructure_element = 1;
    ServiceComponent new_allocated_service_component = 2;
}

message HLODeploymentEngineOutput {
    repeated ServiceComponentAllocation service_component_allocations = 1;
}

```

```
syntax = "proto3";

import "hlo.proto";

message HLOFEInput {
    Service service = 1;
}

message ServiceComponentRequirement {
    ServiceComponent service_component_definition = 2;
    repeated InfrastructureElement infrastructure_element_candidates = 1;
}

message HLOAllocatorOutput {
    repeated ServiceComponentRequirement service_component_requirements  = 1;
}
```

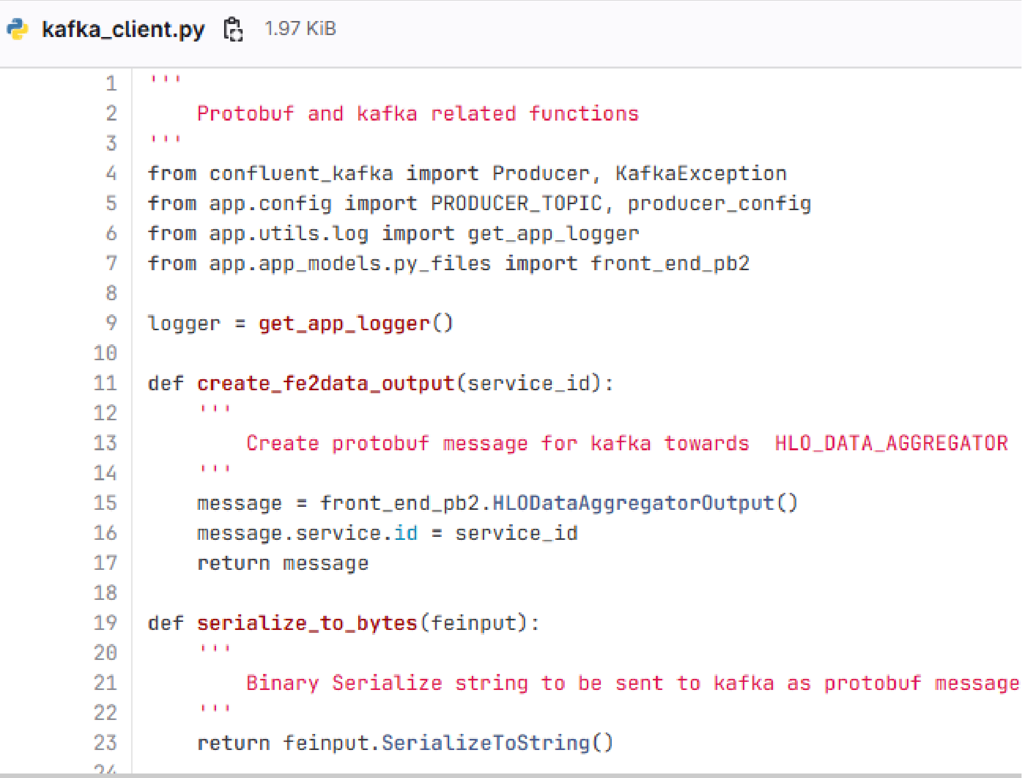
```
syntax = "proto3";

import "hlo.proto";


message ServiceComponentAllocation {
    InfrastructureElement old_allocated_infrastructure_element = 1;
    ServiceComponent new_allocated_service_component = 2;
}

message HLODeploymentEngineInput {
    repeated ServiceComponentAllocation service_component_allocations = 1;
}
```



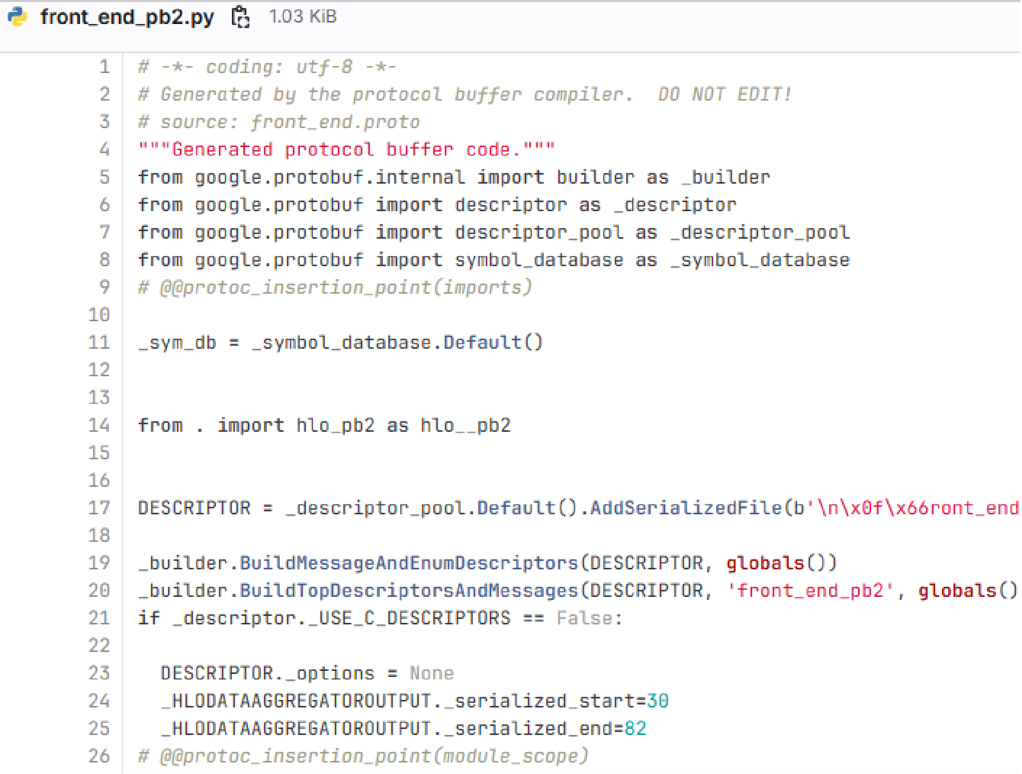
```
syntax = "proto3";

import "hlo.proto";

message HLODataAggregatorOutput {
    Service service = 1;
}
```

```

syntax = "proto3";

message Domain {
    string id = 1;
}

message Service {
    string id = 1;
}

message LowLevelOrchestrator {
    string id = 1;
    string orchestration_type = 3;
    Domain domain = 2;
}

message InfrastructureElement {
    string id = 1;
    float total_ram = 2;
    float cpu_cores = 3;
    float avg_power_consumption = 4;
    bool real_time_capable = 5;
    float current_ram = 6;
    float current_cpu_usage = 7;
    float current_power_consumption = 8;
    Domain domain = 9;
    LowLevelOrchestrator low_level_orchestrator = 10;
    string hostname = 11;
    string container_technology = 12;
}

message Port {
    int32 number = 1;
}

message ServiceComponent {
    string id = 1;
    string image = 2;
    repeated Port ports = 3;
    Service service = 4;
    ServiceComponentConstraints service_component_constraints = 5;
    optional InfrastructureElement infrastructure_element = 6;
    repeated ServiceComponentKeyValue cliArgs = 7;
    repeated ServiceComponentKeyValue envVars = 8;
    //repeated ConnectivityConstraint connectivity_constraints = 9;
    bool exposePorts = 9;
}

message ServiceComponentConstraints {
    map<string, float> constraints = 1;
}
message ServiceComponentKeyValue {
    string key = 1;
    string value = 2;
}
message ConnectivityConstraint {
    string service_component_id = 1;
    float bandwidth = 2;
    uint32 latency  = 3;
    uint32 jitter  = 4;
}

```

Finally, the following screenshots depict some integration examples in the HLO Frontend code. First, some automatically generated code by the protobuf compiler for Python, and then part of the code of the Kafka client.



```
'''
    Protobuf and kafka related functions
'''
from confluent_kafka import Producer, KafkaException
from app.config import PRODUCER_TOPIC, producer_config
from app.utils.log import get_app_logger
from app.app_models.py_files import front_end_pb2

logger = get_app_logger()

def create_fe2data_output(service_id):
    '''
        Create protobuf message for kafka towards  HLO_DATA_AGGREGATOR
    '''
    message = front_end_pb2.HLODataAggregatorOutput()
    message.service.id = service_id
    return message

def serialize_to_bytes(feinput):
    '''
        Binary Serialize string to be sent to kafka as protobuf message
    '''
    return feinput.SerializeToString()
```

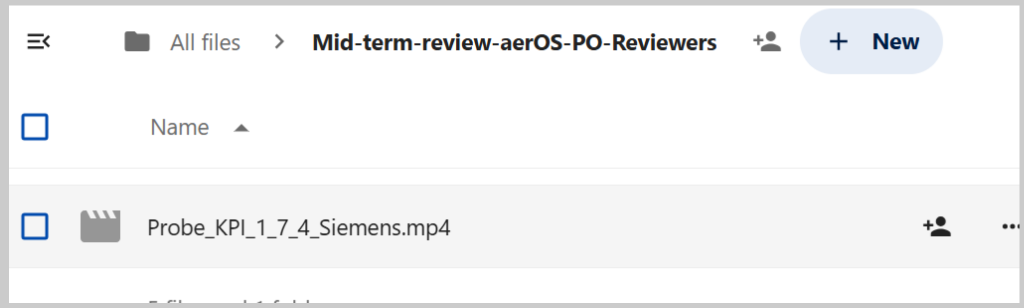


```
# -*- coding: utf-8 -*-
# Generated by the protocol buffer compiler.  DO NOT EDIT!
# source: front_end.proto
"""Generated protocol buffer code."""
from google.protobuf.internal import builder as _builder
from google.protobuf import descriptor as _descriptor
from google.protobuf import descriptor_pool as _descriptor_pool
from google.protobuf import symbol_database as _symbol_database
# @@protoc_insertion_point(imports)

_sym_db = _symbol_database.Default()


from . import hlo_pb2 as hlo__pb2


DESCRIPTOR = _descriptor_pool.Default().AddSerializedFile(b'\n\x0f\x66ront_end

_builder.BuildMessageAndEnumDescriptors(DESCRIPTOR, globals())
_builder.BuildTopDescriptorsAndMessages(DESCRIPTOR, 'front_end_pb2', globals()
if _descriptor._USE_C_DESCRIPTORS == False:

  DESCRIPTOR._options = None
  _HLODATAAGGREGATOROUTPUT._serialized_start=30
  _HLODATAAGGREGATOROUTPUT._serialized_end=82
# @@protoc_insertion_point(module_scope)
```

KPI 1.7.4 Reduce time to deploy service functions by non-technical team members using low code tool integrations

Table 70: KPI 1.7.4 Reduce time to deploy service functions by non-technical team members using low code tool integrations

KPI ID number and partner resp.	**KPI 1.7.4**		
KPI Name	**Reduce time to deploy service functions by non-technical team members using low code tool integrations**		
Description	This KPI focuses on decreasing the time required for non-technical team members to deploy service functions within aerOS by over 40%, leveraging low-code tool integrations.		
Motivation	Facilitating faster deployment of service functions by non-technical staff can significantly enhance operational efficiency. By integrating low-code tools, aerOS can democratize the deployment process, enabling a broader range of team members to contribute to service development and management, thus accelerating the project lifecycle and reducing dependency on technical specialists.		
Target value	Improvement of >40% over a baseline		
Prerequisites	Have a protocol compatible with the low-code tool to be used. ROS, MQTT and web-sockets are currently supported.		
aerOS components (task)	Low-code Tools and AsyncAPI (T3.2)		
Evaluation means	The evaluation of this KPI will involve several key steps. Initially, a baseline measurement of the time currently required for non-technical team members to deploy service functions without low-code tools will be established. Following this, appropriate low-code tools will be identified and integrated into the aerOS environment. The deployment time will then be measured and compared to the baseline to determine the percentage reduction achieved.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***40 seconds***	***20 seconds (120%)***	***20 seconds (120% improvment)***
Outcome elaboration (M38)	To date, the integration and use of aerOS Low-code Tools and AsyncAPI have been tested in a pilot with Siemens, focusing on reprogramming the behaviour of multiple Automated Guided Vehicles (AGVs). This pilot involved two main AGV skills: navigation from point to point and lifting boxes. Traditionally, non-technical users needed 40 seconds to reconfigure an AGV using PLC methods, despite the HMI available for the PLC. With AsyncAPI and low-code tools, skills were defined in an AsyncAPI document, and plugin blocks for the low-code tool (behaviour trees) were generated. This allowed non-technical users to reprogram the AGV in runtime using an intuitive web GUI, reducing the time to 20 seconds and making training much shorter. This progress demonstrates our commitment to enhancing operational efficiency by empowering non-technical team members to deploy and manage service functions more quickly. A video showing the time differences between doing the same task using and not using low code tools could be found here: https://nextcloud.aeros-project.eu/apps/files/files/292399?dir=/Mid-term-review-aerOS-PO-Reviewers&openfile=true		

	 All files > **Mid-term-review-aerOS-PO-Reviewers** + New ☐ Name ▲ ☐ Probe_KPI_1_7_4_Siemens.mp4

aerOS management framework

KPI 1.8.1 # of federated domains in all aerOS continuums

Table 71: KPI 1.8.1 # of federated domains in all aerOS continuums

KPI ID number and partner resp.	**KPI 1.8.1**		
KPI Name	**# of federated domains in all aerOS continuums**		
Description	This KPI will quantify the total number of aerOS domains that have been created (and have been functionally deployed) in the project. It refers to all the domains that will have been federated. It gathers both the domains coming from specific pilots and those created for development or integration purposes, as long as they are federated with other domains.		
Motivation	The motivation of this KPI is to represent the soundness of the design of "domain" concept. The goal is to be able to demonstrate that the theoretical design of IEs inside domains (i.e., designed and created by system administrators based on different criteria, such as topological sense, geographical proximity, container management framework technology, among others) are well translated into real deployments.		
Target value	15 (total domains) / (in) 8 (continuums)		
Prerequisites	A set of fully functional continuums (one for each pilot/testbench) composed by (at least) a functional domain with all the aerOS Basic Services running and all the IEs properly set up.		
aerOS components (task)	aerOS Management Portal (T4.6), aerOS Federator (T4.6), Context Broker (T4.2 and T4.6), aerOS AAA (T3.4), API Gateway (T4.2). Despite using the portal to check the number of federated domains, the action is performed in the Orion-LD instances of the entry point domains of each continuum		
Evaluation means	Total functional domains will be counted and indicated as the KPI. This will be endorsed by being able to connect to aerOS portal of the different continuums and checking the number of existing (and functional) domains in each.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**

Measured value (% achieved)	***0 (nothing before aerOS)***	***6 domains (40%) in 2 continuums (25%).***	***47 domains (313%) in 17 continuums (213%)***

Outcome elaboration (M38)

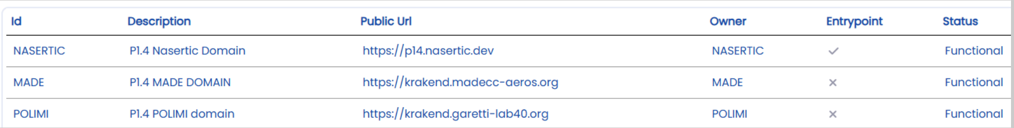
In M24 (D5.5), the installation of aerOS was being performed in the pilots following the first version of the installation guide (D5.2), so their continuums weren't fully available at that point of time. Currently, the installation of aerOS has been successfully performed in all the pilots and in open call projects. Therefore, these continuums have been taken into consideration for this KPI:

- Pilot 1.1, composed of 2 domains.

Domains list

Id	Description	Public Url	Owner	Entrypoint	Status
P11Domain1	This is the first example domain from SIPBB	https://p11.nasertic.dev	SIPBB	✓	Functional
P11Domain2	Pilot 1.1 - SIPBB Domain	https://146.4.102.10:30815	SIPBB	×	Functional

- Pilot 1.2, with 2 domains
- Pilot 1.3, composed of 3 domains
- Pilot 1.4, with 3 domains



Id	Description	Public Url	Owner	Entrypoint	Status
NASERTIC	P1.4 Nasertic Domain	https://p14.nasertic.dev	NASERTIC	✓	Functional
MADE	P1.4 MADE DOMAIN	https://krakend.madecc-aeros.org	MADE	×	Functional
POLIMI	P1.4 POLIMI domain	https://krakend.garetti-lab40.org	POLIMI	×	Functional

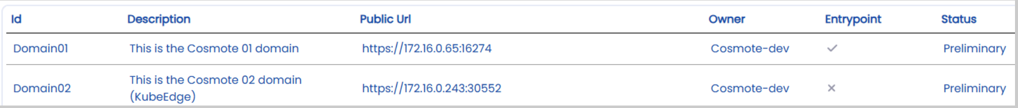
- Pilot 2, with 4 domains

Id	Description	Public Url	Owner	Entrypoint	Status
aeros-central	This is entrypoint domain running in cloud environment, not for running workloads.	https://entrypoint.pilot2.aeros-project.eu	CloudFerro	✓	Preliminary
aeros1	This is domain nr 1 for running workloads located in a container.	https://aeros1.aeros.staging.intra.cloudferro.com	CloudFerro	×	Preliminary
aeros2	This is domain nr 2 for running workloads located in a container.	https://aeros2.aeros.staging.intra.cloudferro.com	CloudFerro	×	Preliminary
aeros-electrum	Electrum Domain	https://electrum-aeros.com	Electrum	×	Preliminary

- Pilot 3, with 2 domains
- Pilot 4, composed of 3 domains

Id	Description	Public Url	Owner	Entrypoint	Status
pilot4_domain1	Pilot4 domain1	https://aeros-pilot4.prodevelopataws.com	pilot4	✓	Preliminary
eurogate_limassol	EUROGATE CTL domain	http://10.10.11.6:31583	pilot4	×	Functional
cutdomain	This is a new domain, which is not the entrypoint	https://cutdom.aeros.cut.ac.cy	CUT	×	Preliminary

- Pilot 5, with 2 domains



Id	Description	Public Url	Owner	Entrypoint	Status
Domain01	This is the Cosmote 01 domain	https://172.16.0.65:16274	Cosmote-dev	✓	Preliminary
Domain02	This is the Cosmote 02 domain (KubeEdge)	https://172.16.0.243:30552	Cosmote-dev	×	Preliminary

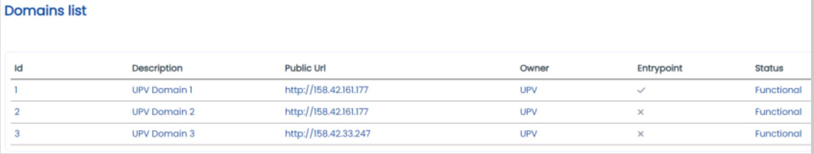
- Open Calls (Round 1): 7 continuums composed of a total of 9 domains
- Open Calls (Round 2): 7 continuums composed of a total of 12 domains

- aerOS MVP, composed of 2 domains

Domains list

Id	Description	Public Url	Owner	Entrypoint	Status
CloudFerro	CloudFerro Domain	https://cf-mvp-domain.aeros-project.eu	CloudFerro	✓	Functional
Mobile	Mobile domain for the demo	https://mobile-domain.aeros-project.eu	UPV	×	Functional
NCSRD	NCSRD aerOS MVP Domain	https://ncsrd-mvp-domain.aeros-project.eu	NCSRD	×	Functional

- Internal testing environment of the UPV, composed of 3 domains



Domains list

Id	Description	Public Url	Owner	Entrypoint	Status
1	UPV Domain 1	http://158.42.161.177	UPV	✓	Functional
2	UPV Domain 2	http://158.42.161.177	UPV	×	Functional
3	UPV Domain 3	http://158.42.33.247	UPV	×	Functional

The followed procedure consists of accessing to the domains section of the Management Portal of each continuum to count the number of federated and available (with *Functional* status) domains. Moreover, additional proofs (images of real equipment, etc.) are provided to improve the value of this KPI.

KPI 1.8.2 # of continuum functionalities available and operational through the Management Portal

Table 72: KPI 1.8.2 # of continuum functionalities available and operational through the Management Portal

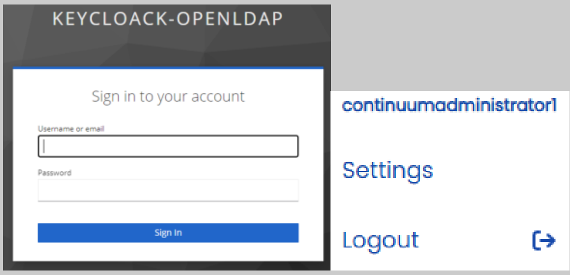
KPI ID number and partner resp.	**KPI 1.8.2**
KPI Name	**# of continuum functionalities available and operational through the Management Portal**
Description	This KPI will quantify the total number of available functionalities to operate the resources and services operating the continuum that can be managed via the aerOS Management Portal. The various functionalities will be associated to resources (e.g., creation of IEs, removal of IEs from the continuum, domains enabling…), users management (e.g., creation, roles assignation, …), services (e.g., deployment, monitoring…) and/or data (i.e., inspecting the Data Fabric…).
Motivation	All continuum management functionalities will be available via APIs. This is managed by T3.2 and will serve as the basis for aerOS continuum establishment. However, some of them will also be available via an UI (in the aerOS portal). The motivation of this KPI is to represent how many of them will be usable via the UI, enlarging the human-oriented capacity of the Meta-OS (in this case, for system administrators' configurations, etc).
Target value	10

<table>
<tr><td>Prerequisites</td><td colspan="3">The aerOS entry point domain with a <i>Functional</i> status. It means that the continuum is composed by (at least) a functional domain with all the aerOS Basic Services running and all the IEs properly set up.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">The KPI targets the aerOS Management Portal (T4.6), but other components are needed: Orion-LD (T4.2 and T4.6), aerOS AAA (T3.4), HLO (T3.3), Self-* modules (T3.5), Data Fabric (T4.2) and Entrypoint Balancer (T4.6)..</td></tr>
<tr><td>Evaluation means</td><td colspan="3">At the end of the project, functionalities included in the portal will be endorsed via showcasing the possibilities of usage (screenshots, videos, demos).</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M36 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>0</i></td><td><i>6 (60%)</i></td><td><i>13 (130%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">

Direct visual and functional testing by potential end users (aerOS technical developers and pilot users) after accessing the portal.

This is the final list of the functionalities available through the Management Portal:

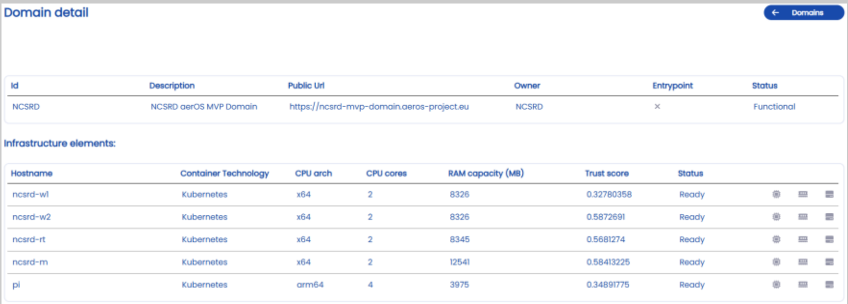
- Access with various user profiles validated by AAA.



- Observing all IEs and domains in the form of a table. Enhanced with trust score and container technology

Domains list

<table>
<tr><th>Id</th><th>Description</th><th>Public Url</th><th>Owner</th><th>Entrypoint</th><th>Status</th></tr>
<tr><td>CloudFerro</td><td>CloudFerro Domain</td><td>https://cf-mvp-domain.aeros-project.eu</td><td>CloudFerro</td><td>✓</td><td>Functional</td></tr>
<tr><td>Mobile</td><td>Mobile domain for the demo</td><td>https://mobile-domain.aeros-project.eu</td><td>UPV</td><td>×</td><td>Functional</td></tr>
<tr><td>NCSRD</td><td>NCSRD aerOS MVP Domain</td><td>https://ncsrd-mvp-domain.aeros-project.eu</td><td>NCSRD</td><td>×</td><td>Functional</td></tr>
</table>



Domain detail

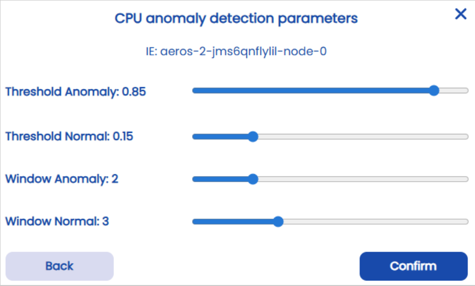
<table>
<tr><th>Id</th><th>Description</th><th>Public Url</th><th>Owner</th><th>Entrypoint</th><th>Status</th></tr>
<tr><td>NCSRD</td><td>NCSRD aerOS MVP Domain</td><td>https://ncsrd-mvp-domain.aeros-project.eu</td><td>NCSRD</td><td>×</td><td>Functional</td></tr>
</table>

Infrastructure elements:

<table>
<tr><th>Hostname</th><th>Container Technology</th><th>CPU arch</th><th>CPU cores</th><th>RAM capacity (MB)</th><th>Trust score</th><th>Status</th></tr>
<tr><td>ncsrd-w1</td><td>Kubernetes</td><td>x64</td><td>2</td><td>8326</td><td>0.32780358</td><td>Ready</td></tr>
<tr><td>ncsrd-w2</td><td>Kubernetes</td><td>x64</td><td>2</td><td>8326</td><td>0.5872691</td><td>Ready</td></tr>
<tr><td>ncsrd-rt</td><td>Kubernetes</td><td>x64</td><td>2</td><td>8345</td><td>0.5681274</td><td>Ready</td></tr>
<tr><td>ncsrd-m</td><td>Kubernetes</td><td>x64</td><td>2</td><td>12541</td><td>0.58413225</td><td>Ready</td></tr>
<tr><td>pi</td><td>Kubernetes</td><td>arm64</td><td>4</td><td>3975</td><td>0.34891775</td><td>Ready</td></tr>
</table>

</td></tr>
</table>

- Configuration of the Self-optimization parameters for each IE



CPU anomaly detection parameters

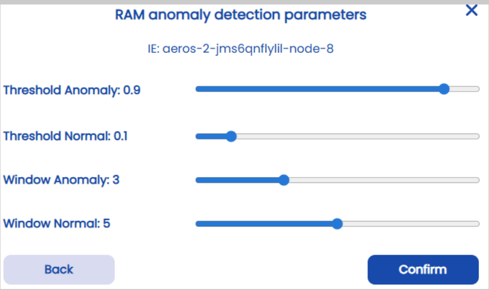
IE: aeros-2-jms6qnflylil-node-0

Threshold Anomaly: 0.85

Threshold Normal: 0.15

Window Anomaly: 2

Window Normal: 3



RAM anomaly detection parameters

IE: aeros-2-jms6qnflylil-node-8

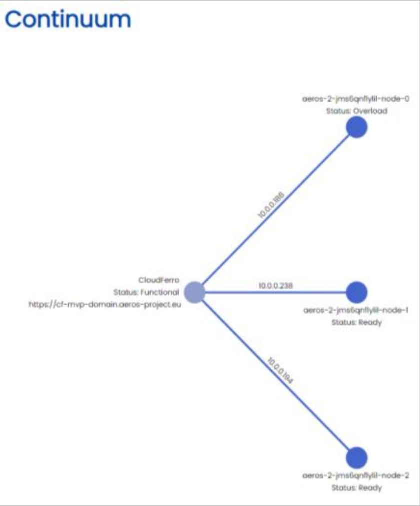
Threshold Anomaly: 0.9

Threshold Normal: 0.1

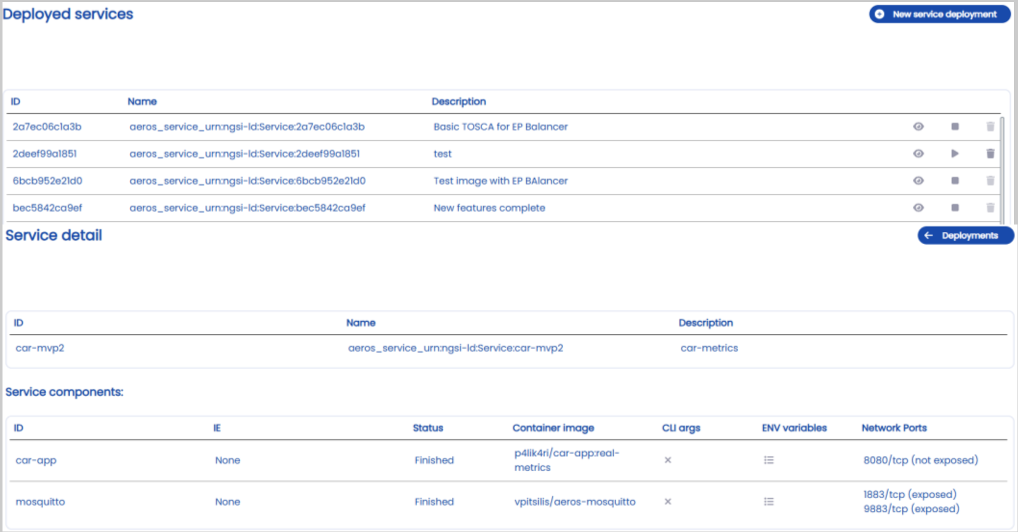
Window Anomaly: 3

Window Normal: 5

- Visualizing the topology of the continuum in the form of a graph.



- Observing all deployed services and their underlying components in the form of a table. Enhanced with the information of exposed network ports



Deployed services

ID	Name	Description
2a7ec06c1a3b	aeros_service_urn:ngsi-ld:Service:2a7ec06c1a3b	Basic TOSCA for EP Balancer
2deef99a1851	aeros_service_urn:ngsi-ld:Service:2deef99a1851	test
6bcb952e21d0	aeros_service_urn:ngsi-ld:Service:6bcb952e21d0	Test image with EP BAlancer
bec5842ca9ef	aeros_service_urn:ngsi-ld:Service:bec5842ca9ef	New features complete

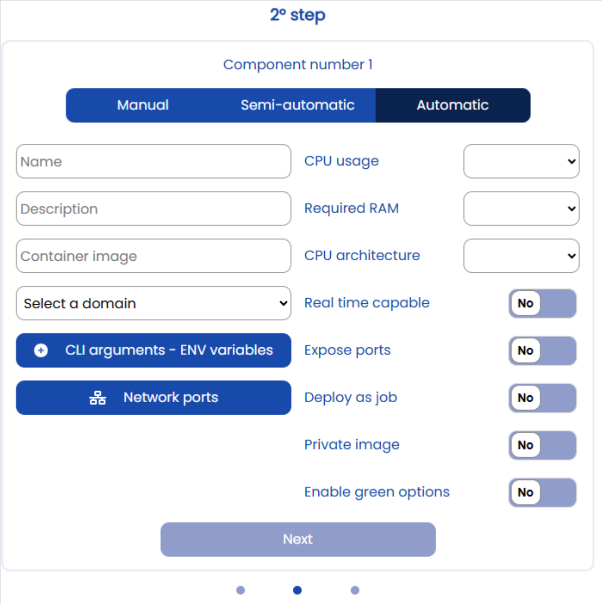
Service detail

ID	Name	Description
car-mvp2	aeros_service_urn:ngsi-ld:Service:car-mvp2	car-metrics

Service components:

ID	IE	Status	Container image	CLI args	ENV variables	Network Ports
car-app	None	Finished	p4lik4ri/car-app:real-metrics	×	☰	8080/tcp (not exposed)
mosquitto	None	Finished	vpitsilis/aeros-mosquitto	×	☰	1883/tcp (exposed) 9883/tcp (exposed)

- Describing a service to be orchestrated using a guided form. Final version with the 3 orchestration modes (manual, semi-automatic, automatic).



2º step

Component number 1

Name

Description

Container image

Select a domain

CPU usage

Required RAM

CPU architecture

Real time capable: No

Expose ports: No

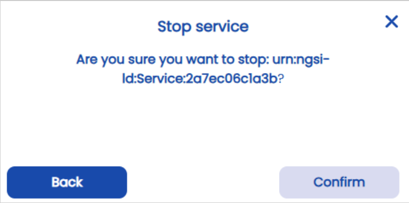
Deploy as job: No

Private image: No

Enable green options: No

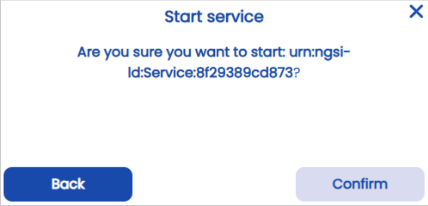
- Commissioning the orchestration of a service (connection with Entrypoint Balancer and HLO).

- Managing the complete lifecycle of the deployed services: stop, redeployment and permanent deletion.



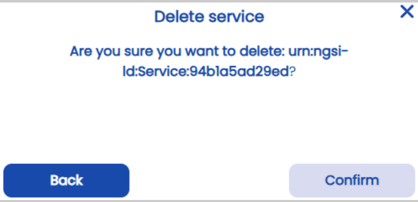
Stop service

Are you sure you want to stop: urn:ngsi-Id:Service:2a7ec06c1a3b?



Start service

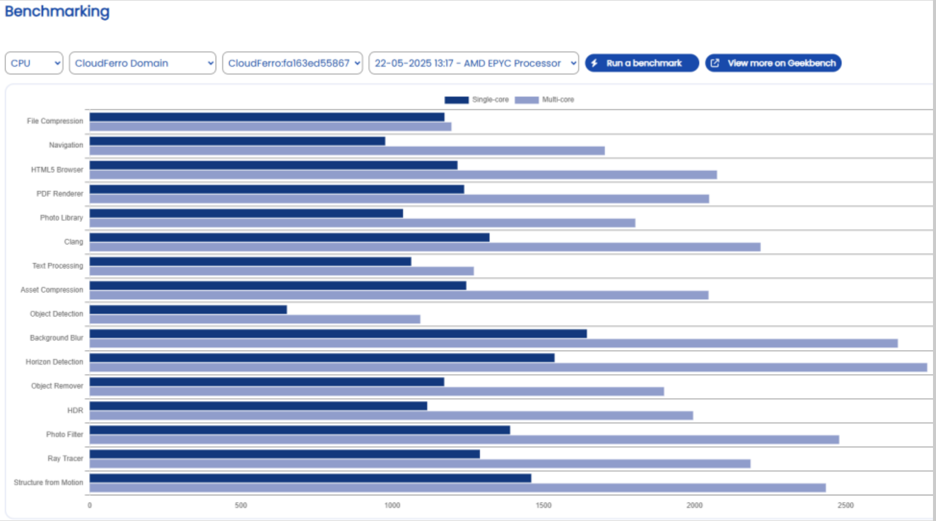
Are you sure you want to start: urn:ngsi-Id:Service:8f29389cd873?

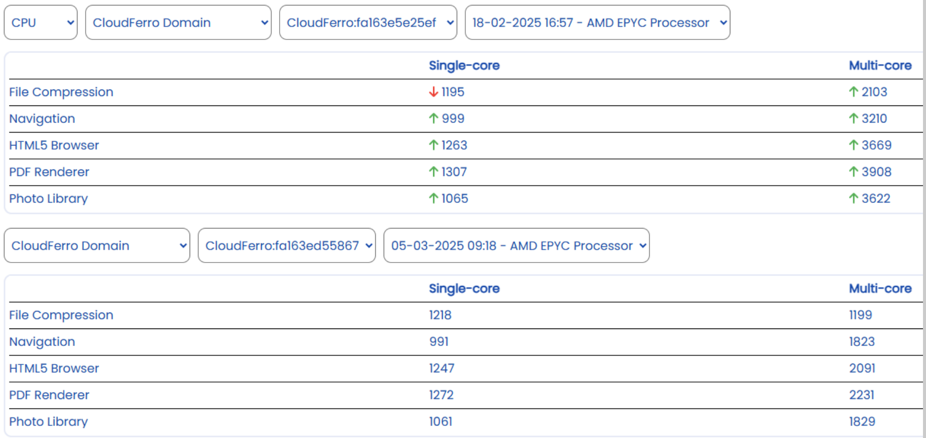


Delete service

Are you sure you want to delete: urn:ngsi-Id:Service:94b1a5ad29ed?

- Benchmarking tool: run benchmarking tests in a specific IE and display the results of the already performed benchmarking tests (CPU and network) in each IE, along with a comparison tool.





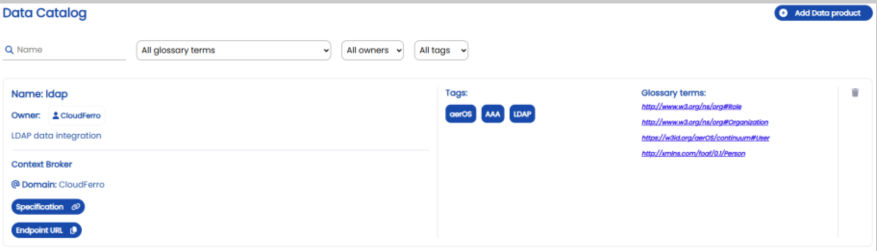
CPU | CloudFerro Domain | CloudFerro:fa163e5e25ef | 18-02-2025 16:57 - AMD EPYC Processor

	Single-core	Multi-core
File Compression	↓ 1195	↑ 2103
Navigation	↑ 999	↑ 3210
HTML5 Browser	↑ 1263	↑ 3669
PDF Renderer	↑ 1307	↑ 3908
Photo Library	↑ 1065	↑ 3622

CloudFerro Domain | CloudFerro:fa163ed55867 | 05-03-2025 09:18 - AMD EPYC Processor

	Single-core	Multi-core
File Compression	1218	1199
Navigation	991	1823
HTML5 Browser	1247	2091
PDF Renderer	1272	2231
Photo Library	1061	1829

- List of published and available Data Products in the Data Catalog service.



Name: ldap

Owner: CloudFerro

LDAP data integration

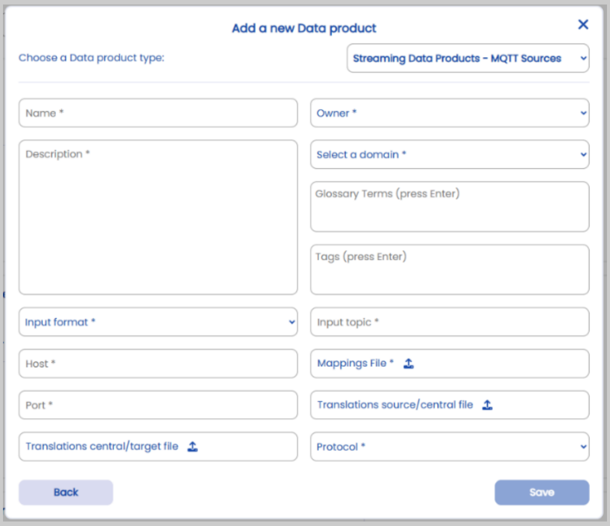
Context Broker

@ Domain: CloudFerro

Tags: aerOS AAA LDAP

Glossary terms:
http://www.w3.org/ns/org#Role
http://www.w3.org/ns/org#Organization
https://w3id.org/aerOS/continuum#User
http://xmlns.com/foaf/0.1/Person

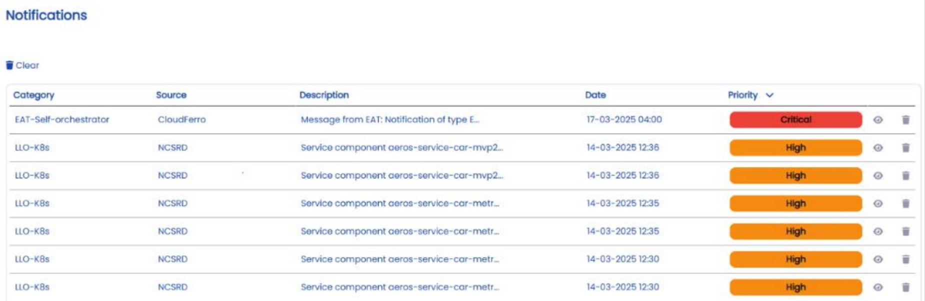
Describe and publish new Data Products.



Add a new Data product

Choose a Data product type: Streaming Data Products - MQTT Sources

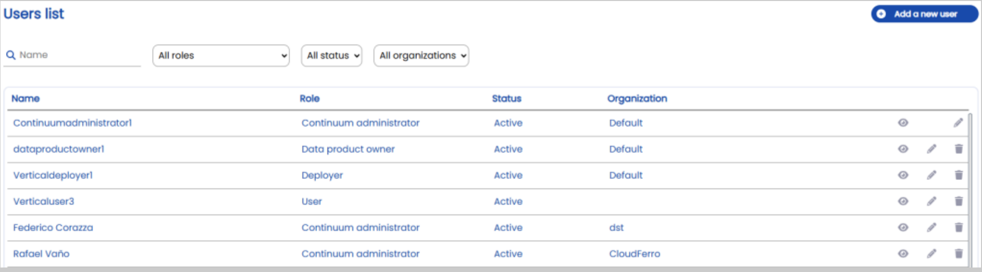
- A complete user notifications system, which is fully coupled with the aerOS orchestration module.



Notifications

Category	Source	Description	Date	Priority
EAT-Self-orchestrator	CloudFerro	Message from EAT: Notification of type E...	17-03-2025 04:00	Critical
LLO-K8s	NCSRD	Service component aeros-service-car-mvp2...	14-03-2025 12:36	High
LLO-K8s	NCSRD	Service component aeros-service-car-mvp2...	14-03-2025 12:36	High
LLO-K8s	NCSRD	Service component aeros-service-car-metr...	14-03-2025 12:35	High
LLO-K8s	NCSRD	Service component aeros-service-car-metr...	14-03-2025 12:35	High
LLO-K8s	NCSRD	Service component aeros-service-car-metr...	14-03-2025 12:30	High
LLO-K8s	NCSRD	Service component aeros-service-car-metr...	14-03-2025 12:30	High

- Managing the users and roles of the Management Portal.



Users list

Name	Role	Status	Organization
Continuumadministrator1	Continuum administrator	Active	Default
dataproductowner1	Data product owner	Active	Default
Verticaldeployer1	Deployer	Active	Default
Verticaluser3	User	Active	
Federico Corazza	Continuum administrator	Active	dst
Rafael Vaño	Continuum administrator	Active	CloudFerro

Therefore, **13 relevant functionalities are finally available through the platform**.

KPI 1.8.3 Performance of aerOS Federation Context Broker

Table 73: KPI 1.8.3 Performance of aerOS Federation Context Broker

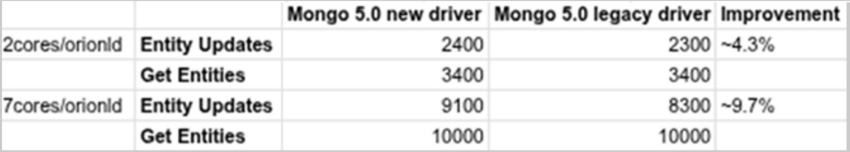
KPI ID number and partner resp.	**KPI 1.8.3**
KPI Name	**Performance of aerOS Federation Context Broker**
Description	This KPI represents the capacity of the aerOS Federator to withstand high querying and update loads. This will be a direct result of the capacity by the core element of

<table>
<tr><td></td><td colspan="3">such Federator, the ORION-LD context broker (release for aerOS). The metrics that will be assessed in this KPI will be: (a) number of simultaneous queries to get entities from the Broker–in the same second, (b) number of simultaneous updates of entities – in the same second.</td></tr>
<tr><td>Motivation</td><td colspan="3">The performance of Orion-LD is crucial for earOS because it will allow to understand how many data can be processed, as well as the capacity of the federated network of brokers to support automated distribution of the state of the continuum.
Orion-LD, by default, and for historical reasons, uses an old, deprecated driver for MongoDB (note that the associated database –MongoDB- needs more resources, but the restrictive element is the Context Broker).
However, the entire database layer has been reimplemented, using the newest MongoDB driver (if requested via a CLI option).
The old implementation is C++ while the new is pure C and thus, a boost in performance is expected.
The differences in "queries per second" and "entity updates per second" between the default (old, deprecated MongoDB driver) and the new implementation (new MongoDB driver) are the basis for this KPI.</td></tr>
<tr><td>Target value</td><td colspan="3">5000 queries/s, 2500 updates/s</td></tr>
<tr><td>Prerequisites</td><td colspan="3">Orion-LD, as the Context Broker which is part of aerOS core services, must be functional in (at least) a cloud domain.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">aerOS portal and the federation thanks to the Context Broker, allowing the DSNB (T4.6).</td></tr>
<tr><td>Evaluation means</td><td colspan="3">There is planned to realise a performance-measurement exercise in a development/integration environment This measurement exercise will be done at the end of the project (M38).
The procedure of measuring performance for this KPI is described here.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>3500 requests/s in the small scenario (70%)</i></td><td><i>3400 updates/s (136%), improvement of ~9,7% in entity updates</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">There is a strong dependency between Orion-ld and its supporting database mongoDB. In general, the performance of Orion-ld depends on its backend database performance. Thus, the most challenging thing done in ORION-LD, which can impact its performance is the implementation of a new driver for Mongo-db that could help improve the performance in a scenario with Context Source Registration federation. The driver has been built to work noy only with specific MongoDB versions (4.x, 5.0) but also with the newer versions, so that consistency is kept with internal needed functionalities (and staying lightweight, following aerOS' mindset).
A relevant note must be made related to this KPI:</td></tr>
</table>

- The tests performed (for the small scenario) back in D5.5 cannot be reproduced for the last testing iteration. The reason is that, when those analyses were made, the infrastructure was not operating over IoT-edge-cloud nature (as aerOS domains), but using AWS, cloud-only setup. For this iteration, the "small scenario" refers to a deployment made with Google Cloud with 2 CPU / 12 GB RAM Orion-LD, 8 CPU / 32 GB RAM Mongo-DB. In these new set of tests, similar and somehow equivalent configurations of previous tests have been used. However, the difference is that a local and edge environment has been utilized using docker (in terms of CPU and memory configurations) and will compare the improvements comparing the old mongodb driver with the new one. In this case, the equipment used is an Intel i7, 64Gb RAM and SSD storage.

Therefore, a direct comparison would not be fair, nor statistically valid. The data below are empirical but should be understood considering the previous.

Tests have been made increasing using 2 cores at the beginning for Orion-ld and also incrementing the Orion-ld cores to 7 and the mongodb cores to 10. The comparative results are seen below:



		Mongo 5.0 new driver	Mongo 5.0 legacy driver	Improvement
2cores/orionld	**Entity Updates**	2400	2300	~4.3%
	Get Entities	3400	3400	
7cores/orionld	**Entity Updates**	9100	8300	~9.7%
	Get Entities	10000	10000	

It can be seen that - considering the new testing environment-, there is clear improvement using the new driver vs the old (legacy) one, especially in updates per second, the really crucial operation for aerOS federation.

Therefore, attending the tests performed:

- Consistency with lightweight Ness and other internal operations is maintained (for the sake of aerOS' Orion-LD context broker as the heart of Data Fabric and the federation in the continuum). This, Mongo 5.0 is employed.
- Mongo 5.0 old driver was not valid, since it did not meet the requested (target) queries/s and updates/s, and was coded in a non-optimal way (C++, before, in contrast to pure C, now).
- The improvement with the new driver tailored for aerOS is ~4.3% using a 2-cores CB, and ~8,7% using a 7-core CB.

To sum up, improvement in the response capacity has been achieved thanks to aerOS innovations, which have meant the upgrading of MongoDB driver as well as making extensive use of Context Source Registration (feature of NGSI-LD) towards achieving federation in the continuum.

KPI 1.8.4 Federation asymptote with minimum latency (domains)

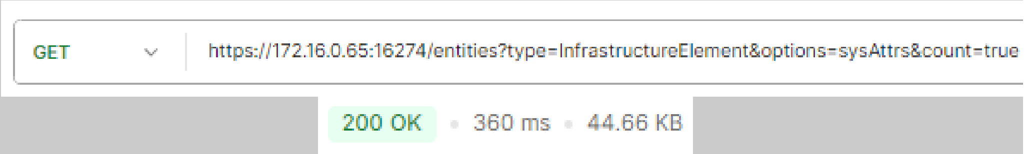
Table 74: KPI 1.8.4 Federation asymptote with minimum latency (domains)

KPI ID number and partner resp.	**KPI 1.8.4**
KPI Name	**Federation asymptote with minimum latency (domains)**

Description	The goal is to understand how many new domains can be added to a continuum (installation and addition procedure explained in deliverable D5.2) without trespassing a latency threshold. The latency will be associated to the retrieval of aerOS continuum monitoring information (e.g. the Infrastructure Elements or orchestrated Service Components) from all the available domains in the continuum. Here, the relevant aspect is to understand that aerOS federation is based on NGSI-LD's Context Source Registrations, that will automatically connect the ORION-LD of each domain with all the others (full connection network topology) for keeping updated the distributed state repository of the continuum. That connection is materialised in the exchange of distributed messages. The moment in which all the "update messages" will be completed since a new domain is added will be considered as the latency of the process.
Motivation	This KPI will represent the capacity of aerOS structure of federated domains in a single continuum to scale while maintaining a maximum decided latency.
Target value	≥ 4 domains
Prerequisites	The prerequisites will depend on the kind of tests performed to measure the value of the KPI as the evaluation will be composed of different scenarios. Thus, the prerequisites can be: (i) the installation in a local machine of the functional tests tool suite provided by the developers of Orion-LD; or (ii) a set of aerOS domains with, at least, Orion-LD and aerOS AAA components installed.
aerOS components (task)	Orion-LD (T4.2 and T4.6), aerOS AAA (T3.4) and Self-* modules (T3.5).
Evaluation means	Even though this measurement might be influenced by many factors not strictly related to aerOS traits (network delay, network load, processing time of messages in network nodes, etc.), data will be presented to gauge a relevant figure for this KPI. The measurement methodology planned in M38 has been: • A maximum "latency of the process" defined (when more federations are taking place in aerOS). • In real/relevant continuums (a selected set of pilots of the project), tests are conducted. Therefore, unlike the evaluation performed at M24, is not possible to dynamically add or remove domains solely for the purpose of this KPI evaluation. Instead, the same test will be repeated accross different pilots of the project until the maximum number of domains is reached. The process will start with a pilot whose continuum is composed of 2 domains, followed by one with 3 domains, and finally with a pilot composed of 4 domains. This will be the maximum capacity which can be measured in a real scenario. In M24, the defined tests were repeated in some testing environments (UPV and aerOS MVP continuums) because the pilots were still performing the installation of aerOS in their infrastructure. However, in M38 the process has been repeated in some pilots of the project because they are fully integrated with the aerOS Meta-OS. Moreover, this was a requirement stated in D5.5. Consequently, this represents the maximum level achieved at TRL5. Four different scenarios have been envisaged to elaborate the outcome of this KPI because as explained before, the measurement of this KPI might be influenced by many external factors to aerOS (e.g. network latency due to the physical distance of domains and message exchange among heterogeneous networks).

	Creation of several Orion-LD instances in the same testing machine, leveraging the functional testing tool suite provided by Orion-LD: this scenario aims to provide the theoretical maximum value for the federation asymptote since the vast majority of external factors are avoided, but it presents a strong dependence with the testing machine hardware. **Pilot 5**, which represents a continuum composed of two different domains (Application and Main). These domains are connected to an internal network of the OTE research labs. **Pilot 4**: represents a continuum with 3 geographically distributed domains, which include a public cloud domain (deployed in AWS), a private cloud domain hosted at CUT university and an edge domain deployed at the Port of Limassol. This edge domain can be considered a restrictive environment. **Pilot 2**: a continuum composed of 4 multi cloud-edge distributed domains. As the continuum of this pilot includes the maximum number of domains in the project (4), it serves as the reference case to determine whether this KPI reaches its target validation value (at least 4 federated domains		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0***	***Scenario 1: Unlimited, tested in 10 domains.*** ***Scenario 2: >3 domains*** ***Scenario 3: 3 domains***	***Scenario 1: Unlimited, tested in 15 domains.*** ***Scenario 2: 2 domains*** ***Scenario 3: 3 domains*** ***Scenario 4: 4 domains*** ***Total: 24 (>600% achieved***
Outcome elaboration (M38)	For the first scenario, these are the results: in a medium-powerful VM (4 vCPU and 16GB RAM), the functional tests are run quickly with a response time of less than 40 ms for the GET requests (the entire test is run in less than 1 second) in a scenario consisting of 15 Orion-LD instances. This means that (i) the results of this scenario are highly dependent on the hardware on which it is executed, (ii) it's an ideal scenario without external factors such as networking issues, and (iii) it clearly presents a strong relationship with KPI 1.8.3, which aims to evaluate the performance of the context broker. In M38, as stated in the "evaluation means" section, instead of increasing the number of domains in a testing environment, it has been decided to repeat the same test across different pilots of the project until the maximum number of domains is reached. The "latency of the process" has been decided to be set to 5 seconds, based on several factors: i) the results of tests performed in different scenarios for this KPI in D5.5 (including the demo performed in the aerOS mid-term review), ii) the default timeout value for distributed requests in Orion-LD, and iii) the consensus that this maximum value ensures an acceptable user experience after deploying and testing the aerOS Meta-OS in the pilots of the project. In scenario 2 (Pilot 5), whose continuum is composed of 2 domains, the current response time for distributed requests is between 300-400ms. The measured response time is appropiate, but this result can be explained due to the use of an internal network		

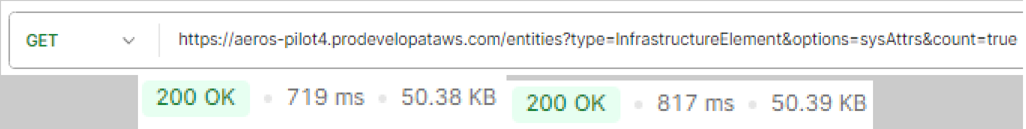
and the reduced number of domains, which enables a bidirectional communications channel.



GET https://172.16.0.65:16274/entities?type=InfrastructureElement&options=sysAttrs&count=true

200 OK • 360 ms • 44.66 KB

Scenario 3 represents a significant breakthrough due to the fact that Pilot 4 is composed of 3 geographically distributed domains and one of them is connected to a restrictive network environment. Nevertheless, the measured results are pretty good (around 700-800 ms) and demonstrate that the behaviour of the aerOS federation aligns with expectations, remaining well below the defined maximum threshold of 5 seconds

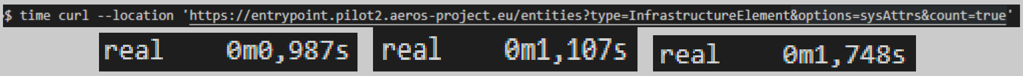


GET https://aeros-pilot4.prodevelopataws.com/entities?type=InfrastructureElement&options=sysAttrs&count=true

200 OK • 719 ms • 50.38 KB

200 OK • 817 ms • 50.39 KB

The last scenario represents a cloud multi-edge continuum, which is composed of a total of 4 distributed domains, the greatest continuum of the pilots of the project. The response times of the distributed requests increase to around 1 second, but some peaks of less than 2 seconds appear. This result is satisfactory because i) it's still under the defined maximum latency of the process (5 seconds) and ii) these measurements are strongly influenced by external factors such as network conditions.



```
$ time curl --location 'https://entrypoint.pilot2.aeros-project.eu/entities?type=InfrastructureElement&options=sysAttrs&count=true'
real	0m0,987s
real	0m1,107s
real	0m1,748s
```

Finally, the data obtained so far confirms that with four real domains, the federation provided by the aerOS Meta-OS is fully functional, even under less reliable network connections across different countries and within restrictive scenarios. This proves that the KPI target has been successfully achieved

KPI 1.8.5 Average offloading ratio of entrypoint balancing in aerOS scenarios

Table 75: KPI 1.8.5 Average offloading ratio of entrypoint balancing in aerOS scenarios

KPI ID number and partner resp.	**KPI 1.8.5**
KPI Name	**Average offloading ratio of entrypoint balancing in aerOS scenarios**
Description	This KPI represents the effectiveness of the balancing algorithms selected and deployed in the Management Portal for achieving entry point diversity usage. The goal is to demonstrate that, on average, 30% of requests sent to EB are distributed for first processing to HLOs located in domains other than the one containing EB (entry point domain).

Motivation	Measure the effectiveness of EB in the distribution of requests between HLOs, which aims to minimize single HLO overutilization
Target value	30%
Prerequisites	Management Portal must be deployed on the aerOS and must correctly properly pass requests to EB; The information about HLOs and their domains must be present in Orion-LD Context Broker; The FE of HLOs should be accessible under addresses registered in Orion-LD Context Broker.
aerOS components (task)	API Gateway (T3.4), Orion-LD (T4.2), aerOS Management Portal (T4.6)
Evaluation means	The EB algorithm will be tested both in development/integration scenarios and in pilots. The tests will be performed separately on the continuum with 2 domains (preliminary tests) and 3 domains (advanced tests). For each test, 15 workloads will be used. In total, it is envisioned that 4 types of tests are going to be performed, aiming to encompass and represent various scenarios, in which the EB can be used: Test Case 1: There are no running services present in the continuum. The client requests are sent directly to the EB, which uses the weighting function based on CPU usage. The scenario aims to evaluate, whether the EB will be able to distribute the requests to different domains even when the score (i.e., number of running services divided by the weight) of each domain remains the same. Test Case 2: There are running services deployed on each of the domains: • For the continuum with 2 domains: 2 services running on *Domain1*, 1 service running on the *Domain2*. • For the continuum with 3 domains: 2 services running on *Domain1*, 1 service running for *Domain2*, 1 service running for *Domain3*. The client requests are directly sent to the EB, which uses the weighting function based on CPU usage. The scenario aims to evaluate, whether the EB will be able to distribute the requests when the continuum domains are in different states. Test Case 3: The state of the continuum is the same as in Test Case 2. However, this time, a different weighting function is going to be used. In particular, the RAM usage is to be considered along with CPU usage. The scenario aims to compare the effectiveness of different weighting functions, so that the one for which the better outcomes are achieved (i.e., offloading ratio) can be selected. Test Case 4: The state of the continuum is the same as in Test Case 2. However, in this scenario, part of the requests will have a target domain indicated by the client. For these requests, the execution of the load balancing algorithm should be omitted by EB. The scenario aims to evaluate the decision-making process of EB, as well as its ability of capturing dynamic changes of the continuum. For the states of the continuum described in Test Case 1 and Test Case 2, the tests are also going to be performed without the usage of EB, so that both of the obtained outcomes can be compared. To measure the offloading ratio, a strategy will be envisaged to quantify/catalogue whether a service deployment request originated in a certain domain and queried the HLO of a different domain as first processing option.

Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	**0% (when all clients select the target HLO explicitly)**	**Test Case 1 (preliminary): 33.3% offloading ratio (25%)**	**Test Case 1: 50% (accomplished)** **Test Case 2: 44% (accomplished)** **Test Case 3: 50% (accomplished)** **Test Case 4: 50% (accomplished)**

Outcome elaboration (M38)

Test Case 1

As part of the experiments, **5 services** were deployed in aerOS continuum. The offloading ratio was continuously monitored by the EB component and reflected in the component's logs. As can be observed in the following Figure, the offloading ratio was kept at the **50%:**

```
 is marked as available for selection.
2025-10-07T15:08:11.279Z  INFO 1 --- [nio-8080-exec-4] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.5
2025-10-07T15:08:42.639Z  INFO 1 --- [nio-8080-exec-2] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.5
2025-10-07T15:08:52.867Z  INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Domain https://cf-mvp-domain.aeros-project.eu
 has been made temporarily disabled from selection.
2025-10-07T15:08:52.868Z  INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.5
2025-10-07T15:10:48.988Z  INFO 1 --- [nio-8080-exec-7] o.aeros.service.LoadBalancerServiceImpl  : Domain https://cf-mvp-domain.aeros-project.eu
 is marked as available for selection.
2025-10-07T15:10:48.988Z  INFO 1 --- [nio-8080-exec-7] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.5
```

Furthermore, by analysing the Management Portal logs, it was confirmed that the requests were successfully distributed among both domains:

```
2025-10-07T15:06:25.610Z DEBUG 1 --- [io-8080-exec-14] e.a.m.services.DeploymentServiceImpl     : Request was distributed successfully to HLO
 located in the domain https://cf-mvp-domain.aeros-project.eu.
2025-10-07T15:06:25.738Z DEBUG 1 --- [nio-8080-exec-7] o.s.security.web.FilterChainProxy        : Securing GET ↲
↳/portal-backend/deployments/services
```

```
2025-10-07T15:08:11.997Z DEBUG 1 --- [io-8080-exec-19] e.a.m.services.DeploymentServiceImpl     : Request was distributed successfully to HLO
 located in the domain https://ncsrd-mvp-domain.aeros-project.eu.
2025-10-07T15:08:12.148Z DEBUG 1 --- [io-8080-exec-14] o.s.security.web.FilterChainProxy        : Securing GET
 /portal-backend/deployments/services
```

These experimental results confirm the correctness of EB operations and the achievement of the estimated target value.

Test Case 2

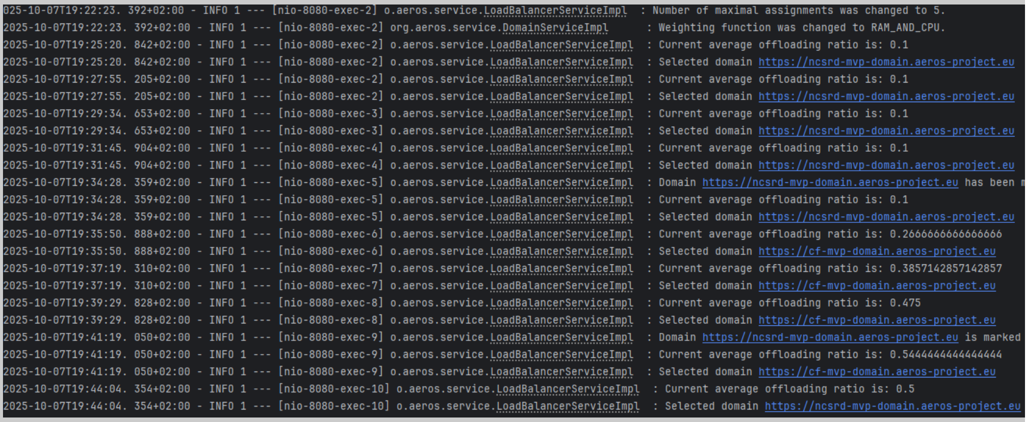
For the purpose of the second Test Case, selected services, deployed as part of Test Case 1, were started as described in "Evaluation Means" section. Afterwards, **5 new services** were deployed in order to measure the changes in the offloading ratio. The logging mechanisms were extended to better illustrate the obtained behaviour of EB. The results of the experiments performed on the continuum with 2 domains are illustrated in the following figure:

```
-2025-10-07T18:31:38. 578+02:00 - INFO 1 --- [nio-8080-exec-2] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
-2025-10-07T18:32:43. 869+02:00 - INFO 1 --- [nio-8080-exec-3] o.aeros.service.LoadBalancerServiceImpl  : Domain https://ncsrd-mvp-domain.aeros-project.eu is marked as availa
-2025-10-07T18:32:43. 869+02:00 - INFO 1 --- [nio-8080-exec-3] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.43333333333333335
-2025-10-07T18:32:43. 869+02:00 - INFO 1 --- [nio-8080-exec-3] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://cf-mvp-domain.aeros-project.eu
-2025-10-07T18:34:04. 257+02:00 - INFO 1 --- [nio-8080-exec-4] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.35
-2025-10-07T18:34:04. 257+02:00 - INFO 1 --- [nio-8080-exec-4] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
-2025-10-07T18:35:34. 785+02:00 - INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Domain https://ncsrd-mvp-domain.aeros-project.eu has been made tempo
-2025-10-07T18:35:34. 785+02:00 - INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.29999999999999993
-2025-10-07T18:35:34. 785+02:00 - INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
-2025-10-07T18:36:56. 225+02:00 - INFO 1 --- [nio-8080-exec-6] o.aeros.service.LoadBalancerServiceImpl  : Domain https://ncsrd-mvp-domain.aeros-project.eu is marked as availa
-2025-10-07T18:36:56. 225+02:00 - INFO 1 --- [nio-8080-exec-6] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.43333333333333335
-2025-10-07T18:36:56. 225+02:00 - INFO 1 --- [nio-8080-exec-6] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://cf-mvp-domain.aeros-project.eu
```

As can be observed, the offloading ratio is slightly lower than in Test Case 1. It can be attributed to the initially unbalanced state of domains, which is being balanced over time as indicated by the increasing value of the offloading ratio. The final result of the experiment indicated offloading ratio being around **44%**, which still overcomes the target value.

Test Case 3:

In the next test scenario, the configuration of EB was dynamically adjusted using the */configure* endpoint. In particular, two parameters were modified: (1) maximal assignments number (i.e. maximal number of consecutive redirections of a service to a specific domain) and (2) weighting function (i.e., function used to compute the score of individual domains, so that the one with the best score can be selected for a service). The current implementation of EB supports two default types of weighting functions: *CPU* (calculates the domain's score solely based on the current CPU usage) and *RAM_AND_CPU* (calculates the domain's score taking into account the combination of CPU and RAM usage). Both Test Case 1 and Test Case 2, used the CPU weighting function. Moreover, in those experiments, a relatively low (equal to 2) value of the maximal assignments parameter was set. Therefore, to evaluate the EB upon different system configurations, for Test Case 3, the weighting function was set to ***RAM_AND_CPU***, while the maximal assignments parameter was set to **5**. The experiment was run by consecutively deploying **10 services**, to observe the trend in offloading ratio for a larger number of requests. The results are illustrated in the following figure:

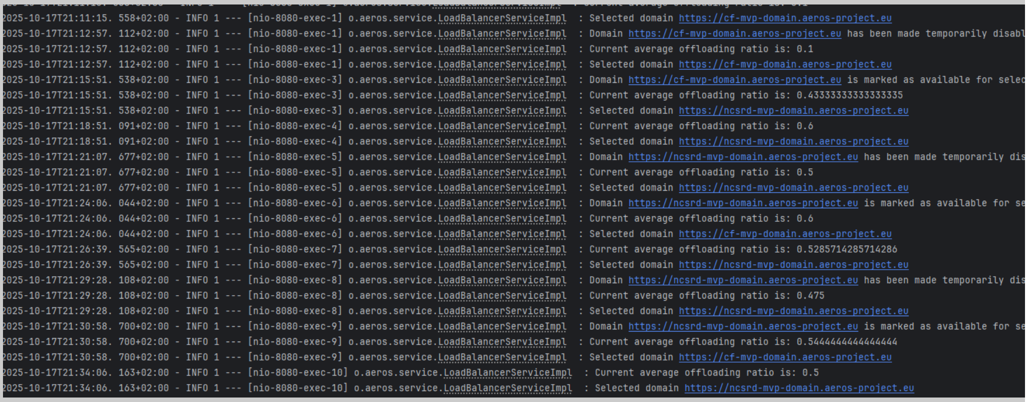


```
025-10-07T19:22:23. 392+02:00 - INFO 1 --- [nio-8080-exec-2] o.aeros.service.LoadBalancerServiceImpl  : Number of maximal assignments was changed to 5.
2025-10-07T19:22:23. 392+02:00 - INFO 1 --- [nio-8080-exec-2] org.aeros.service.DomainServiceImpl      : Weighting function was changed to RAM_AND_CPU.
2025-10-07T19:25:20. 842+02:00 - INFO 1 --- [nio-8080-exec-2] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.1
2025-10-07T19:25:20. 842+02:00 - INFO 1 --- [nio-8080-exec-2] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
2025-10-07T19:27:55. 205+02:00 - INFO 1 --- [nio-8080-exec-2] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.1
2025-10-07T19:27:55. 205+02:00 - INFO 1 --- [nio-8080-exec-2] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
2025-10-07T19:29:34. 653+02:00 - INFO 1 --- [nio-8080-exec-3] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.1
2025-10-07T19:29:34. 653+02:00 - INFO 1 --- [nio-8080-exec-3] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
2025-10-07T19:31:45. 904+02:00 - INFO 1 --- [nio-8080-exec-4] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.1
2025-10-07T19:31:45. 904+02:00 - INFO 1 --- [nio-8080-exec-4] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
2025-10-07T19:34:28. 359+02:00 - INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Domain https://ncsrd-mvp-domain.aeros-project.eu has been 
2025-10-07T19:34:28. 359+02:00 - INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.1
2025-10-07T19:34:28. 359+02:00 - INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
2025-10-07T19:35:50. 888+02:00 - INFO 1 --- [nio-8080-exec-6] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.2666666666666666
2025-10-07T19:35:50. 888+02:00 - INFO 1 --- [nio-8080-exec-6] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://cf-mvp-domain.aeros-project.eu
2025-10-07T19:37:19. 310+02:00 - INFO 1 --- [nio-8080-exec-7] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.3857142857142857
2025-10-07T19:37:19. 310+02:00 - INFO 1 --- [nio-8080-exec-7] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://cf-mvp-domain.aeros-project.eu
2025-10-07T19:39:29. 828+02:00 - INFO 1 --- [nio-8080-exec-8] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.475
2025-10-07T19:39:29. 828+02:00 - INFO 1 --- [nio-8080-exec-8] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://cf-mvp-domain.aeros-project.eu
2025-10-07T19:41:19. 050+02:00 - INFO 1 --- [nio-8080-exec-9] o.aeros.service.LoadBalancerServiceImpl  : Domain https://ncsrd-mvp-domain.aeros-project.eu is marked
2025-10-07T19:41:19. 050+02:00 - INFO 1 --- [nio-8080-exec-9] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.5444444444444444
2025-10-07T19:41:19. 050+02:00 - INFO 1 --- [nio-8080-exec-9] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://cf-mvp-domain.aeros-project.eu
2025-10-07T19:44:04. 354+02:00 - INFO 1 --- [nio-8080-exec-10] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.5
2025-10-07T19:44:04. 354+02:00 - INFO 1 --- [nio-8080-exec-10] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
```

It is visible that the EB acknowledged the re-configuration of weighting function and the maximal assignments parameter. Moreover, it can be noticed that, contrary to the previous examples, the domains were switched on and off less frequently. It is also evident that over time the offloading ratio was increasing, reaching the value of **50%.**

Test Case 4:

For this test case, the same configuration as in Test Case 1 and Test Case 2 (i.e. maximal assignments = 2, weighting function = CPU) was provided. The main purpose of this test was to evaluate whether the EB will be able to correctly handle the semi-automatic deployment of services (i.e. where the end user indicates the pool of IEs for service deployment). In total **10 services** were deployed. The first **3** were deployed by using semi-automatic deployment and indicating among the target IEs, all IEs belonging to CF domain. The remaining **7 services** were deployed with standard automatic deployment. The results are presented in the following figure:



```
2025-10-17T21:11:15. 558+02:00 - INFO 1 --- [nio-8080-exec-1] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://cf-mvp-domain.aeros-project.eu
2025-10-17T21:12:57. 112+02:00 - INFO 1 --- [nio-8080-exec-1] o.aeros.service.LoadBalancerServiceImpl  : Domain https://cf-mvp-domain.aeros-project.eu has been made temporarily disabl
2025-10-17T21:12:57. 112+02:00 - INFO 1 --- [nio-8080-exec-1] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.1
2025-10-17T21:12:57. 112+02:00 - INFO 1 --- [nio-8080-exec-1] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://cf-mvp-domain.aeros-project.eu
2025-10-17T21:15:51. 538+02:00 - INFO 1 --- [nio-8080-exec-3] o.aeros.service.LoadBalancerServiceImpl  : Domain https://cf-mvp-domain.aeros-project.eu is marked as available for selec
2025-10-17T21:15:51. 538+02:00 - INFO 1 --- [nio-8080-exec-3] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.43333333333333335
2025-10-17T21:15:51. 538+02:00 - INFO 1 --- [nio-8080-exec-3] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
2025-10-17T21:18:51. 091+02:00 - INFO 1 --- [nio-8080-exec-4] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.6
2025-10-17T21:18:51. 091+02:00 - INFO 1 --- [nio-8080-exec-4] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
2025-10-17T21:21:07. 677+02:00 - INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Domain https://ncsrd-mvp-domain.aeros-project.eu has been made temporarily dis
2025-10-17T21:21:07. 677+02:00 - INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.5
2025-10-17T21:21:07. 677+02:00 - INFO 1 --- [nio-8080-exec-5] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
2025-10-17T21:24:06. 044+02:00 - INFO 1 --- [nio-8080-exec-6] o.aeros.service.LoadBalancerServiceImpl  : Domain https://ncsrd-mvp-domain.aeros-project.eu is marked as available for se
2025-10-17T21:24:06. 044+02:00 - INFO 1 --- [nio-8080-exec-6] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.6
2025-10-17T21:24:06. 044+02:00 - INFO 1 --- [nio-8080-exec-6] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://cf-mvp-domain.aeros-project.eu
2025-10-17T21:26:39. 565+02:00 - INFO 1 --- [nio-8080-exec-7] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.5285714285714286
2025-10-17T21:26:39. 565+02:00 - INFO 1 --- [nio-8080-exec-7] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
2025-10-17T21:29:28. 108+02:00 - INFO 1 --- [nio-8080-exec-8] o.aeros.service.LoadBalancerServiceImpl  : Domain https://ncsrd-mvp-domain.aeros-project.eu has been made temporarily dis
2025-10-17T21:29:28. 108+02:00 - INFO 1 --- [nio-8080-exec-8] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.475
2025-10-17T21:29:28. 108+02:00 - INFO 1 --- [nio-8080-exec-8] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
2025-10-17T21:30:58. 700+02:00 - INFO 1 --- [nio-8080-exec-9] o.aeros.service.LoadBalancerServiceImpl  : Domain https://ncsrd-mvp-domain.aeros-project.eu is marked as available for se
2025-10-17T21:30:58. 700+02:00 - INFO 1 --- [nio-8080-exec-9] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.5444444444444444
2025-10-17T21:30:58. 700+02:00 - INFO 1 --- [nio-8080-exec-9] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://cf-mvp-domain.aeros-project.eu
2025-10-17T21:34:06. 163+02:00 - INFO 1 --- [nio-8080-exec-10] o.aeros.service.LoadBalancerServiceImpl  : Current average offloading ratio is: 0.5
2025-10-17T21:34:06. 163+02:00 - INFO 1 --- [nio-8080-exec-10] o.aeros.service.LoadBalancerServiceImpl  : Selected domain https://ncsrd-mvp-domain.aeros-project.eu
```

It can be observed that at first CF domain was primarily selected by EB, but after deploying the first 3 services (i.e., those using semi-automatic deployment) and starting them in CF domain, the NCSRD started to be selected more frequently. At the end of the conducted experiments, EB was selecting the domains for services by turns, which was reflected through the offloading ratio that was reaching around **50%**.

KPI 1.8.6 QoE of Management Portal deployed on pilots

Table 76: KPI 1.8.6 QoE of Management Portal deployed on pilots

KPI ID number and partner resp.	**KPI 1.8.6**
KPI Name	**QoE of Management Portal deployed on pilots**
Description	The idea of this KPI is to evaluate the QoE of the UI of the Management Portal. In particular, it is the intention for quantify a metric for tracking both behavioural and attitudinal perception of the webapp.
Motivation	The rationale behind this KPI is to be able to assess the quality of experience of stakeholders using the aerOS Management Portal. Note that stakeholders (users) of the implemented UI will be both system administrators (for e.g., configuring domains) and 4owners (for e.g., monitoring data or KPIs).
Target value	>=68 SUS score
Prerequisites	aerOS runtime working, web service ready, HLO ready and Management Portal deployed in pilots
aerOS components (task)	HLO (T3.3), Management Portal (T4.6)
Evaluation means	For this first piloting phase, it was decided, as also anticipated in D5.2, to create a QoE survey based on the **System Usability Scale (SUS),** a widely recognized and standardised tool for assessing the usability of a system. The SUS provides a reliable measure of usability with a small number of questions, making it efficient for both respondents and analysts. It consists of 10 statements with alternating positive and negative phrasing to reduce response bias. Each item is scored on a 5-point Likert scale, ranging from Strongly Disagree (1) to Strongly Agree (5). This final score represents the usability of the system. By gathering feedback on key usability aspects, the survey aims

to identify strengths and areas for improvement in the portal's design and functionality. In detail, At M29 it is planned the first QoE survey that effectively evaluates user experience in digital services and portals, particularly within the context of role-specific functionalities and interactions. This QoE survey will be designed based on several established QoE standards and frameworks to ensure comprehensive and relevant questions that cover key aspects of user experience:

- ITU-T P.800 Series, methods for subjective determination of transmission quality, adaptable for interactive digital experiences.
- ISO 9241-11:2018, defines usability based on effectiveness, efficiency, and satisfaction. Questions on usability, ease of navigation, and intuitiveness are based on this standard.
- ISO/IEC 25010:2011, provides a quality model including characteristics such as functional suitability, performance efficiency, usability, reliability, security, compatibility, maintainability, and portability. The survey addresses aspects like performance, functionality, and overall satisfaction.
- Nielsen's Usability Heuristics principles for user interface design, including visibility of system status, consistency, error prevention, and ease of use.

Finally, an updated and final QoE survey will be conducted at M35.

Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***N/A***	***70.8 average value***

Outcome elaboration (M38)

Context of the Survey

To evaluate the usability and effectiveness of the management portal, we administered the System Usability Scale (SUS) survey to pilot partners of aerOS. These partners had fully installed the platform and actively used all core features, including Home, Domains, Deployments, Data Products, Benchmarking, and Users. Their feedback reflects hands-on experience with the complete functionality of the portal and provides a reliable benchmark for assessing the portal’s user experience.

SUS Score by Participant

Participant	SUS	Indicative Grade*	Adjective Range*
A	52.5	D	Poor / Marginal
B	87.5	A	Excellent
C	72.5	C	OK / Above Average
Mean	70.8	C	OK / Above Average

*Approximate mappings from commonly used SUS interpretation guidelines.

Average Scores by Survey Item

Negative statements have been reverse-scored to ensure consistency when interpreting the data. In a System Usability Scale (SUS) questionnaire, half of the items are worded negatively, which would otherwise skew the score if interpreted at face value. To normalize responses, the scoring is inverted for these negative items so that a “Strongly Disagree” response is converted to a high score, matching the positive direction of the other items. For quick scanning and to provide a more intuitive visual reference, we have also converted the standard 1–5 Likert values into a 0–100% scale, where 1 equals 0% (least positive response) and 5 equals 100% (most positive response).

Item	Statement (abridged)	A	B	C	Mean (1–5)	Mean %
Q1	Easy to navigate sections	SA	SA	SA	5.00	100.0
Q2	Layout confusing (neg)	D	SD	D	4.33	83.3
Q3	Confident doing role tasks	N	SA	A	4.00	75.0
Q4	Needed tech support (neg)	N	D	D	3.67	66.7
Q5	Network graph clear	SA	SA	A	4.67	91.7
Q6	Sections inconsistent (neg)	A	D	D	3.33	58.3
Q7	Learned Benchmarking quickly	N	A	N	3.33	58.3
Q8	Data Products complex (neg)	SA	D	N	2.67	41.7
Q9	Portal had needed info/tools	N	A	A	3.67	66.7
Q10	Took lots of time to learn (neg)	A	SD	D	3.67	66.7

Key Findings

The analysis of the three SUS responses shows that overall usability of the portal is rated as acceptable, with a mean SUS score of approximately 70.8. Navigation is the strongest KPI, achieving a perfect average rating (100%) across all participants, which confirms that users can easily move between sections such as Home, Domains, Deployments, Data Products, Benchmarking, and Users. The Continuum network graph is another strong performer with a mean rating of 91.7%, indicating that the visualization of computing resources is clear and effective. However, there are performance gaps in workflow efficiency and consistency. The processfor managing or submitting Data Products records a low average rating (41.7%), highlighting a potential usability bottleneck. This underscores the importance of future implementing an onboarding process to improve these metrics and, therefore, support users more effectively.

aerOS embedded analytics

KPI 1.9.1 # pre-packaged functions supported by Embedded Analytics Tool (EAT)

Table 77: KPI 1.9.1 # pre-packaged functions supported by Embedded Analytics Tool (EAT)

KPI ID number and partner resp.	**KPI 1.9.1**
KPI Name	**# pre-packaged functions supported by Embedded Analytics Tool (EAT)**
Description	The Embedded Analytics Tool is a platform for the design, development and deployment of analytical functions. Several generalised functions are packaged with the Embedded Analytics Tool to address common operations to provide insights for management and AI/ML components.

Motivation	The pre-packaged functions of the Embedded Analytics Tool provide basic operations for the aerOS system. These functions are leveraged by other components to provide insights such as data samples or highlight anomalies and data drifts. These functions are also generalised and can be customised through user parameters, allowing them to have "plug and play" characteristics in a range of different environments and scenarios.		
Target value	3		
Prerequisites	The Embedded Analytics Tool must be installed according to the instructions provided in the project repository. These instructions address security and privacy features through access tokens for downloading EAT components and credentials for dashboard login. EAT is considered successfully installed when all EAT components report "Running".		
aerOS components (task)	Embedded Analytics Tool (T4.4)		
Evaluation means	The development of each function follows an incremental development approach with a unique predefined test data set for each function. Functions are evaluated based on expected versus actual results of function execution.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*0*	*3 (100%)*	*3 (100%)*
Outcome elaboration (M38)	Explanation of the functions was provided in D4.2 and D4.3, and the individual links to them were provided in D5.5. Series of functions have been developed and demonstrated as part of the MVP experiments. Demonstrations are presented here on the project YouTube channel: https://youtu.be/UV4mnN4CrwI?si=RH1ERVu7QWV0-jBv		

KPI 1.9.2 # northbound wrappers designed for common operations with EAT

Table 78: KPI 1.9.2 # northbound wrappers designed for common operations with EAT

KPI ID number and partner resp.	**KPI 19.2**
KPI Name	**# northbound wrappers designed for common operations with EAT**
Description	Wrappers will be designed and implemented for the creation of Embedded Analytics Tool functions. These wrappers will be available to all function authors who utilise the aerOS templates. These templates will be stored in the Embedded Analytics Tool repository and be available to all project partners.
Motivation	Adopting a Function as a Service approach for the Embedded Analytics Tool enables flexibility for the function authors, but also introduces function design and communication challenges. Templating allows for function design and communication

	to be structured and guided. Function authors can focus on the core logic of their functions while using approved and tested implementations for common operations such as data retrieval or triggering external actions.		
Target value	3		
Prerequisites	Installation of the EAT functions repository is required. EAT specific applications such as faas-cli allow users to engage with EAT to create, deploy and remove functions. The aerOS template provides a structured model preconfigured with aerOS specific features such as visualization.		
aerOS components (task)	Embedded Analytics Tool (T4.4)		
Evaluation means	The aerOS template provides 3 defined operations within aerOS functions. These are Data Retrieval (e.g., requesting data from Data Fabric), aerOS component communication (e.g., forwarding data to HLO), and data visualization (e.g., exposing in-function metrics to EAT Grafana component). These operations are evaluated based on their successful execution as part of EAT pre-packaged functions, and the creation of use case specific functions.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0***	***3 (100%)***	***3 (100%)***
Outcome elaboration (M38)	Templates created for the Embedded Analytics Tool provide mechanisms for the user created functions to engage with other aerOS components such as the Data Fabric and HLO. Explanation of the templates was provided in D4.2 and D4.3, and the individual links to them were provided in D5.5. In an effort to streamline this experience templates provide interfaces using approved techniques through a generalised approach. Again, these templates are utilised in the MVP demonstrator here: https://youtu.be/UV4mnN4CrwI?si=RH1ERVu7QWV0-jBv Templates are also utilised in every pilot deploying the Embedded Analytics Tool as they are required for the creation and deployment of functions in aerOS		

Stakeholder user satisfaction

KPI 1.10.1 Successful conduction of Open Calls (KVI-7.1)

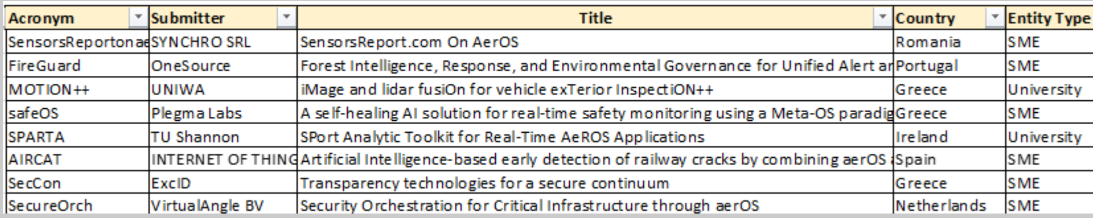
Table 79: KPI 1.10.1 Successful conduction of Open Calls (KVI-7.1)

KPI ID number and partner resp.	**KPI 1.10.1**
KPI Name	**Successful conduction of Open Calls (KVI-7.1)**

<table>
<tr><td>Description</td><td colspan="3">This KPI will represent the combined number of applications received between round 1 and round 2 of Open Calls funding opportunity. Additionally, this KPI will refer to the successful implementation of Open Call projects (60k€, 9 months each).</td></tr>
<tr><td>Motivation</td><td colspan="3">To illustrate the capacity of the Consortium to engage dynamic, vibrant researchers in Universities, SMEs and RTOs to utilise aerOS technologies and provided added value to the pilots or the research strategy of the project.</td></tr>
<tr><td>Target value</td><td colspan="3">> 80 applicants</td></tr>
<tr><td>Prerequisites</td><td colspan="3">• Pre-requisites for M24:
○ The Process for the first Open Call award has been completed: publication, application window, evaluation, ranking, decision, announcement of winners and commencement of projects.
○ The Process for the 2nd Open Call is initiated: publication and opening of application window.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">First Open Call projects are in execution, and so far, no specific components have been exploited yet.</td></tr>
<tr><td>Evaluation means</td><td colspan="3">On the one hand, the number of applications received will be published in aerOS website some days after the close of each of the two application windows. On the other hand, the final reports of the 15 OC projects to be funded will be summarised and included in deliverable D1.4.
The KPI target will be then represented as follows: >80, and 15 out of 15 successfully completed projects that provide lessons learnt</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>0</i></td><td><i>38 applicants (47.5%)</i></td><td><i>72 applicants (90%)</i>
<i>All 15 OC projects completed successfully</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">A total of 38 proposals were submitted for Open Call Application #1 of aerOS. The awarded proposals (that became successful projects) are:
<table>
<tr><th>Acronym</th><th>Submitter</th><th>Title</th><th>Pilot</th><th>Challenge</th><th>Country</th><th>Entity Type</th></tr>
<tr><td>HACER</td><td>Bytek</td><td>High Accuracy Cost Efficient Differential Positioning System using Real-Time Kinematics to optimise port logistics</td><td>P4</td><td>P4C1</td><td>Spain</td><td>SME</td></tr>
<tr><td>DAIMon</td><td>UPCT</td><td>Distributed AI-based Atmospheric Visibility Index Service for Agricultural Mobile Machinery within the aerOS framework</td><td>P3</td><td>P3C1</td><td>Spain</td><td>University</td></tr>
<tr><td>EcoQM</td><td>Nissatech</td><td>Framework for eco-quality monitoring and control supported by aerOS</td><td>P1</td><td>P1C1</td><td>Serbia</td><td>SME</td></tr>
</table></td></tr>
</table>

ENERGETIC	Nextworks	Energy management and comfort living for green, healthy and productive offices	P5	P5C2	Italy	SME
ANEOSP	Secmotic Innovation SL	AI Nodes for Enhanced Occupational Safety in Ports	P4	P4C3	Spain	SME
IBRTEFC	The Data Cooks	IoT-Based Real-Time Environmental Footprint Calculator	P1	P1C1	Netherlands	SME
GreenAnalyzer	UCY	A framework for Geo-distRibuted Edge-cloud Energy consumption ANALYsis towards Zero Emission Rates	P2	P2C2	Cyprus	University

A total of 34 proposals were submitted for Open Call Application #2 of aerOS. The approved proposals (that became successful projects) are:



Acronym	Submitter	Title	Country	Entity Type
SensorsReportonae	SYNCHRO SRL	SensorsReport.com On AerOS	Romania	SME
FireGuard	OneSource	Forest Intelligence, Response, and Environmental Governance for Unified Alert ar	Portugal	SME
MOTION++	UNIWA	iMage and lidar fusiOn for vehicle exTerior InspectiON++	Greece	University
safeOS	Plegma Labs	A self-healing AI solution for real-time safety monitoring using a Meta-OS paradig	Greece	SME
SPARTA	TU Shannon	SPort Analytic Toolkit for Real-Time AeROS Applications	Ireland	University
AIRCAT	INTERNET OF THING	Artificial Intelligence-based early detection of railway cracks by combining aerOS	Spain	SME
SecCon	ExcID	Transparency technologies for a secure continuum	Greece	SME
SecureOrch	VirtualAngle BV	Security Orchestration for Critical Infrastructure through aerOS	Netherlands	SME

Even though the total number of applicants has fallen short to the expected 80, it must be considered that the quality of the proposals has lived up to the expectations (even surpassed it). All projects ended successfully, all delivered a series of reflections for improvement of our Meta-OS, and several of them provided datasets for public dissemination.

KPI 1.10.2 # of stakeholders deploying aerOS

Table 80: KPI 1.10.2 # of stakeholders deploying aerOS

KPI ID number and partner resp.	**KPI 1.10.2**
KPI Name	**# of stakeholders deploying aerOS**
Description	The number of entities public/private deploying aerOS Meta-OS components to support operation and implementation of advanced hyper distributed applications.
Motivation	The number of stakeholders deploying aerOS will generate the necessary evidence to support future adoption of Meta-OS assets
Target value	5
Prerequisites	aerOS ready to be deployed, with all the needs that implies.
aerOS components (task)	At least, the core aerOS services needed to set up an aerOS domain: HLO (T3.3), LLO (T3.3), Self-* (T3.5), Context Broker (T4.2), Data Fabric (T4.2)

Evaluation means	Counting only the real stakeholders that have already deployed any aerOS domain, regardless of the number of IE involved or its purpose (testing or production environment).		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0***	***2 (40%)***	***23 (460%)***
Outcome elaboration (M38)	**Pilot 1 – 6 Stakeholders** - **NASERTIC** (Infrastructure provider): **1** As the infrastructure provider for at least the first PoC phases and for each of the pilot 1 scenarios. - **SSF** (Scenario1) – **1**: aerOS has been deployed in the SSF on multiple domains - **INNOVALIA** (Scenario2) – **1** aerOS has been deployed in the Innovalia domain. - **SIEMENS** (Scenario3) – **1** stakeholder. aerOS deployed in the TechHall of SIEMENS Nuremberg and in the Testbed of SIEMENS Munich. - **POLIMI & MADE** (Scenario4) – **2**. An aerOS domain has been deployed on both POLIMI and MADE **Pilot 2 – 2** Stakeholders **CLOUDFERRO** has deployed 3 aerOS domains and **ELECTRUM** 1 domain. **Pilot 3 – 2** Stakeholder. The Entry domain on **TTC** and the cloud domain (Provided by **JOHNDEERE** have been deployed. **Pilot 4 – 3** Stakeholder. 3 interconnected domains have been deployed. 1 on each partner premises. **PRODEVELOP, EUROGATE,** and **CUT**. **Pilot 5 – 2** Stakeholder**:** aerOS domain deployed on both **COSMOTE** and **NSCRD** (one for each pilot scenario). Aside from the pilots of the project, **8 Open call companies** deployed aerOS as well. - 		

KPI 1.10.3 # Energy consumption & e-waste reduction in aerOS adopters

Table 81: KPI 1.10.3 # Energy consumption & e-waste reduction in aerOS adopters

<table>
<tr><td>KPI ID number and partner resp.</td><td colspan="3">KPI 1.10.3</td></tr>
<tr><td>KPI Name</td><td colspan="3"># Energy consumption & e-waste reduction in aerOS adopters</td></tr>
<tr><td>Description</td><td colspan="3">The adopters of the aerOS platform will be asked to provide information on energy consumption and e-waste compared to their baseline operation.</td></tr>
<tr><td>Motivation</td><td colspan="3">aerOS will not only support productivity enhancement of European companies to increase economic growth across the EU, but will also support tackling relevant social challenges, including energy consumption and e-waste. As part of the ability to manage such urgent social challenges, it is important to show demonstrated figures towards this direction coming from the stakeholders adopting the platform.</td></tr>
<tr><td>Target value</td><td colspan="3">2% to 10%</td></tr>
<tr><td>Prerequisites</td><td colspan="3">Integration is complete, and aerOS platform has been embraced by adopters.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">All core aerOS components</td></tr>
<tr><td>Evaluation means</td><td colspan="3">Collecting data from adopters about energy consumption and e-waste before and after integrating the aerOS platform. To do so, a 4-step methodology per adopter is followed, namely:
• Baseline Energy and waste consumption analysis based on literature (adopter specific)
• Validation of the infrastructure that will be used for KPIs measurements
• KPI measurement campaigns and
• KPI collection and analysis (comparison with baseline and target values)</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td>Use case specific value measured in the baseline scenario with no aerOS services</td><td>N/A</td><td>• Pilot 3: 54% power reduction.
• Pilot 5: 15% power reduction (330W to 280W)</td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">The purpose of KPI 1.10.3 was to demonstrate the ability of aerOS adopters to save energy and e-waste. This KPI was approached in pilot level and almost all the pilots could provide data for both of the plots of this KPI and some of them were successfully achieved and surpassed at the pilot implementation level the traget valu, even if a final total value for all partners was not possible. The quantifiable results from the pilots confirm the platform's environmental impact: Pilot 3 logged a substantial 54% reduction in power usage for its edge device by smartly distributing workload on the edge and cloud continuum. Moreover, Pilot 5 achieved a 15% reduction in power usage through a simplification of infrastructure from six to four virtual machines, a reduction that not only achieves but exceeds the KPI's upper goal value of 10%. This reduction of resources on Pilot 5 went to directly reduce hardware requirements as well as, consequently, produce less electronic waste. In addition to these metrics, Pilot 1 also featured an optimized, energy-efficient physical infrastructure design, and Pilot 2 reinforced e-waste reduction positively by adopting a green IT practice of recycling old server infrastructure. The challenge to report a single final value of this KPI was related to two primary technical limitations:</td></tr>
</table>

1. **Baseline Heterogeneity (Pilot 2 and Pilot 4):** In cases like Pilot 2 and 4, the use case were *newly deployed* within the aerOS framework, meaning a true "pre-aerOS" baseline measurement for energy consumption was not available.
2. **Technical Constraints (Pilot 4):** The specialized, low-power ARM-based architecture and Linux kernel used in Pilot 4's Industrial Edge (IEs) prevented the reliable estimation of energy consumption using the proposed standard tools (Kpler, Scaphandre, etc.), making a quantifiable before-and-after comparison impossible.

Beside these challenges, other pilots managed to provide data that confirmed double-digit savings in energy and guaranteed a commitment to resource efficiency and e-waste minimization.

Pilot 1:

From Pilot 1 perspective, only sub-Pilot 1.1 could provide data on energy consumption. More specifically, before the implementation of aerOS, the setup of the 3D-Printer Farm had the robot positioned in the center of a cube-shaped arrangement, moving vertically between multiple levels of printers placed all around it. This configuration required longer movements, higher energy consumption, and created uneven airflow and heat distribution. After optimization, all printers were repositioned on one side of the robot, allowing it to access each printer directly without vertical motion. The entire system is now integrated inside a container, ensuring better thermal control, reduced motion energy, and improved efficiency monitoring through Node-RED dashboards managed by aerOS. Unfortunately, no data on e-waste could be provided.

Pilot 2:

Pilot 2 could not include any pre-aerOS measurements of energy consumption, since the pilot use case was deployed within the aerOS framework. The measurements related to the energy consumption of Pilot 2 are provided and described in the context of KPI 2.2.1. Regarding e-waste, Pilot 2 reuses servers from a previous project, minimizing the need for new hardware procurement and reducing e-waste by avoiding the disposal or recycling of unused servers. By reusing existing infrastructure and optimizing CPU usage, the pilot contributes to sustainable IT practices, lowering the carbon footprint associated with e-waste.

Pilot 3:

Pilot 3 leverages the edge-cloud continuum within the aerOS ecosystem by utilizing the TTControl ECU as an edge device for pre-processing image data. The cloud domain is employed for executing AI algorithms to generate spatially accurate prescription maps for herbicide application. Executing the entire processing pipeline solely on the ECU would result in a power consumption of 48 Wh. By distributing the workload between the edge and cloud, the ECU’s power consumption is reduced to 22 Wh, representing **a reduction of approximately 54%**. Pilot 3 could not report any data related to e-waste.

Pilot 4:

Unfortunately, Pilot 4 IEs are built under ARM architecture with Linux kernel, so none of the proposed tools (Kpler, Scaphandre, or embedded self-awareness functionalities and Linux commands) were able to estimate the energy being consumed. Thus, the results before and after aerOS platform is deployed on them cannot be provided.

Pilot 5:

	Pilot 5 required six virtual machines (1 for MQTT, 1 for influxDB, 1 for Grafana, 1 for the GUI, 1 for the AI models and 1 VM for the recommender) before implementing aerOS to ensure clear separation of concerns and optimized resource allocation across functional domains such as ingestion, processing, and visualization. Deploying components across multiple machines instead of a single or fewer VMs was necessary to avoid resource contention, improve system stability, and manage workloads more efficiently,especially under high data throughput. These six VMs were measured to consume approximately **330 watts** in total. After transitioning to the aerOS architecture, the system now operates with just **four VMs** consuming **280 watts**, resulting in a measurable reduction in power usage. Additionally, reducing the number of required VMs contributes to lower hardware demands and consequently less electronic waste (e-waste). This leaner setup aligns with more sustainable computing practices. Furthermore, by leveraging KubeEdge, we eliminated the need for UPS systems dedicated to network continuity. Even if the internet connection is lost, the edge nodes continue collecting and storing measurements locally, enhancing fault tolerance and system resilience while reducing infrastructure overhead.

C. Appendix C. Pilot and Overall KPIs

Pilot 1 Data-driven cognitive production lines

KPI 2.1.1 Production process accuracy

Table 82: KPI 2.1.1 Production process accuracy

KPI ID number and partner resp.	**KPI 2.1.1**		
KPI Name	**Production process accuracy**		
Description	The accuracy of the process based on digital and virtual part analysis – online and offline.		
Motivation	The quality of the process is based on the adequate selection of the dimensional quality control instrumentation and the optimisation of the quality control strategy and configuration of the manufacturing equipment.		
Target value	10% increase		
Prerequisites	N/A		
aerOS components (task)	HLO (T.3.3), LLO (k8s, Docker) (T3.3), Keycloak (T3.4), KrakenD (T3.4), Self* (T3.5), aerOS portal (T4.6)		
Evaluation means	Comparison of point-cloud datasets generated by RobotLink and ScanLink before and after aerOS deployment, analysed through metrology software logs to quantify deviations and accuracy improvements under identical environmental conditions.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**

Measured value (% achieved)	***Dependent on product GD&T complexity***	***N/A***	***9.2% increase (92%)***
Outcome elaboration (M38)	The RobotLink (RL) and ScanLink (SL) modules operate as integrated software components that feed data into the Data Assembler. Together, they synchronize the measurements obtained from the touch probe and the OptiScan system to generate a unified cloud of points. While the touch probe acquires data point by point—later extrapolating them to define geometric features—the OptiScan continuously scans the surface, detects logical patterns, and reconstructs the geometries directly. Both systems are therefore complementary, combining precision in discrete probing with the broader spatial coverage of optical scanning to achieve a high-accuracy point cloud. However, the outermost points of the cloud remain the most susceptible to error. These boundary measurements are more sensitive to environmental variations—especially temperature changes and sudden service interruptions, which can momentarily desynchronize the two data streams. When a crash or measurement halt occurs, misalignment arises because the timestamps of RL and SL diverge (e.g., RL-position x, y, z and vector i, j, k versus SL-position x', y', z' and vector i', j', k'). Before aerOS, the detection and recovery of such crashes took nearly a minute, during which positional data could drift. With aerOS—and particularly through its **Self-healing module**—service restoration is nearly instantaneous, keeping both RL and SL synchronized and preserving the accuracy of the point cloud, especially along its perimeter. To assess this KPI, the same measurement sequence was executed on two different gages (Vulkan and Spark) and under two conditions, with and without aerOS. Each setup was repeated **7** times, both remotely and client premises, having these last factor no significant differences in baseline accuracy with and without aerOS in place. The selected part to be measured is the same one as for KPI 2.1.3 and under the same conditions (temperature and connectivity) For each test series, a controlled disturbance was intentionally introduced to simulate realistic disruptions: - Probe collision (×2) - Optical obstruction (×2) - Arm overreach - Interruption of the OPC UA connection - Manual software crash by overloading M3 As expected, the presence of aerOS significantly reduced service downtime, which translated in to faster recovery and re-synchronization, the **global measurement accuracy increased by an average of 9.2%** compared with the pre-aerOS configuration.		

KPI 2.1.2 Digital service programming time

Table 83: KPI 2.1.2 Digital service programming time

KPI ID number and partner resp.	**KPI 2.1.2**
KPI Name	**Digital service programming time**

Description	The time it takes to program quality control services and routines based on traditional monolithic (client/server) vs computing continuum platform.		
Motivation	Autonomous operation of zero-defect manufacturing services and dimensional quality control demand that automation services are highly available to ensure synchronised and safe.		
Target value	2 days		
Prerequisites	N/A		
aerOS components (task)	HLO (T.3.3), LLO (k8s, Docker) (T3.3), Keycloak (T3.4), KrakenD (T3.4), Self* (T3.5), aerOS portal (T4.6)		
Evaluation means	Measurement of total configuration time using M3 and OPC-UA manual service logs, comparing manual on-site setup durations with automated, containerized deployments managed through the aerOS orchestration layer.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***2 weeks***	***< 2weeks (10 business days)***	***2.3 business days (96%)***
Outcome elaboration (M38)	In order to address this KPI, the traditional service programming time is directly compared against the enhanced aerOS environment (see table below). The comparison demonstrates how aerOS metaOS and remote access and operations have drastically reduced the overall duration—from **10 days to 2.3 days**, representing **96% of the target**. In the traditional process, each step required physical presence and manual setup. The metrologist had to travel to the client's premises, manually check and update OS, and perform individual installations of metrology software (M3) per machine. In contrast, aerOS introduces **remote accessibility**, **containerized services**, and **centralized management**, which shorter and makes more flexible most of these operations. Key gains were achieved through: - **Remote access and deployment**, which completely removed travel time and enabled remote OS updates and machine configuration. - **Containerized and standardized services**, replacing complex on-site software installation with automated deployment via the aerOS Management Portal. - **Centralized data and service management**, eliminating redundant database setups and allowing a unified configuration across devices. - **Remote training, testing, and debugging**, significantly reducing on-site time while maintaining quality assurance through iterative and continuous supported validation. Only the **initial CMM installation and calibration** still require on-site presence, although even this phase is now partially fastened through remote calibration. Overall, aerOS transforms a mainly manual and sequential workflow into a **digital and automated process inside the aerOS enabled Continuum**, resulting in faster setup		

and configuration, reduced human intervention, and consistent performance across sites.

Step in QC cycle	Traditional service programming - Days	aerOS enabled programming - Days	Dif. (+)
Initial travel to site	1	0	1
OS installation/update	1	0	1
CMM installation and calibration	1.5	1	0.5
M3 (Metrology SW) Installation	1	0.25	0.75
Service configuration	2	0.25	1.75
DB setup	1	0.1	0.9
Training	1	0.25	0.75
Initial test run	1	0.1	0.9
Debugging and corrections	1	0.1	0.9
Documentation	0.25	0.25	0
Total (days)	**10.75**	**2.3**	**8.45**
Time to program QC services and routines. Baseline 10 days	10.75		
Time to program QC services and routines. Target 2 days	2.30		

KPI 2.1.3 Dimensional quality control productivity

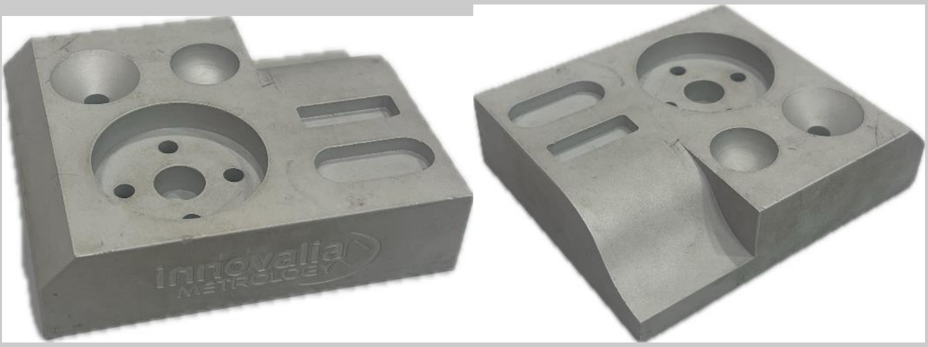
Table 84: KPI 2.1.3 Dimensional quality control productivity

<table>
<tr><th>KPI ID number and partner resp.</th><th colspan="3">KPI 2.1.3</th></tr>
<tr><td>KPI Name</td><td colspan="3">Dimensional quality control productivity</td></tr>
<tr><td>Description</td><td colspan="3">The time to perform a quality control cycle (specification, programming and execution)</td></tr>
<tr><td>Motivation</td><td colspan="3">Zero-defect manufacturing services and dimensional quality control are usually manually driven processes locked to users and machines being physically interacting. The productivity decoupling task programming, dispatching and execution of metrology routines can increase factory productivity.</td></tr>
<tr><td>Target value</td><td colspan="3">5 parts/hour</td></tr>
<tr><td>Prerequisites</td><td colspan="3">Special proprietary machine (Custom Positioning and Palletizing system)</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">HLO (T.3.3), LLO (k8s, Docker) (T3.3), Keycloak (T3.4), KrakenD (T3.4), Self* (T3.5), aerOS portal (T4.6)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">Calculation based on the number of parts measured per hour in real production cycles, using time-stamped inspection logs and aerOS monitoring dashboards to compare baseline and M36 performance.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
</table>

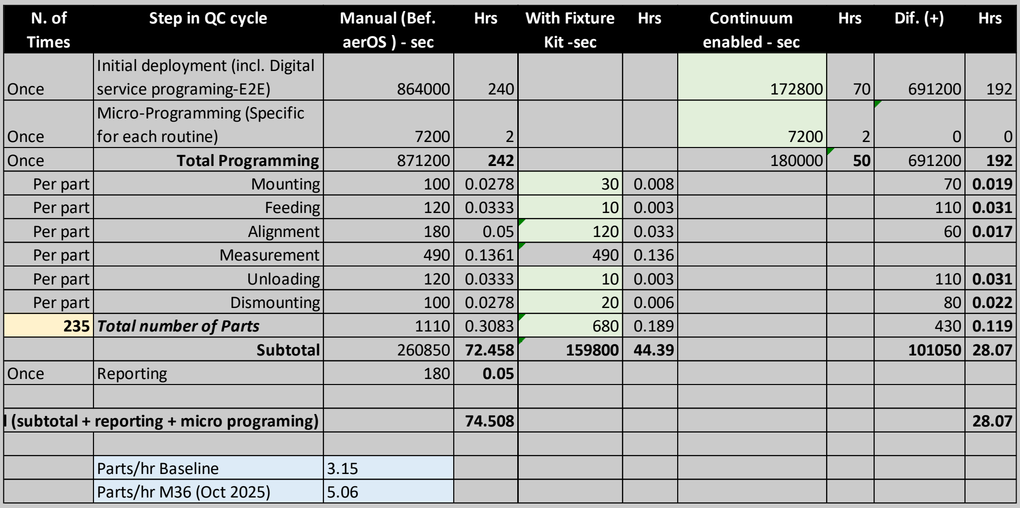
Measured value (% achieved)	***3 parts/hour (depending on GD&T complexity)***	***2 parts/hour (40%)***	***5.06 parts/hour (101%)***

Outcome elaboration (M38)

For the measurement, the next part has been selected due to the heterogeneity of the model, containing all the common geometries usually encountered by the metrologist, being therefore a great example for addressing this KPI. Is easy to see curves, cylinders, drill holes, spheres or sharp edges.

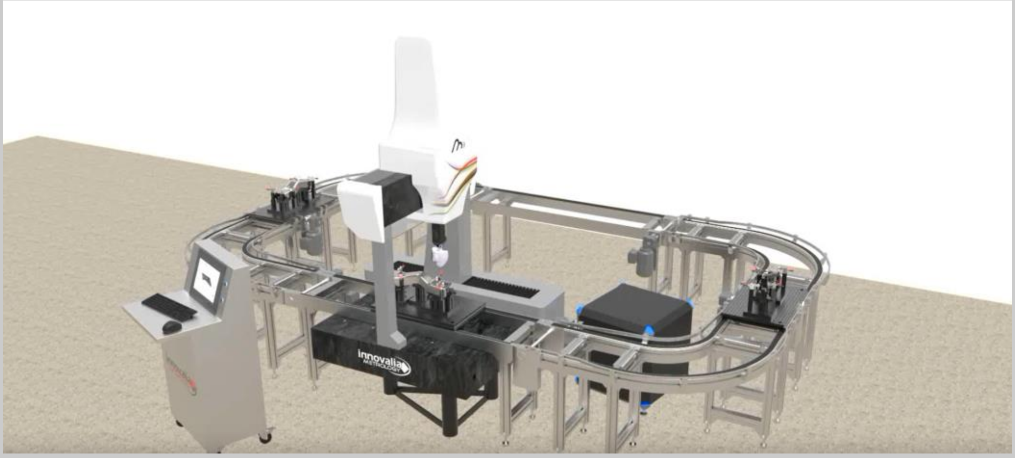


The following table list the main steps and their times during the measurement phase. For the calculation, there is a first programming step (*Micro-programming*) that only happens once for the whole measurement project, followed by 6 steps (*Mounting, Feeding, Alignment, Measurement, Unloading, and Dismounting*) that are repeated as many times as the total number of parts measured (235 parts for this project), and is within this repetitive cycle where the productivity increase takes place.



N. of Times	Step in QC cycle	Manual (Bef. aerOS) - sec	Hrs	With Fixture Kit -sec	Hrs	Continuum enabled - sec	Hrs	Dif. (+)	Hrs
Once	Initial deployment (incl. Digital service programing-E2E)	864000	240			172800	70	691200	192
Once	Micro-Programming (Specific for each routine)	7200	2			7200	2	0	0
Once	**Total Programming**	871200	**242**			180000	**50**	691200	**192**
Per part	Mounting	100	0.0278	30	0.008			70	**0.019**
Per part	Feeding	120	0.0333	10	0.003			110	**0.031**
Per part	Alignment	180	0.05	120	0.033			60	**0.017**
Per part	Measurement	490	0.1361	490	0.136				
Per part	Unloading	120	0.0333	10	0.003			110	**0.031**
Per part	Dismounting	100	0.0278	20	0.006			80	**0.022**
235	***Total number of Parts***	1110	0.3083	680	0.189			430	**0.119**
	Subtotal	260850	**72.458**	**159800**	**44.39**			**101050**	**28.07**
Once	Reporting	180	**0.05**						
l (subtotal + reporting + micro programing)			**74.508**						**28.07**
	Parts/hr Baseline	3.15							
	Parts/hr M36 (Oct 2025)	5.06							

Before aerOS, each part had to be manually checked, attached to the platform, and aligned with the Zero-Reference Axis, while now, using the palletizing system depicted below, every pre and post measurement steps are shortened. The mounting-dismounting (attaching the part to the base) and feeding-unfeeding (inserting and removing the set part + base to the measurement area) are dramatically boosted. On top of that, thanks to the automatic feeding, the relative positioning of the base axis with the Gage Zero-reference axis is faster and more accurate, speeding up the alignment step.

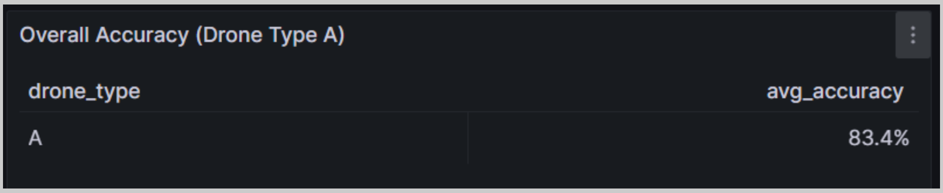
	 As a result, the total number of parts measured per hour increase substantially, optimizing the whole process. These results highly depend on the part chosen and the number of times measured, therefore for more simpler or higher volumes, the productivity increase will be even higher.

KPI 2.1.4 Accuracy of the CO2-footprint prediction (%)

Table 85: KPI 2.1.4 Accuracy of the CO2-footprint prediction (%)

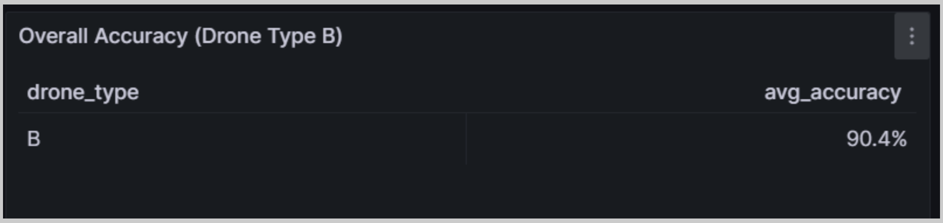
KPI ID number and partner resp.	**KPI 2.1.4**
KPI Name	**Accuracy of CO2-footprint prediction (%)**
Description	This KPI represents the fidelity of the value obtained for the CO2 footprint
Motivation	Based on the methods used for CO2 footprint calculation and the data captured from the shopfloor the accuracy of the PCF value is higher and hence the impact and costs associated with product-related emissions lowered.
Target value	>80%
Prerequisites	To complete the predicted and actual CO2 footprint for each product, the IEs setup must be completed, Node-Red configured for CO2 emissions calculation, the data collection tool set up and the network and APIs configured.
aerOS components (task)	Networking (T3.1), API & Low Code tools (T3.2), Orchestration (T3.3), T3.5 (Self-*), T4.2 (Data Fabric), aerOS Portal (T4.6)
Evaluation means	The actual CO2 footprint of each product is based on the production data sent by the various IEs involved in the scenario, which are then processed using the Node-Red tool to obtain the final PCF value. This value is collected for each new order, with at least one test carried out every two to three weeks (1 to 2 tests per month). It is then compared with the predicted CO2 emissions calculated before production of each product, to measure the evolution of the PCF accuracy in grams. The overall accuracy of the prediction of the CO2 footprint for the entire production is estimated after

<table>
<tr><td></td><td colspan="3">aggregating the accuracy for each product and displayed in the SSF using visualization tools.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>1. CO2-footprint prediction for F330 Model and High Edition: 450 grams.</i>
<i>2. Actual CO2 Emissions for F330 Model and High Edition: 459 grams.</i>
<i>Accuracy of the predicted value is within 50% tolerance.</i></td><td><i>83.4% - drone type A</i>
<i>90.4% - drone type B</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">
We achieved an average prediction accuracy of 83.4% for Drone Type A and 90.4% for Drone Type B, exceeding the KPI target of >80%.



Overall Accuracy (Drone Type A)

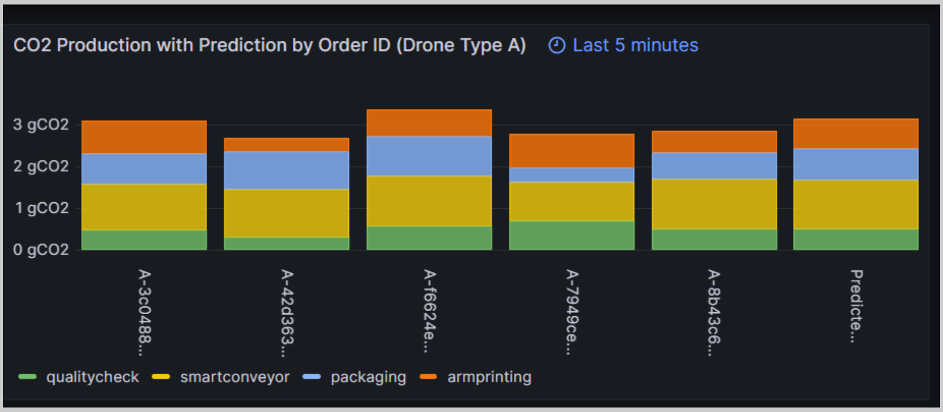
<table>
<tr><th>drone_type</th><th>avg_accuracy</th></tr>
<tr><td>A</td><td>83.4%</td></tr>
</table>



Overall Accuracy (Drone Type B)

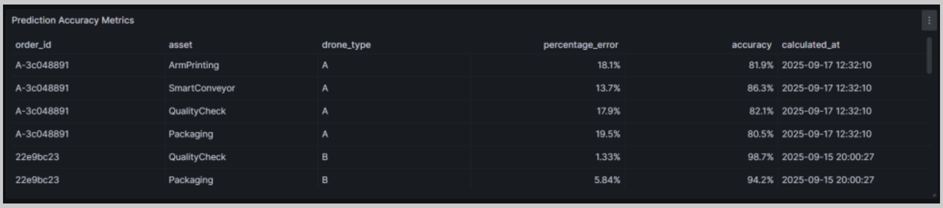
<table>
<tr><th>drone_type</th><th>avg_accuracy</th></tr>
<tr><td>B</td><td>90.4%</td></tr>
</table>

These results are computed and displayed by the co2-total-barplot-prediction function. This service:
(i) selects a lightweight forecasting model per asset and drone type, (ii) produces fresh CO₂ predictions,



</td></tr>
</table>

and (iii) calculates accuracy by matching each actual to the *nearest* prediction in time.



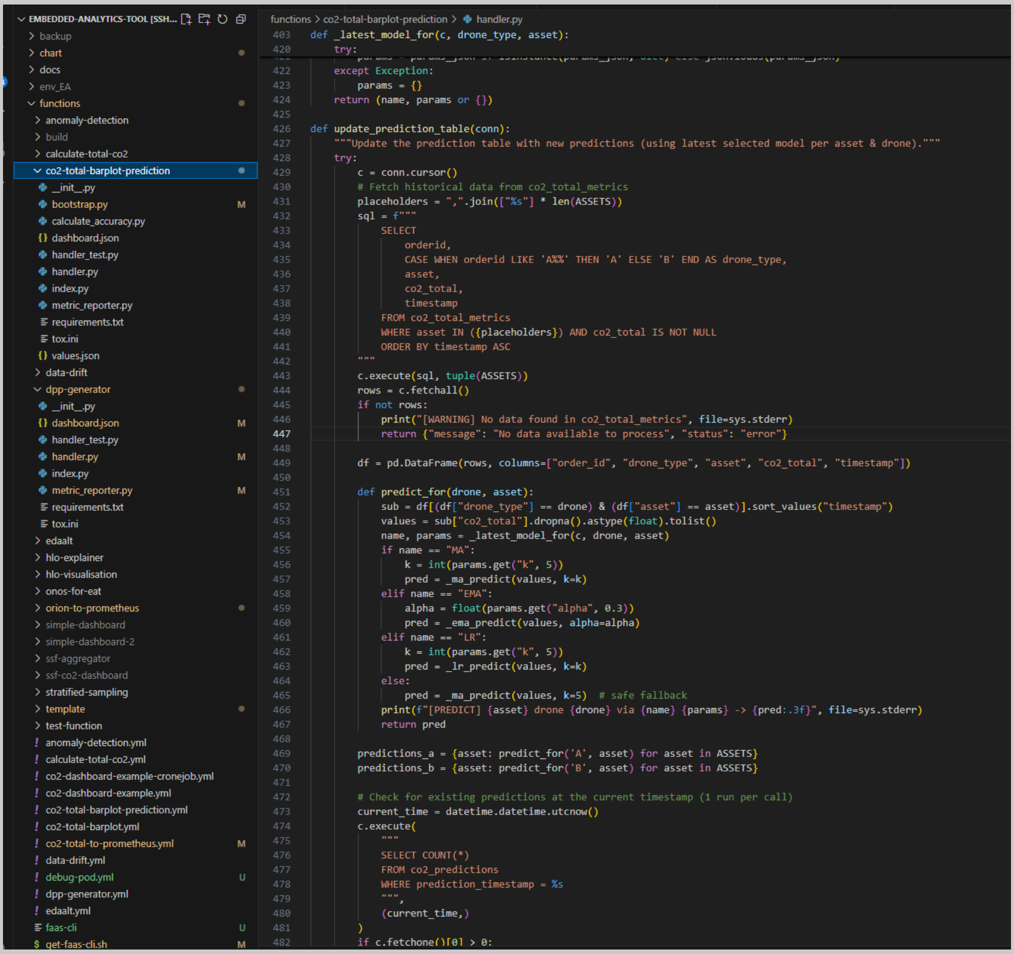
Prediction Accuracy Metrics

order_id	asset	drone_type	percentage_error	accuracy	calculated_at
A-3c048891	ArmPrinting	A	18.1%	81.9%	2025-09-17 12:32:10
A-3c048891	SmartConveyor	A	13.7%	86.3%	2025-09-17 12:32:10
A-3c048891	QualityCheck	A	17.9%	82.1%	2025-09-17 12:32:10
A-3c048891	Packaging	A	19.5%	80.5%	2025-09-17 12:32:10
22e9bc23	QualityCheck	B	1.33%	98.7%	2025-09-15 20:00:27
22e9bc23	Packaging	B	5.84%	94.2%	2025-09-15 20:00:27

The function bootstraps the Grafana dashboard and initializes the accuracy workflow at cold start, ensuring tables exist and the dashboard is (re)uploaded.

For each asset (QualityCheck, SmartConveyor, Packaging, ArmPrinting), the function inserts timestamped predictions for Types A and B, then computes absolute error, percentage error, and an ‘accuracy’ metric. Percentage error is clamped (≤20%) and accuracy is derived as *max(100 − percentage error, 80)* to provide a robust, KPI-aligned score used in the dashboard tables.

The “actual” totals come from the calculate-total-co2 pipeline, which aggregates machine telemetry (per order and asset) into co2_total_metrics; these values are then joined with predictions to produce per-order accuracy records.



```python
def _latest_model_for(c, drone_type, asset):
    try:
    except Exception:
        params = {}
    return (name, params or {})

def update_prediction_table(conn):
    """Update the prediction table with new predictions (using latest selected model per asset & drone)."""
    try:
        c = conn.cursor()
        # Fetch historical data from co2_total_metrics
        placeholders = ",".join(["%s"] * len(ASSETS))
        sql = f"""
            SELECT
                orderid,
                CASE WHEN orderid LIKE 'A%%' THEN 'A' ELSE 'B' END AS drone_type,
                asset,
                co2_total,
                timestamp
            FROM co2_total_metrics
            WHERE asset IN ({placeholders}) AND co2_total IS NOT NULL
            ORDER BY timestamp ASC
        """
        c.execute(sql, tuple(ASSETS))
        rows = c.fetchall()
        if not rows:
            print("[WARNING] No data found in co2_total_metrics", file=sys.stderr)
            return {"message": "No data available to process", "status": "error"}

        df = pd.DataFrame(rows, columns=["order_id", "drone_type", "asset", "co2_total", "timestamp"])

        def predict_for(drone, asset):
            sub = df[(df["drone_type"] == drone) & (df["asset"] == asset)].sort_values("timestamp")
            values = sub["co2_total"].dropna().astype(float).tolist()
            name, params = _latest_model_for(c, drone, asset)
            if name == "MA":
                k = int(params.get("k", 5))
                pred = _ma_predict(values, k=k)
            elif name == "EMA":
                alpha = float(params.get("alpha", 0.3))
                pred = _ema_predict(values, alpha=alpha)
            elif name == "LR":
                k = int(params.get("k", 5))
                pred = _lr_predict(values, k=k)
            else:
                pred = _ma_predict(values, k=5)  # safe fallback
            print(f"[PREDICT] {asset} drone {drone} via {name} {params} -> {pred:.3f}", file=sys.stderr)
            return pred

        predictions_a = {asset: predict_for('A', asset) for asset in ASSETS}
        predictions_b = {asset: predict_for('B', asset) for asset in ASSETS}

        # Check for existing predictions at the current timestamp (1 run per call)
        current_time = datetime.datetime.utcnow()
        c.execute(
            """
            SELECT COUNT(*)
            FROM co2_predictions
            WHERE prediction_timestamp = %s
            """,
            (current_time,)
        )
        if c.fetchone()[0] > 0:
```

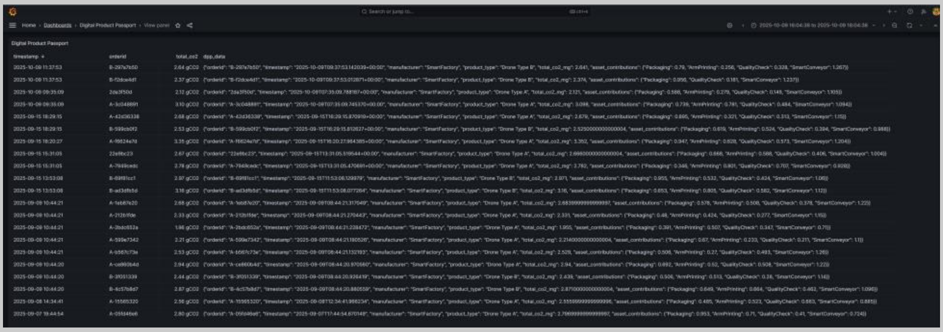
	This image shows a Python script in Visual Studio Code that handles CO2 prediction updates for drone assets. The function update_prediction_table() retrieves historical CO2 data from a database, predicts future values using selected models for each drone type, and stores results back into a predictions table. It includes error handling, data processing with pandas, and safeguards to avoid duplicate timestamp entries.

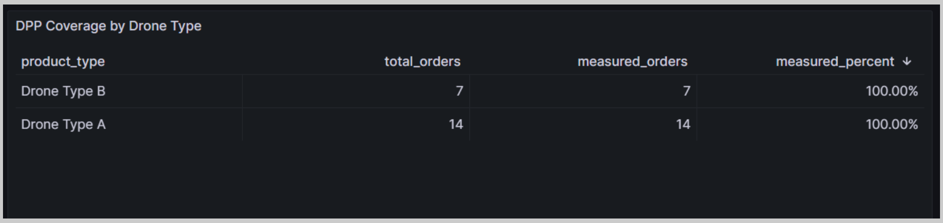
KPI 2.1.5 CO2-footprint measurement (% products)

Table 86: KPI 2.1.5 CO2-footprint measurement (% products)

KPI ID number and partner resp.	**KPI 2.1.5**		
KPI Name	**CO2-footprint measurement (% products)**		
Description	This KPI assess the number of products that can be assessed for CO2 footprint calculation (unit, batch, family level)		
Motivation	The PCF calculation and the associated DPP information for Scope 3 environments demand different level of granularity in terms of data collection and product-level calculation to meet with EU regulation.		
Target value	10% - 100%		
Prerequisites	To calculate the number of products that can be assessed for the CO2 footprint (at unit, batch or family level), the IEs configuration must be complete, Node-Red configured, the data collection tool set up and the network and APIs configured. To measure the global percentage of products that can be assessed, the aerOS runtime must be running and the web service must be ready.		
aerOS components (task)	Networking (T3.1), API & Low Code tools (T3.2), Orchestration (T3.3), T3.5 (Self-*), T4.2 (Data Fabric), aerOS Portal (T4.6)		
Evaluation means	To measure the number of products assessed for CO2 footprint calculation the count of products at unit, batch, and family levels are tracked. Data is collected monthly, noting how many products are assessed in each category. ERP systems and data visualization tools are used for tracking and reporting. The performance is measured by the total counts and percentage coverage of assessments.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*N/A*	*1 out of 20 drones assessed (5% at unit level)*	*100%*
Outcome elaboration (M38)	The measurement pipeline has reached full coverage (100%): we now measure and persist the total CO_2 footprint for every machine and for the drone Types A and B. The calculate-total-co2 function aggregates the relevant telemetry and process metadata per order into a normalized "total CO_2" record, while the dpp-generator		

publishes these records as standardized data products (with order IDs, timestamps, and asset/drone identifiers) to ensure they are queryable, usable across dashboards and reports. Together, these services enhance the energy-data transparency of our drone products and the machines involved in the assembly process.

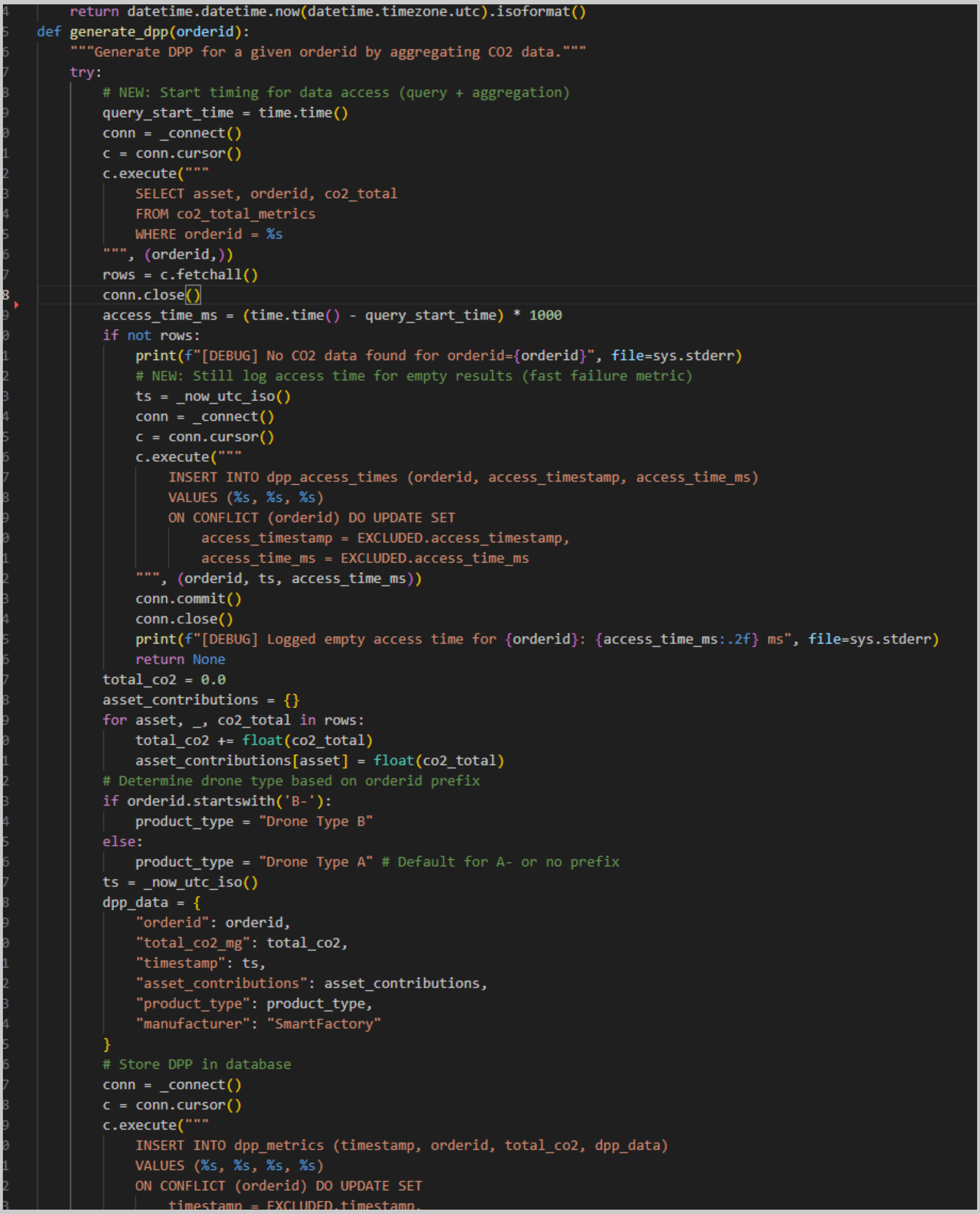




DPP Coverage by Drone Type

product_type	total_orders	measured_orders	measured_percent ↓
Drone Type B	7	7	100.00%
Drone Type A	14	14	100.00%

As a result, the KPI “CO₂-footprint measurement (% products)” is met at 100% of products/orders covered at M38, with measurements stored reliably for historical analysis and continuous monitoring.



```
    return datetime.datetime.now(datetime.timezone.utc).isoformat()
def generate_dpp(orderid):
    """Generate DPP for a given orderid by aggregating CO2 data."""
    try:
        # NEW: Start timing for data access (query + aggregation)
        query_start_time = time.time()
        conn = _connect()
        c = conn.cursor()
        c.execute("""
            SELECT asset, orderid, co2_total
            FROM co2_total_metrics
            WHERE orderid = %s
        """, (orderid,))
        rows = c.fetchall()
        conn.close()
        access_time_ms = (time.time() - query_start_time) * 1000
        if not rows:
            print(f"[DEBUG] No CO2 data found for orderid={orderid}", file=sys.stderr)
            # NEW: Still log access time for empty results (fast failure metric)
            ts = _now_utc_iso()
            conn = _connect()
            c = conn.cursor()
            c.execute("""
                INSERT INTO dpp_access_times (orderid, access_timestamp, access_time_ms)
                VALUES (%s, %s, %s)
                ON CONFLICT (orderid) DO UPDATE SET
                    access_timestamp = EXCLUDED.access_timestamp,
                    access_time_ms = EXCLUDED.access_time_ms
            """, (orderid, ts, access_time_ms))
            conn.commit()
            conn.close()
            print(f"[DEBUG] Logged empty access time for {orderid}: {access_time_ms:.2f} ms", file=sys.stderr)
            return None
        total_co2 = 0.0
        asset_contributions = {}
        for asset, _, co2_total in rows:
            total_co2 += float(co2_total)
            asset_contributions[asset] = float(co2_total)
        # Determine drone type based on orderid prefix
        if orderid.startswith('B-'):
            product_type = "Drone Type B"
        else:
            product_type = "Drone Type A" # Default for A- or no prefix
        ts = _now_utc_iso()
        dpp_data = {
            "orderid": orderid,
            "total_co2_mg": total_co2,
            "timestamp": ts,
            "asset_contributions": asset_contributions,
            "product_type": product_type,
            "manufacturer": "SmartFactory"
        }
        # Store DPP in database
        conn = _connect()
        c = conn.cursor()
        c.execute("""
            INSERT INTO dpp_metrics (timestamp, orderid, total_co2, dpp_data)
            VALUES (%s, %s, %s, %s)
            ON CONFLICT (orderid) DO UPDATE SET
                timestamp = EXCLUDED.timestamp,
```

This screenshot shows a Python function generate_dpp(orderid) that aggregates CO2 data for a specific order ID and stores a DPP in our database. It queries historical CO2 metrics, calculates total emissions and asset contributions, identifies the drone type based on the order ID prefix, and logs access times for performance tracking. Finally, it saves or updates the computed DPP record in the dpp_metrics table.

KPI 2.1.6 CO2 emissions reduction (kg)

Table 87: KPI 2.1.6 CO2 emissions reduction (kg)

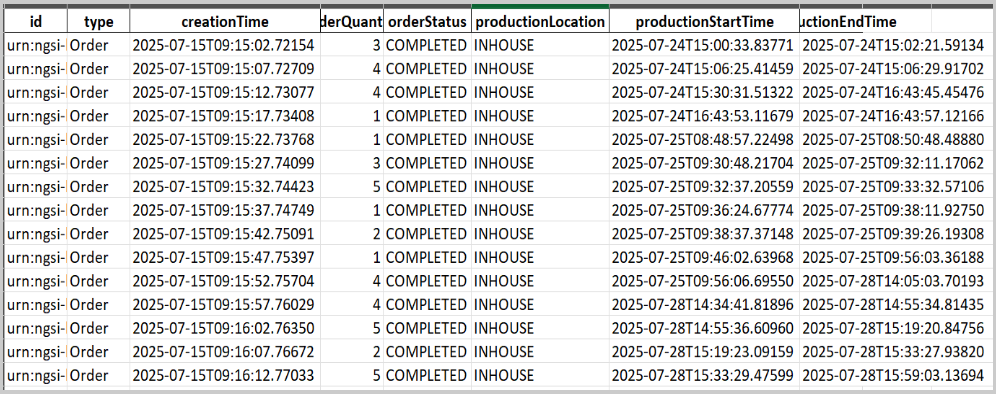
KPI ID number and partner resp.	**KPI 2.1.6**
KPI Name	**CO2 emissions reduction (kg)**

<table>
<tr><td>Description</td><td colspan="3">aerOS system should permit with AI/ML component contribution to optimize travels of AGV and infer CO2 emission reduction</td></tr>
<tr><td>Motivation</td><td colspan="3">The PCF should contribute to industrial competitiveness</td></tr>
<tr><td>Target value</td><td colspan="3"><20%</td></tr>
<tr><td>Prerequisites</td><td colspan="3">Fully integrated system</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">KrakenD (T3.1), Ingress (T3.1), Ingress&KrakenD conf (T3.1), CertManager&LetsEncrypt (TLS) (T3.1), FDQN (T3.1), NAT Capable (T3.1), OpenAPI (T3.2), AsyncAPI (T3.2), Low-Code (T3.2), HLO (T.3.3), LLO (k8s, Docker) (T3.3), KeyCloack (T3.4), KrakenD (T3.4), OpenLDAP (T3.4), Self-awareness (T3.5), Self-orchestration (T3.5), Self-diagnose (T3.5), Self-security (T3.5), Self-healing (T3.5), Self Configuration (T3.5), Self-API (T3.5), Data-Interoperability (T4.1), LDAP (T4.2), EAT (T4.4), Trust SCO. (T4.5), IOTA (T4.5), Portal (T4.6).</td></tr>
<tr><td>Evaluation means</td><td colspan="3">The methodology used to evaluate this KPI will be as follow:
• Measure the energy consumption of the AGV per travel in kWh.
• Find correlation with CO2 emission based on the electricity grid emission factor. This factor represents the average amount of CO2 emitted per unit of electricity produced in the region where the AGV operates, in our case Lombardy, Italy.
CO2 emissions (kg) = Electricity consumption (kWh) * Grid emission factor Lombardy(IT) (kg CO2/kWh).</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>0%</i></td><td><i>N/A</i></td><td><i>39.42%</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">In order to calculate this KPI, POLIMI and MADE first ran the production line in baseline state to collect data for baseline generation. In order to maintain the data storage modalities, the team also persisted this data to the Orion-LD but in this case the optimization was bypassed by creating a dedicated docker image where the orders coming from the order generator are directly passed to the MADE LEA system without any changes or clubbing of orders.
In the second phase the optimization was turned on. Again, in this case also all the production data like order receive times, production start and end times etc. was captured inside Orion-LD. Finally, we used a Python script to query all the data and save this in an excel format.</td></tr>
</table>

```python
import requests
import pandas as pd
base_url = "http://localhost:1026/ngsi-ld/v1/entities"
query_url = f"{base_url}/?type=Order&limit=1000&q=productionStartTime!=\"timestamp\""
headers = {
    'Link': '<http://context:5051/ngsi-context.jsonld>; rel="http://www.w3.org/ns/json-ld#cont
}

response = requests.get(query_url, headers=headers)
if response.status_code == 200:
    entities = response.json()
    print(f"Fetched {len(entities)} entities")
    data = []
    for e in entities:
        row = {
            "id": e.get("id"),
            "type": e.get("type"),
            "creationTime": e.get("creationTime", {}).get("value", ""),
            "orderQuantity": e.get("orderQuantity", {}).get("value", ""),
            "orderStatus": e.get("orderStatus", {}).get("value", ""),
            "productionLocation": e.get("productionLocation", {}).get("value", ""),
            "productionStartTime": e.get("productionStartTime", {}).get("value", ""),
            "productionEndTime": e.get("productionEndTime", {}).get("value", ""),
            "orderOutsourceTime": e.get("orderOutsourceTime", {}).get("value", ""),
            "productReceiveTime": e.get("productReceiveTime", {}).get("value", ""),
            "totalProductionTime": e.get("totalProductionTime", {}).get("value", "")
        }
        data.append(row)
    df = pd.DataFrame(data)
    df.to_excel("orders.xlsx", index=False)
    print("Saved entities to orders.xlsx")
else:
    print(f"Failed to fetch entities: {response.status_code} {response.text}")
```

Python script query all the data and save in an excel format.

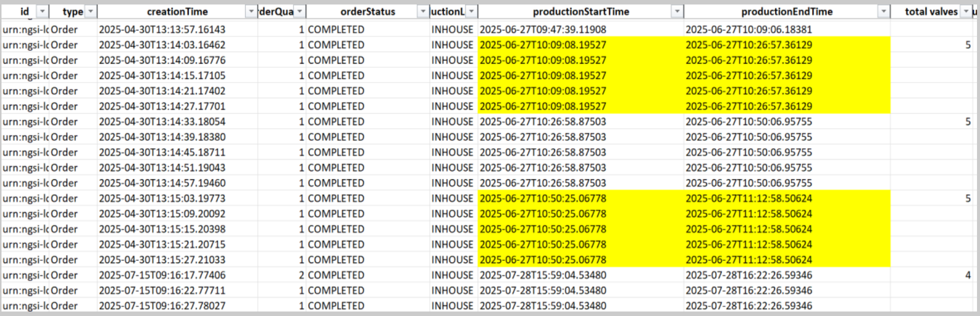
For baseline case, it is possible to directly use the order quantity to calculate the average number of valves per travel from the order quantity.



id	type	creationTime	derQuant	orderStatus	productionLocation	productionStartTime	uctionEndTime
urn:ngsi-	Order	2025-07-15T09:15:02.72154	3	COMPLETED	INHOUSE	2025-07-24T15:00:33.83771	2025-07-24T15:02:21.59134
urn:ngsi-	Order	2025-07-15T09:15:07.72709	4	COMPLETED	INHOUSE	2025-07-24T15:06:25.41459	2025-07-24T15:06:29.91702
urn:ngsi-	Order	2025-07-15T09:15:12.73077	4	COMPLETED	INHOUSE	2025-07-24T15:30:31.51322	2025-07-24T16:43:45.45476
urn:ngsi-	Order	2025-07-15T09:15:17.73408	1	COMPLETED	INHOUSE	2025-07-24T16:43:53.11679	2025-07-24T16:43:57.12166
urn:ngsi-	Order	2025-07-15T09:15:22.73768	1	COMPLETED	INHOUSE	2025-07-25T08:48:57.22498	2025-07-25T08:50:48.48880
urn:ngsi-	Order	2025-07-15T09:15:27.74099	3	COMPLETED	INHOUSE	2025-07-25T09:30:48.21704	2025-07-25T09:32:11.17062
urn:ngsi-	Order	2025-07-15T09:15:32.74423	5	COMPLETED	INHOUSE	2025-07-25T09:32:37.20559	2025-07-25T09:33:32.57106
urn:ngsi-	Order	2025-07-15T09:15:37.74749	1	COMPLETED	INHOUSE	2025-07-25T09:36:24.67774	2025-07-25T09:38:11.92750
urn:ngsi-	Order	2025-07-15T09:15:42.75091	2	COMPLETED	INHOUSE	2025-07-25T09:38:37.37148	2025-07-25T09:39:26.19308
urn:ngsi-	Order	2025-07-15T09:15:47.75397	1	COMPLETED	INHOUSE	2025-07-25T09:46:02.63968	2025-07-25T09:56:03.36188
urn:ngsi-	Order	2025-07-15T09:15:52.75704	4	COMPLETED	INHOUSE	2025-07-25T09:56:06.69550	2025-07-28T14:05:03.70193
urn:ngsi-	Order	2025-07-15T09:15:57.76029	4	COMPLETED	INHOUSE	2025-07-28T14:34:41.81896	2025-07-28T14:55:34.81435
urn:ngsi-	Order	2025-07-15T09:16:02.76350	5	COMPLETED	INHOUSE	2025-07-28T14:55:36.60960	2025-07-28T15:19:20.84756
urn:ngsi-	Order	2025-07-15T09:16:07.76672	2	COMPLETED	INHOUSE	2025-07-28T15:19:23.09159	2025-07-28T15:33:27.93820
urn:ngsi-	Order	2025-07-15T09:16:12.77033	5	COMPLETED	INHOUSE	2025-07-28T15:33:29.47599	2025-07-28T15:59:03.13694

Number of valves per travel from the order quantity

For the Improved case, it was needed to first club the orders in which production start time and end time are identical (meaning these were clubbed together) to create a new column showing the total number of valves carried in a trip. Figure below shows a snapshot of this process



id	type	creationTime	derQua	orderStatus	uctionL	productionStartTime	productionEndTime	total valves	u
urn:ngsi-lc	Order	2025-04-30T13:13:57.16143	1	COMPLETED	INHOUSE	2025-06-27T09:47:39.11908	2025-06-27T10:09:06.18381		
urn:ngsi-lc	Order	2025-04-30T13:14:03.16462	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129	5	
urn:ngsi-lc	Order	2025-04-30T13:14:09.16776	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129		
urn:ngsi-lc	Order	2025-04-30T13:14:15.17105	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129		
urn:ngsi-lc	Order	2025-04-30T13:14:21.17402	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129		
urn:ngsi-lc	Order	2025-04-30T13:14:27.17701	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129		
urn:ngsi-lc	Order	2025-04-30T13:14:33.18054	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755	5	
urn:ngsi-lc	Order	2025-04-30T13:14:39.18380	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755		
urn:ngsi-lc	Order	2025-04-30T13:14:45.18711	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755		
urn:ngsi-lc	Order	2025-04-30T13:14:51.19043	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755		
urn:ngsi-lc	Order	2025-04-30T13:14:57.19460	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755		
urn:ngsi-lc	Order	2025-04-30T13:15:03.19773	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624	5	
urn:ngsi-lc	Order	2025-04-30T13:15:09.20092	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624		
urn:ngsi-lc	Order	2025-04-30T13:15:15.20398	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624		
urn:ngsi-lc	Order	2025-04-30T13:15:21.20715	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624		
urn:ngsi-lc	Order	2025-04-30T13:15:27.21033	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624		
urn:ngsi-lc	Order	2025-07-15T09:16:17.77406	2	COMPLETED	INHOUSE	2025-07-28T15:59:04.53480	2025-07-28T16:22:26.59346	4	
urn:ngsi-lc	Order	2025-07-15T09:16:22.77711	1	COMPLETED	INHOUSE	2025-07-28T15:59:04.53480	2025-07-28T16:22:26.59346		
urn:ngsi-lc	Order	2025-07-15T09:16:27.78027	1	COMPLETED	INHOUSE	2025-07-28T15:59:04.53480	2025-07-28T16:22:26.59346		

Finally, from both these values, the average values are taken for both the conditions and then calculate the reduction in terms of travels per valve. Which is reported in the following table. This same reduction directly impacts the CO2 production.

Reduction directly impact the CO2 production.

Parameter	Final	Initial	delta	Percentage reduction
Valves / travel	4.95238	3	1.95238	
Travels / valves	0.201923	0.333333	-0.13141	-39.42%
CO2 emissions reduction (Kg/year)	61,56	101,632	- 40,06	-39.42%

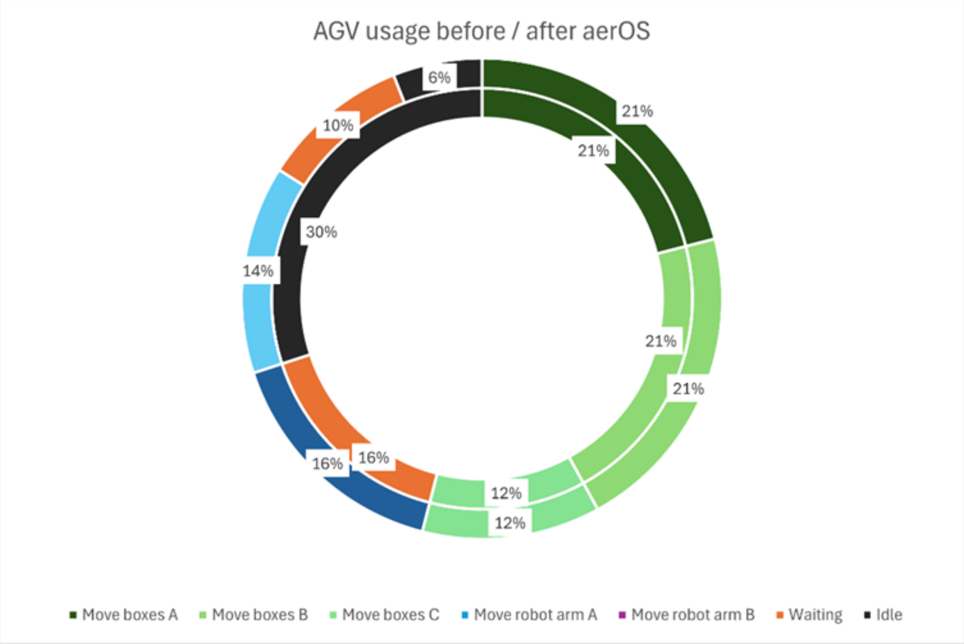
All the underlying data is available in both the Orion-LD as well as the MADE LEA System and it is possible to verify the same on request.

To calculate the total amount of CO₂ saved, the energy consumption and emission factors for both baseline and optimized conditions were compared. The calculation of the total CO₂ saved is based on the energy consumption and emission factor shown in the table. Starting from an average energy use of 0.4 KWh/g over 30 production cycles, with an energy consumption of 0.0133 KWh per cycle, and 4 cycles per hour for 16 hours per day, the total daily energy consumption is 0.8533 KWh/g. Scaled to an annual production of 300 grams, this results in 256 KWh per year. Using an emission factor of 397 g CO₂/kWh, the total annual emissions correspond to 101,632 g CO₂, or approximately 101.6 kg CO₂ per year. The optimized condition achieved a 39–40% reduction, which represents the total CO₂ saved compared to the baseline.

KPI 2.1.7 AGV usage

Table 88: KPI 2.1.7 AGV usage

KPI ID number and partner resp.	**KPI 2.1.7**
KPI Name	**AGV usage**
Description	AGV use above 80%
Motivation	The AGV usage should be optimized to exploit as much as possible its work in manufacturing areas. The AGV usage optimization will impact also AGV availability
Target value	>80%

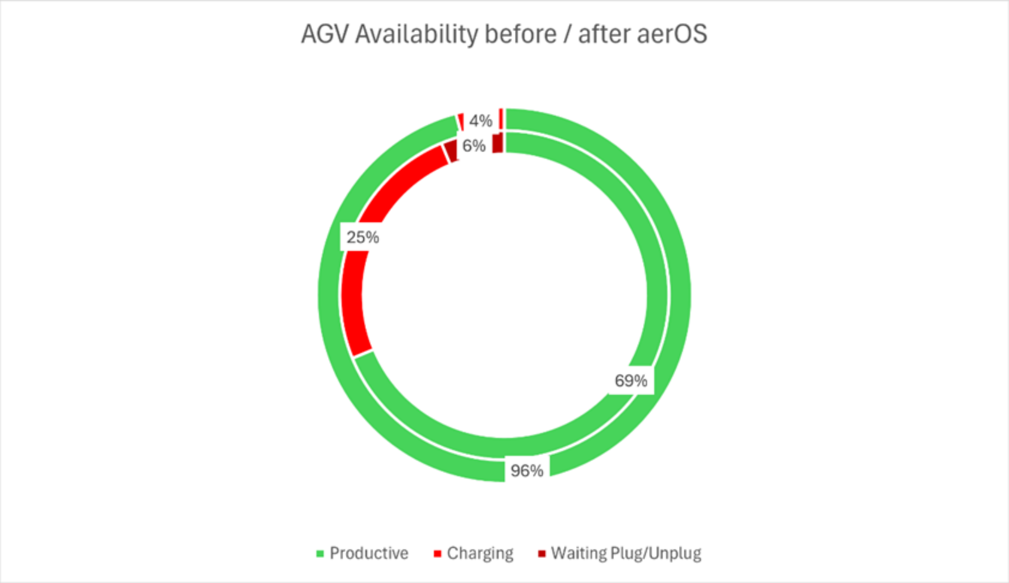
<table>
<tr><td>Prerequisites</td><td colspan="3">The AGVs must be fully operational and integrated into the manufacturing workflow. An efficient orchestrator and load balancing system must be established to distribute tasks evenly among all AGVs. Additionally, staff should be trained to manage AGV operations and use the monitoring systems effectively to ensure balanced AGV usage.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">HLO (T3.3), Data Fabric (T4.2), Low Code Tools (T3.2)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">The evaluation of this KPI will involve continuous monitoring of AGV usage data to ensure balanced workload distribution. This includes collecting data on the operational hours and tasks completed by each AGV within a given time frame. The data will be analysed to calculate the percentage of time each AGV is in use compared to its total available time. The goal is to ensure that all AGVs are utilized evenly, preventing any single AGV from being overused. Comparisons will be made against the target value of 80% to ensure optimal usage. Any imbalances in AGV usage will be investigated, and adjustments to the orchestrator system will be made to address them.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>54 %</i></td><td><i>50%</i></td><td><i>80% (accomplished)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">• Baseline utilisation was 54 %—AGVs were often idle while waiting for box-moving jobs.
• A new aerOS-delivered skill lets mobile robotic-arm workstations broadcast relocation requests, giving AGVs an additional task class.
• Usage monitoring during the latest production month shows that AGVs were active for 84 % of their available time, thus satisfying the "> 80 %" target and representing a 30 percentage-point increase over the pre-skill situation.
 </td></tr>
</table>

	Justification of how aerOS has allowed this KPI to be met is included in Section 4 of this document.

KPI 2.1.8 AGV availability

Table 89: KPI 2.1.8 AGV availability

KPI ID number and partner resp.	**KPI 2.1.8**		
KPI Name	**AGV availability**		
Description	AGV availability above 95%		
Motivation	The AGV availability should be increased to make the manufacturing process leaner and more responsive.		
Target value	>95%		
Prerequisites	There should be a enough number of AGVs in the fleet that have no issues and connected to the aerOS continuum. Fully integrated system		
aerOS components (task)	KrakenD (T3.1), Ingress (T3.1), Ingress&KrakenD conf (T3.1), CertManager&LetsEncrypt (TLS) (T3.1), FDQN (T3.1), NAT Capable (T3.1), OpenAPI (T3.2), AsyncAPI (T3.2), Low-Code (T3.2), HLO (T.3.3), LLO (k8s, Docker) (T3.3), KeyCloack (T3.4), KrakenD (T3.4), OpenLDAP (T3.4), Self-awareness (T3.5), Self-orchestration (T3.5), Self-diagnose (T3.5), Self-security (T3.5), Self-healing (T3.5), Self-Configuration (T3.5), Self-API (T3.5), Data-Interoperability (T4.1), LDAP (T4.2), EAT (T4.4), Trust SCO. (T4.5), IOTA (T4.5), Portal (T4.6).		
Evaluation means	The evaluation of this KPI will involve continuous monitoring of AGV operational status. This includes collecting data on the total available time and the actual operational time of each AGV. The data will be analysed to calculate the availability percentage, ensuring it meets or exceeds the target value of 95%.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***69 %***	***N/A***	***96 %***
Outcome elaboration (M38)	• Before the aerOS update, every AGV needed ≈ 2 h of wired charging plus ≈ 0.5 h of waiting/handling per 8-h shift. • Productive time/shift = 8 h – 2.5 h = 5.5 h → 69 % availability. • After deploying the autonomous docking skill as a Docker container via aerOS, each AGV now charges wirelessly on demand. • Average charging time/shift fell to ≈ 0.33 h (20 min). • No manual intervention time is required. Productive time/shift = 8 h – 0.33 h = 7.67 h → 96 % availability. • The measured fleet average over the last four weeks of operation is 96 %, thus meeting and slightly exceeding the > 95 % target.		

KPI 2.1.9 AGV travel saved/valve

Table 90: KPI 2.1.9 AGV travel saved/valve

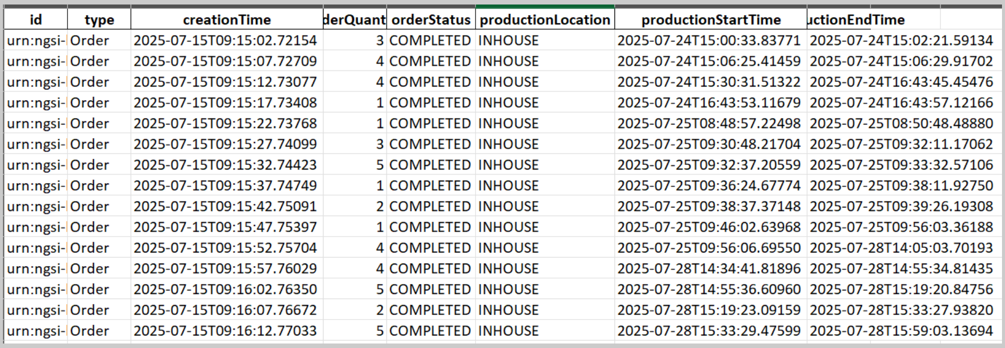
KPI ID number and partner resp.	**KPI 2.1.9**
KPI Name	**AGV travel saved/valve**
Description	aerOS system should permit with AI/ML component contribution to optimize travels of AGV and improve the ratio travel/valve.
Motivation	The AGV travel/valve improved makes the manufacturing processes more lean, responsive and energy demanding.
Target value	<20%
Prerequisites	Fully integrated system
aerOS components (task)	KrakenD (T3.1), Ingress (T3.1), Ingress&KrakenD conf (T3.1), CertManager&LetsEncrypt (TLS) (T3.1), FDQN (T3.1), NAT Capable (T3.1), OpenAPI (T3.2), AsyncAPI (T3.2), Low-Code (T3.2), HLO (T.3.3), LLO (k8s, Docker) (T3.3), KeyCloack (T3.4), KrakenD (T3.4), OpenLDAP (T3.4), Self-awareness (T3.5), Self-orchestration (T3.5), Self-diagnose (T3.5), Self-security (T3.5), Self-healing (T3.5), Self-Configuration (T3.5), Self-API (T3.5), Data-Interoperability (T4.1), LDAP (T4.2), EAT (T4.4), Trust SCO. (T4.5), IOTA (T4.5), Portal (T4.6).
Evaluation means	The methodology for assessing the KPI will be based on measuring the following parameter on a fixed number of cycles: (Travel Factor) TF = Number of AGV travels / Number of valves carried

Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0% (1 Travel per Valve)***	***N/A***	***39.42%***
Outcome elaboration (M38)	In order to calculate this KPI, POLIMI and MADE first ran the production line in baseline state to collect data for baseline generation. In order to maintain the data storage modalities, the team also persisted this data to the Orion-LD but in this case the optimization was bypassed by creating a dedicated docker image where the orders coming from the order generator are directly passed to the MADE LEA system without any changes or clubbing of orders. In the second phase the optimization was turned on. Again, in this case also all the production data like order receive times, production start and end times etc. was captured inside Orion-LD. Finally, we used a Python script to query all the data and save this in an excel format.		

```
import requests
import pandas as pd
base_url = "http://localhost:1026/ngsi-ld/v1/entities"
query_url = f"{base_url}/?type=Order&limit=1000&q=productionStartTime!=\"timestamp\""
headers = {
    'Link': '<http://context:5051/ngsi-context.jsonld>; rel="http://www.w3.org/ns/json-ld#cont
}

response = requests.get(query_url, headers=headers)
if response.status_code == 200:
    entities = response.json()
    print(f"Fetched {len(entities)} entities")
    data = []
    for e in entities:
        row = {
            "id": e.get("id"),
            "type": e.get("type"),
            "creationTime": e.get("creationTime", {}).get("value", ""),
            "orderQuantity": e.get("orderQuantity", {}).get("value", ""),
            "orderStatus": e.get("orderStatus", {}).get("value", ""),
            "productionLocation": e.get("productionLocation", {}).get("value", ""),
            "productionStartTime": e.get("productionStartTime", {}).get("value", ""),
            "productionEndTime": e.get("productionEndTime", {}).get("value", ""),
            "orderOutsourceTime": e.get("orderOutsourceTime", {}).get("value", ""),
            "productReceiveTime": e.get("productReceiveTime", {}).get("value", ""),
            "totalProductionTime": e.get("totalProductionTime", {}).get("value", "")
        }
        data.append(row)
    df = pd.DataFrame(data)
    df.to_excel("orders.xlsx", index=False)
    print("Saved entities to orders.xlsx")
else:
    print(f"Failed to fetch entities: {response.status_code} {response.text}")
```

Python script query all the data and save in an excel format.

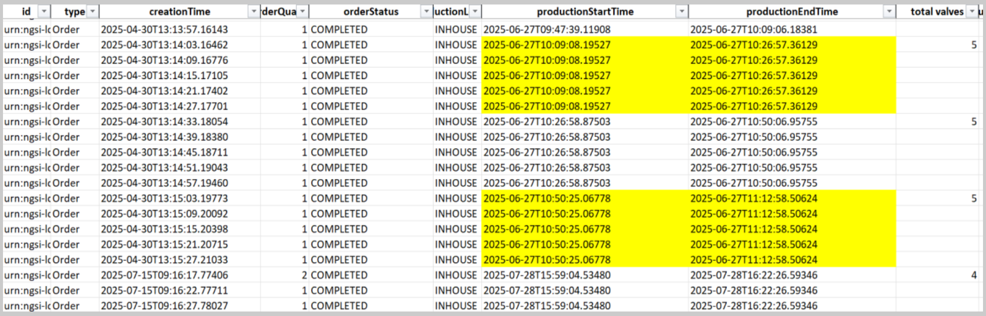
For baseline case, it is possible to directly use the order quantity to calculate the average number of valves per travel from the order quantity.



id	type	creationTime	derQuant	orderStatus	productionLocation	productionStartTime	ıctionEndTime
urn:ngsi-	Order	2025-07-15T09:15:02.72154	3	COMPLETED	INHOUSE	2025-07-24T15:00:33.83771	2025-07-24T15:02:21.59134
urn:ngsi-	Order	2025-07-15T09:15:07.72709	4	COMPLETED	INHOUSE	2025-07-24T15:06:25.41459	2025-07-24T15:06:29.91702
urn:ngsi-	Order	2025-07-15T09:15:12.73077	4	COMPLETED	INHOUSE	2025-07-24T15:30:31.51322	2025-07-24T16:43:45.45476
urn:ngsi-	Order	2025-07-15T09:15:17.73408	1	COMPLETED	INHOUSE	2025-07-24T16:43:53.11679	2025-07-24T16:43:57.12166
urn:ngsi-	Order	2025-07-15T09:15:22.73768	1	COMPLETED	INHOUSE	2025-07-25T08:48:57.22498	2025-07-25T08:50:48.48880
urn:ngsi-	Order	2025-07-15T09:15:27.74099	3	COMPLETED	INHOUSE	2025-07-25T09:30:48.21704	2025-07-25T09:32:11.17062
urn:ngsi-	Order	2025-07-15T09:15:32.74423	5	COMPLETED	INHOUSE	2025-07-25T09:32:37.20559	2025-07-25T09:33:32.57106
urn:ngsi-	Order	2025-07-15T09:15:37.74749	1	COMPLETED	INHOUSE	2025-07-25T09:36:24.67774	2025-07-25T09:38:11.92750
urn:ngsi-	Order	2025-07-15T09:15:42.75091	2	COMPLETED	INHOUSE	2025-07-25T09:38:37.37148	2025-07-25T09:39:26.19308
urn:ngsi-	Order	2025-07-15T09:15:47.75397	1	COMPLETED	INHOUSE	2025-07-25T09:46:02.63968	2025-07-25T09:56:03.36188
urn:ngsi-	Order	2025-07-15T09:15:52.75704	4	COMPLETED	INHOUSE	2025-07-25T09:56:06.69550	2025-07-28T14:05:03.70193
urn:ngsi-	Order	2025-07-15T09:15:57.76029	4	COMPLETED	INHOUSE	2025-07-28T14:34:41.81896	2025-07-28T14:55:34.81435
urn:ngsi-	Order	2025-07-15T09:16:02.76350	5	COMPLETED	INHOUSE	2025-07-28T14:55:36.60960	2025-07-28T15:19:20.84756
urn:ngsi-	Order	2025-07-15T09:16:07.76672	2	COMPLETED	INHOUSE	2025-07-28T15:19:23.09159	2025-07-28T15:33:27.93820
urn:ngsi-	Order	2025-07-15T09:16:12.77033	5	COMPLETED	INHOUSE	2025-07-28T15:33:29.47599	2025-07-28T15:59:03.13694

Number of valves per travel from the order quantity

For the Improved case, it was needed to first club the orders in which production start time and end time are identical (meaning these were clubbed together) to create a new column showing the total number of valves carried in a trip. Figure below shows a snapshot of this process



id	type	creationTime	derQua	orderStatus	uctionL	productionStartTime	productionEndTime	total valves
urn:ngsi-lc	Order	2025-04-30T13:13:57.16143	1	COMPLETED	INHOUSE	2025-06-27T09:47:39.11908	2025-06-27T10:09:06.18381	
urn:ngsi-lc	Order	2025-04-30T13:14:03.16462	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129	5
urn:ngsi-lc	Order	2025-04-30T13:14:09.16776	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129	
urn:ngsi-lc	Order	2025-04-30T13:14:15.17105	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129	
urn:ngsi-lc	Order	2025-04-30T13:14:21.17402	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129	
urn:ngsi-lc	Order	2025-04-30T13:14:27.17701	1	COMPLETED	INHOUSE	2025-06-27T10:09:08.19527	2025-06-27T10:26:57.36129	
urn:ngsi-lc	Order	2025-04-30T13:14:33.18054	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755	5
urn:ngsi-lc	Order	2025-04-30T13:14:39.18380	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755	
urn:ngsi-lc	Order	2025-04-30T13:14:45.18711	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755	
urn:ngsi-lc	Order	2025-04-30T13:14:51.19043	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755	
urn:ngsi-lc	Order	2025-04-30T13:14:57.19460	1	COMPLETED	INHOUSE	2025-06-27T10:26:58.87503	2025-06-27T10:50:06.95755	
urn:ngsi-lc	Order	2025-04-30T13:15:03.19773	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624	5
urn:ngsi-lc	Order	2025-04-30T13:15:09.20092	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624	
urn:ngsi-lc	Order	2025-04-30T13:15:15.20398	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624	
urn:ngsi-lc	Order	2025-04-30T13:15:21.20715	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624	
urn:ngsi-lc	Order	2025-04-30T13:15:27.21033	1	COMPLETED	INHOUSE	2025-06-27T10:50:25.06778	2025-06-27T11:12:58.50624	
urn:ngsi-lc	Order	2025-07-15T09:16:17.77406	2	COMPLETED	INHOUSE	2025-07-28T15:59:04.53480	2025-07-28T16:22:26.59346	4
urn:ngsi-lc	Order	2025-07-15T09:16:22.77711	1	COMPLETED	INHOUSE	2025-07-28T15:59:04.53480	2025-07-28T16:22:26.59346	
urn:ngsi-lc	Order	2025-07-15T09:16:27.78027	1	COMPLETED	INHOUSE	2025-07-28T15:59:04.53480	2025-07-28T16:22:26.59346	

Finally, from both these values, the average values are taken for both the conditions and then calculate the reduction in terms of travels per valve. Which is reported in the following table. This same reduction directly impacts the CO2 production.

Reduction directly impact the CO2 production.

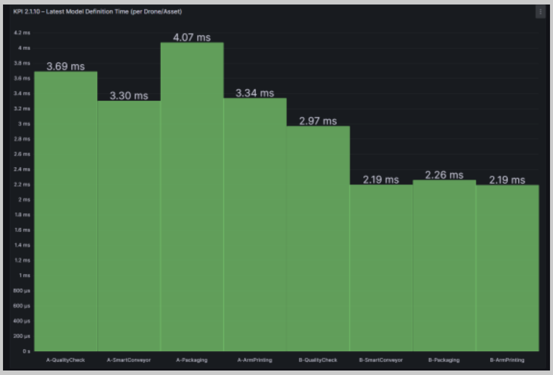
Parameter	**Final**	**Initial**	**delta**	**Percentage reduction**
Valves / travel	4.95238	3	1.95238	
Travels / valves	0.201923	0.333333	-0.13141	-39.42%
CO2 emissions reduction (Kg/year)	61,56	101,632	- 40,06	-39.42%

All the underlying data is available in both the Orion-LD as well as the MADE LEA System and it is possible to verify the same on request.

KPI 2.1.10 Definition of the calculation model

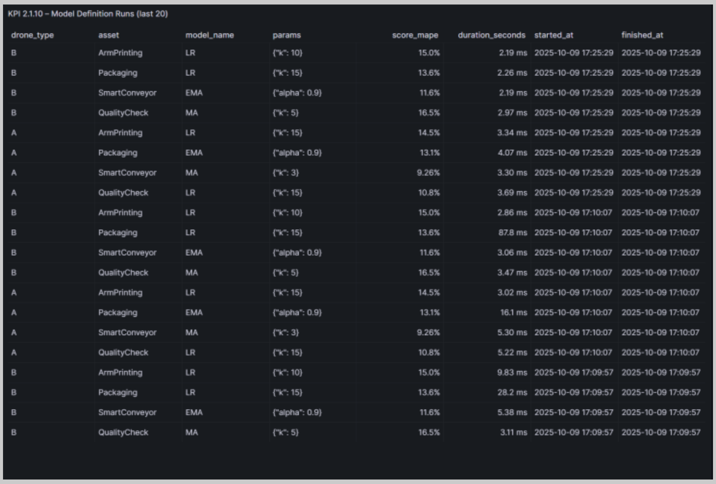
Table 91: KPI 2.1.10 Definition of the calculation model

KPI ID number and partner resp.	**KPI 2.1.10**		
KPI Name	**Definition of the calculation model**		
Description	Time required to define the calculation model for a specific product		
Motivation	Faster definition of the calculation model for each specific product improves the overall efficiency of the production and demonstrates the effectiveness of aerOS regarding real-time data processing.		
Target value	> 30%-time reduction		
Prerequisites	To calculate the time required to define the calculation model for a specific product, the IEs configuration must be complete, Node-Red configured, the data collection tool set up and the network and APIs configured.		
aerOS components (task)	Networking (T3.1), API & Low Code tools (T3.2), Orchestration (T3.3), T3.5 (Self-*), T4.2 (Data Fabric)		
Evaluation means	To measure the time required to define the CO2 footprint calculation model for a specific product, start and end times of the model development process are recorded. This time is continuously tracked for each product using tracking tools. The total and average time taken are then calculated. Dashboards and reports are used to visualize and monitor these times, identifying areas for process improvement.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***120 minutes***	***90 minutes*** ***(25% reduction)***	***< 1 second*** ***(>99,99% reduction)***
Outcome elaboration (M38)	The co2-total-barplot-prediction function defines a simple time-series model (Moving Average, EMA, Linear Regression) for each factory asset to predict the CO_2 footprint for the next drone order. It selects the best-performing model and parameters based on historical CO_2 totals; then it uses that model to write new predictions. Finally, it displays recent actuals and the newest prediction per drone type in Grafana, along with accuracy metrics and KPI 2.1.10 model-definition timings. For every product/modeling run, we record a start timestamp when model definition begins (including data prep and selection) and an end timestamp immediately after the configuration persisted. The difference yields duration seconds for that run. These start/end times and durations are stored per (drone type, asset), and the process runs continuously so each new or updated product gets a fresh measurement. Dashboards aggregate these records to show: 1. a bar view of the latest model-definition time per drone/asset,		



2. a table of the most recent runs (model name, parameters, MAPE score, duration, started/finished times), plus roll-ups (totals/averages) over selectable periods.

This makes the time to define or update the CO₂ calculation model **visible and auditable**, highlights outliers and bottlenecks, and supports targeted process improvements.

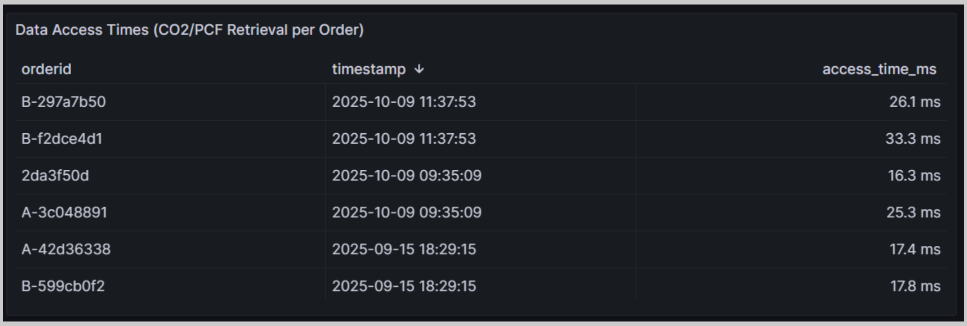

KPI 2.1.10 – Model Definition Runs (last 20)

drone_type	asset	model_name	params	score_mape	duration_seconds	started_at	finished_at
B	ArmPrinting	LR	{"k": 10}	15.0%	2.19 ms	2025-10-09 17:25:29	2025-10-09 17:25:29
B	Packaging	LR	{"k": 15}	13.6%	2.26 ms	2025-10-09 17:25:29	2025-10-09 17:25:29
B	SmartConveyor	EMA	{"alpha": 0.9}	11.6%	2.19 ms	2025-10-09 17:25:29	2025-10-09 17:25:29
B	QualityCheck	MA	{"k": 5}	16.5%	2.97 ms	2025-10-09 17:25:29	2025-10-09 17:25:29
A	ArmPrinting	LR	{"k": 15}	14.5%	3.34 ms	2025-10-09 17:25:29	2025-10-09 17:25:29
A	Packaging	EMA	{"alpha": 0.9}	13.1%	4.07 ms	2025-10-09 17:25:29	2025-10-09 17:25:29
A	SmartConveyor	MA	{"k": 3}	9.26%	3.30 ms	2025-10-09 17:25:29	2025-10-09 17:25:29
A	QualityCheck	LR	{"k": 15}	10.8%	3.69 ms	2025-10-09 17:25:29	2025-10-09 17:25:29
B	ArmPrinting	LR	{"k": 10}	15.0%	2.86 ms	2025-10-09 17:10:07	2025-10-09 17:10:07
B	Packaging	LR	{"k": 15}	13.6%	87.8 ms	2025-10-09 17:10:07	2025-10-09 17:10:07
B	SmartConveyor	EMA	{"alpha": 0.9}	11.6%	3.06 ms	2025-10-09 17:10:07	2025-10-09 17:10:07
B	QualityCheck	MA	{"k": 5}	16.5%	3.47 ms	2025-10-09 17:10:07	2025-10-09 17:10:07
A	ArmPrinting	LR	{"k": 15}	14.5%	3.02 ms	2025-10-09 17:10:07	2025-10-09 17:10:07
A	Packaging	EMA	{"alpha": 0.9}	13.1%	16.1 ms	2025-10-09 17:10:07	2025-10-09 17:10:07
A	SmartConveyor	MA	{"k": 3}	9.26%	5.30 ms	2025-10-09 17:10:07	2025-10-09 17:10:07
A	QualityCheck	LR	{"k": 15}	10.8%	5.22 ms	2025-10-09 17:10:07	2025-10-09 17:10:07
B	ArmPrinting	LR	{"k": 10}	15.0%	9.83 ms	2025-10-09 17:09:57	2025-10-09 17:09:57
B	Packaging	LR	{"k": 15}	13.6%	28.2 ms	2025-10-09 17:09:57	2025-10-09 17:09:57
B	SmartConveyor	EMA	{"alpha": 0.9}	11.6%	5.38 ms	2025-10-09 17:09:57	2025-10-09 17:09:57
B	QualityCheck	MA	{"k": 5}	16.5%	3.11 ms	2025-10-09 17:09:57	2025-10-09 17:09:57

KPI 2.1.11 Transparency of CO2/PCF data (minutes)

Table 92: KPI 2.1.11 Transparency of CO2/PCF data (minutes)

KPI ID number and partner resp.	**KPI 2.1.11 (SIPBB)**
KPI Name	**Transparency of CO2/PCF data (minutes)**
Description	Time required to access CO2/PCF data for a specific product.
Motivation	Faster access to CO2/PCF data for each specific product allows greater transparency for customers and real-time control of the factory.

<table>
<tr><td>Target value</td><td colspan="3">< 2 minutes</td></tr>
<tr><td>Prerequisites</td><td colspan="3">To calculate the time required to access CO2/PCF data for a specific product, the IEs configuration must be complete, Node-Red configured, the data collection tool set up and the network and APIs configured, as well as the aerOS runtime and the web service to access the data.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">Networking (T3.1), API & Low Code tools (T3.2), Orchestration (T3.3), Cybersecurity tools (T3.4), T3.5 (Self-*), T4.2 (Data Fabric), aerOS Portal (T4.6)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">To measure the time required to access CO2/PCF data for a specific product, request and access times for each data retrieval are recorded. These times are continuously tracked using data management systems and time tracking tools. The total and average time taken to access the data are then calculated. Dashboards and reports are used to visualize and monitor these times, identifying areas for process improvement and reduce access times.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td>N/A</td><td>N/A</td><td>~0.025 seconds</td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">
<p>We deliver Digital Product Passports (DPPs) for every drone of Type A and B that surface (i) the total CO2 footprint and (ii) a per-machine/process contribution breakdown for the production flow. The dpp-generator function assembles these passports automatically from the production data lake (including the totals produced by calculate-total-co2), standardizes the schema (order ID, timestamps, asset/machine IDs, totals, and per-asset CO2), and persists the result in a queryable store exposed to dashboards and APIs.</p>
<p>We measure the access time by taking the difference of the timestamp when the request is made and when the first byte of data is returned. Our dashboard continuously displays this to show per-request latency.</p>

<p>Data Access Times (CO2/PCF Retrieval per Order)</p>
<table>
<tr><th>orderid</th><th>timestamp ↓</th><th>access_time_ms</th></tr>
<tr><td>B-297a7b50</td><td>2025-10-09 11:37:53</td><td>26.1 ms</td></tr>
<tr><td>B-f2dce4d1</td><td>2025-10-09 11:37:53</td><td>33.3 ms</td></tr>
<tr><td>2da3f50d</td><td>2025-10-09 09:35:09</td><td>16.3 ms</td></tr>
<tr><td>A-3c048891</td><td>2025-10-09 09:35:09</td><td>25.3 ms</td></tr>
<tr><td>A-42d36338</td><td>2025-09-15 18:29:15</td><td>17.4 ms</td></tr>
<tr><td>B-599cb0f2</td><td>2025-09-15 18:29:15</td><td>17.8 ms</td></tr>
</table>
<p>The average access time is 25 ms for each DPP (covering order ID and total CO2). This number is shown prominently on the dashboard, and a table lists the access time for every individual DPP so it can be audited.</p>
<p><u>Result:</u> Because passports are generated automatically and can be fetched in ~0.025 s, CO2/PCF data becomes available within minutes after an order finishes and is then retrievable in milliseconds. This meets the KPI target for fast, transparent access.</p>
</td></tr>
</table>

Pilot 2 Containerised edge computing near renewable energy sources

KPI 2.2.1 Consumed renewable energy based on decision making process of aerOS

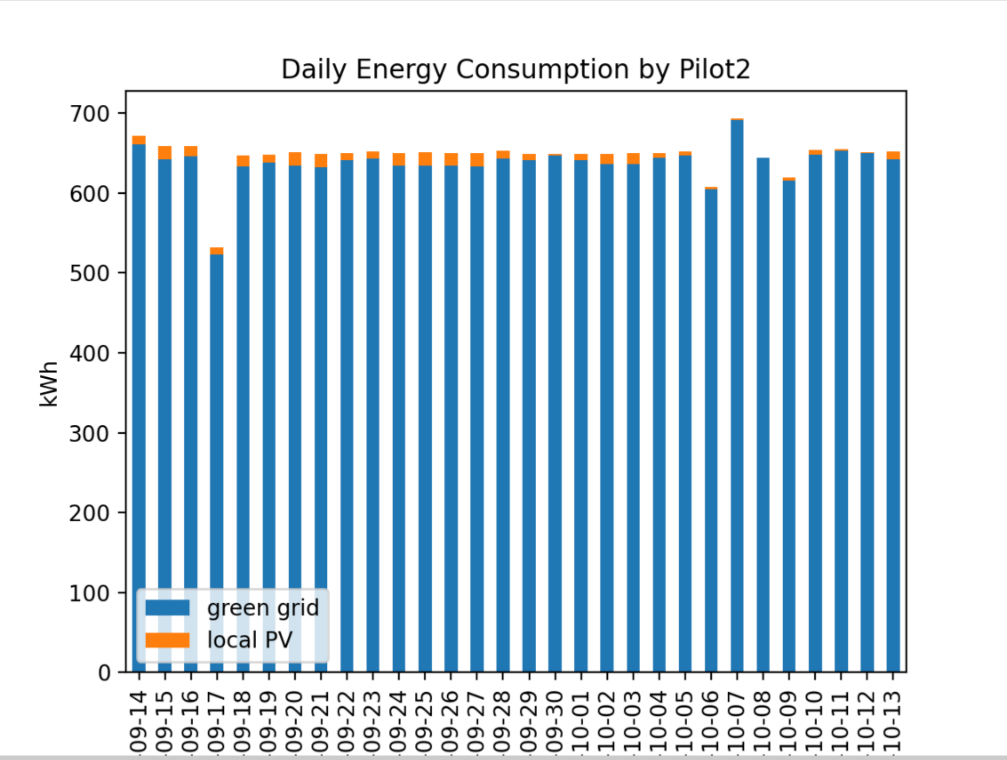
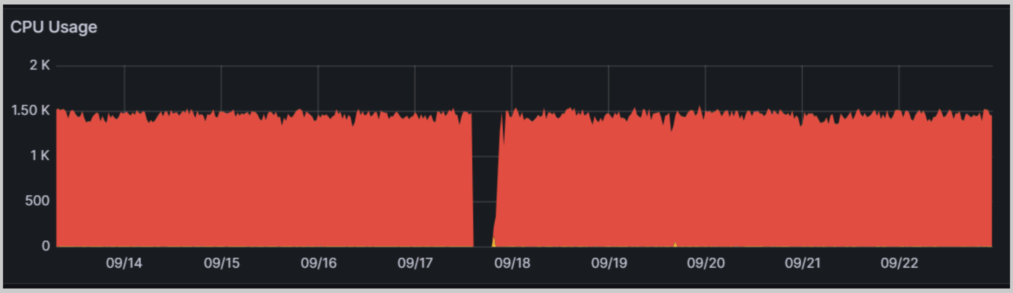
Table 93: KPI 2.2.1 Consumed renewable energy based on decision making process of aerOS

<table>
<tr><th>KPI ID number and partner resp.</th><th colspan="3">KPI 2.2.1</th></tr>
<tr><td>KPI Name</td><td colspan="3">Consumed renewable energy based on decision making process of aerOS</td></tr>
<tr><td>Description</td><td colspan="3">The total amount of renewable energy consumed on monthly basis.</td></tr>
<tr><td>Motivation</td><td colspan="3">KPI shows that the absolute energy usage is big enough to consider the pilot as representable.</td></tr>
<tr><td>Target value</td><td colspan="3">20 MWh/month</td></tr>
<tr><td>Prerequisites</td><td colspan="3">Containers need to be connected to power source.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">All</td></tr>
<tr><td>Evaluation means</td><td colspan="3">Monitoring of power consumption energy meters collected via SCADA interface.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>0 MWh/month</i></td><td><i>N/A</i></td><td><i>19391.54 MWh (97%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">The energy consumption was measured during Scenario 1 calculation and very high average CPU consumption (84%) of all compute nodes.

Daily Energy consumption in the reported period:
<table>
<tr><th>Date</th><th>kWh green grid</th><th>kWh roof PV</th></tr>
<tr><td>9/14/2025</td><td>661.015524</td><td>10.186375</td></tr>
<tr><td>9/15/2025</td><td>642.208029</td><td>16.735757</td></tr>
<tr><td>9/16/2025</td><td>646.301701</td><td>12.487</td></tr>
<tr><td>9/17/2025</td><td>523.155216</td><td>8.767</td></tr>
<tr><td>9/18/2025</td><td>633.494742</td><td>13.767</td></tr>
<tr><td>9/19/2025</td><td>638.302353</td><td>9.231</td></tr>
<tr><td>9/20/2025</td><td>634.096766</td><td>16.881</td></tr>
</table>
</td></tr>
</table>

9/21/2025	632.241877	16.249
9/22/2025	641.408644	8.023
9/23/2025	642.684826	9.513
9/24/2025	634.163122	16.031
9/25/2025	634.484228	16.351
9/26/2025	633.938873	15.707
9/27/2025	633.483184	15.983
9/28/2025	642.587665	9.992
9/29/2025	641.349555	7.71
9/30/2025	646.871244	1.761
10/1/2025	641.416134	7.752
10/2/2025	636.100148	12.616
10/3/2025	636.316501	13.076
10/4/2025	643.516107	6.662
10/5/2025	646.473764	5.28
10/6/2025	604.908593	2.777
10/7/2025	690.872914	1.831
10/8/2025	644.139447	NaN
10/9/2025	615.040054	3.77
10/10/2025	647.936411	6.204
10/11/2025	652.567026	2.407
10/12/2025	649.407577	1.845
10/13/2025	642.26693	9.192
Sum: 19391.54	**19112.75**	**278.79**

On 2025-09-17 you can see the lower value of the energy consumption, and it is related to short break between processing batches. The lower graph shows CPU consumption in number of cores (all compute nodes have 1776 CPU cores). It demonstrates the described correlation between the CPU usage and energy consumption.

KPI 2.2.2 Effectiveness of task distribution through aerOS to nodes

Table 94: KPI 2.2.2 Effectiveness of task distribution through aerOS to nodes

KPI ID number and partner resp.	**KPI 2.2.2**
KPI Name	**Effectiveness of task distribution through aerOS to nodes**
Description	KPI shows the share of scheduled task completed on time.
Motivation	The proper on-schedule job handling is crucial for the overall trust in the compute solution.
Target value	99.5% of tasks executed on schedule
Prerequisites	Container connected to power, connected to the network. aerOS continuum installed and ready to use.

aerOS components (task)	HLO (T3.3), LLO.		
Evaluation means	Due date of each workload will be compared with actual end date of processing.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***N/A***	***100%***
Outcome elaboration (M38)	To validate this KPI we were processing historical data of Sentinel-2 Earth Observation. Our tasks are executed as batch jobs, continuously, without tracking the completion status of individual tasks. As a result, we assume that all tasks are completed on time.		

KPI 2.2.3 Scalability of task distribution and management through aerOS

Table 95: KPI 2.2.3 Scalability of task distribution and management through aerOS

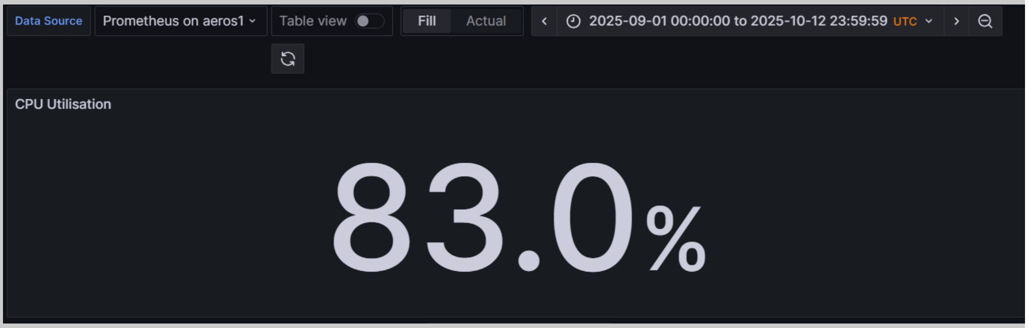
KPI ID number and partner resp.	**KPI 2.2.3**		
KPI Name	**Scalability of task distribution and management through aerOS**		
Description	The amount of task scheduled by aerOS in Pilot 2 compute edges.		
Motivation	KPI shows the flexibility and scalability of aerOS. The task might contain more than one job. Tasks might be batch or interactive type.		
Target value	10k tasks executed/month		
Prerequisites	Container connected to power, connected to the network. aerOS continuum installed and ready to use.		
aerOS components (task)	HLO (T3.3), LLO (T3.3), IdM (T3.4)		
Evaluation means	Task done will be counted based on logs and visualized with Grafana.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***N/A***	***up to 250k jobs/month***
Outcome elaboration (M38)	We can execute up to 250k jobs of Scenario 1 per month.		

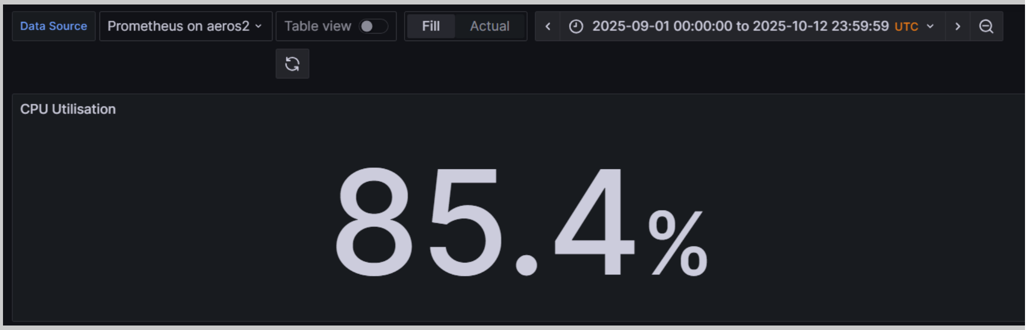
```
eouser@jumper:~$ date
Wed Oct 15 15:45:10 UTC 2025
eouser@jumper:~$ s5cmd --endpoint-url=https://s3.waw4-1.cloudferro.com ls s3://aeros-cloudmask-public/* | cut -d '/' -f
1,2 | sort | uniq -c
  74114 2025/08
 251953 2025/09
 164244 2025/10
eouser@jumper:~$ |
```

All outcomes used to calculate this KPI can be found here: https://s3.waw4-1.cloudferro.com/swift/v1/aeros-cloudmask-public.

KPI 2.2.4 CPU utilization efficiency

Table 96: KPI 2.2.4 4 CPU utilization efficiency

KPI ID number and partner resp.	**KPI 2.2.4**		
KPI Name	**CPU utilization efficiency**		
Description	KPI shows the average CPU consumption by worker nodes (excluding master nodes control and network devices).		
Motivation	For energy saving it's very important to have a proper autoscaling solution. KPI shows the capability of disabling unused nodes to save energy.		
Target value	80%		
Prerequisites	Container connected to power, connected to the network. aerOS continuum installed and ready to use.		
aerOS components (task)	HLO (T3.3), LLO (T3.3)		
Evaluation means	Metrics from nodes will be reported in Prometheus and displayed on Grafana.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0%***	***N/A***	***Average 84%***
Outcome elaboration (M38)	During Scenario 1 cloud masks calculation we got over 80% CPU utilization (as a proof we attach the average CPU utilization from the period between 2025-09-01 and 2025-10-12 for each aerOS edge node.  		

Data Source Prometheus on aeros2

2025-09-01 00:00:00 to 2025-10-12 23:59:59 UTC

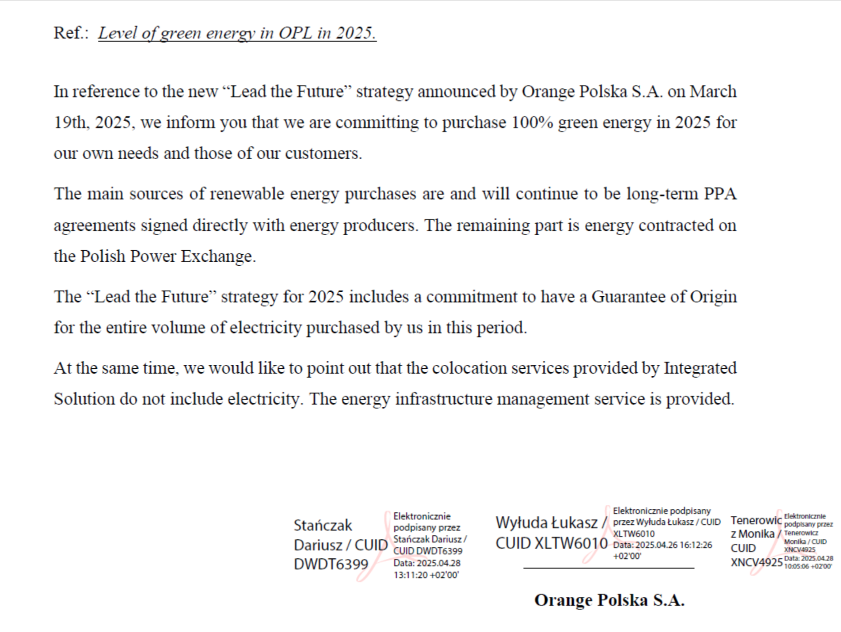
CPU Utilisation

85.4%

KPI 2.2.5 Carbon awareness share of green energy

Table 97: KPI 2.2.5 Carbon awareness share of green energy

KPI ID number and partner resp.	**KPI 2.2.5**		
KPI Name	**Carbon awareness share of green energy**		
Description	KPI shows the green energy share for jobs with green energy preference label.		
Motivation	System shall support the choice of the green energy when scheduling job. Some urgent jobs might be launched regardless of energy source, and some can strongly prefer/require green energy.		
Target value	60%		
Prerequisites	Containers are connected to green energy. Energy meters and Data Logger connected to appropriate places.		
aerOS components (task)	HLO (T3.3), LLO (T3.3)		
Evaluation means	Monitoring consumption of green energy at energy meter.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0%***	***N/A***	***100%***
Outcome elaboration (M38)	Orange Polska S.A. the energy provider on the site where container with edge nodes is located confirms that they use 100% green energy.		



Ref.: *Level of green energy in OPL in 2025.*

In reference to the new "Lead the Future" strategy announced by Orange Polska S.A. on March 19th, 2025, we inform you that we are committing to purchase 100% green energy in 2025 for our own needs and those of our customers.

The main sources of renewable energy purchases are and will continue to be long-term PPA agreements signed directly with energy producers. The remaining part is energy contracted on the Polish Power Exchange.

The "Lead the Future" strategy for 2025 includes a commitment to have a Guarantee of Origin for the entire volume of electricity purchased by us in this period.

At the same time, we would like to point out that the colocation services provided by Integrated Solution do not include electricity. The energy infrastructure management service is provided.

Stańczak Dariusz / CUID DWDT6399 Elektronicznie podpisany przez Stańczak Dariusz / CUID DWDT6399 Data: 2025.04.28 13:11:20 +02'00'

Wyłuda Łukasz / CUID XLTW6010 Elektronicznie podpisany przez Wyłuda Łukasz / CUID XLTW6010 Data: 2025.04.26 16:12:26 +02'00'

Tenerowicz Monika / CUID XNCV4925 Elektronicznie podpisany przez Tenerowicz Monika / CUID XNCV4925 Data: 2025.04.28 10:05:06 +02'00'

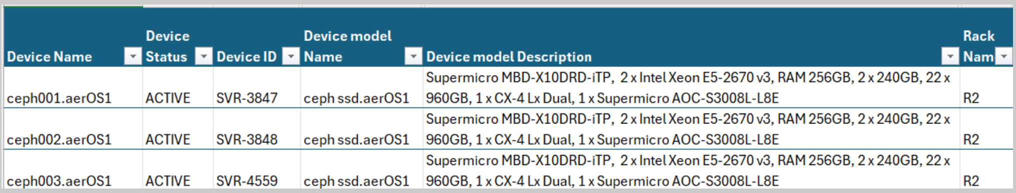
Orange Polska S.A.

KPI 2.2.6 Number of edge nodes connected in the aerOS continuum

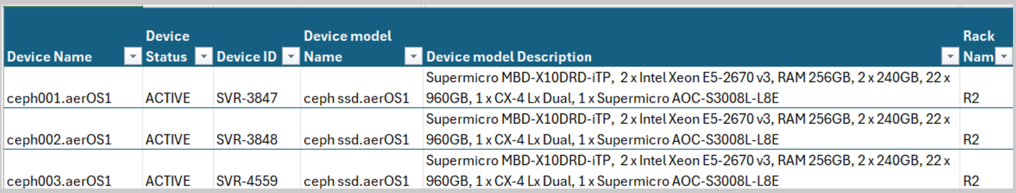
Table 98: KPI 2.2.6 Number of edge nodes connected in the aerOS continuum

KPI ID number and partner resp.	**KPI 2.2.6**		
KPI Name	**Number of edge nodes connected in the aerOS continuum**		
Description	The total count of pilot's edge nodes (physical locations).		
Motivation	KPI shows more than one edge node with the different energy supply, it gives the opportunity to show advantages of the aerOS job distribution subsystem.		
Target value	2		
Prerequisites	Container is ready to Host RACK and HW. HW is available. HW is installed in container and properly configured.		
aerOS components (task)	All		
Evaluation means	Count Number of containers serving as Edge Node.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0***	***N/A***	***2***
Outcome elaboration (M38)	Number of edge nodes: 2. The tables below showcase the device list in each edge node (aerOS1, aerOS2).		

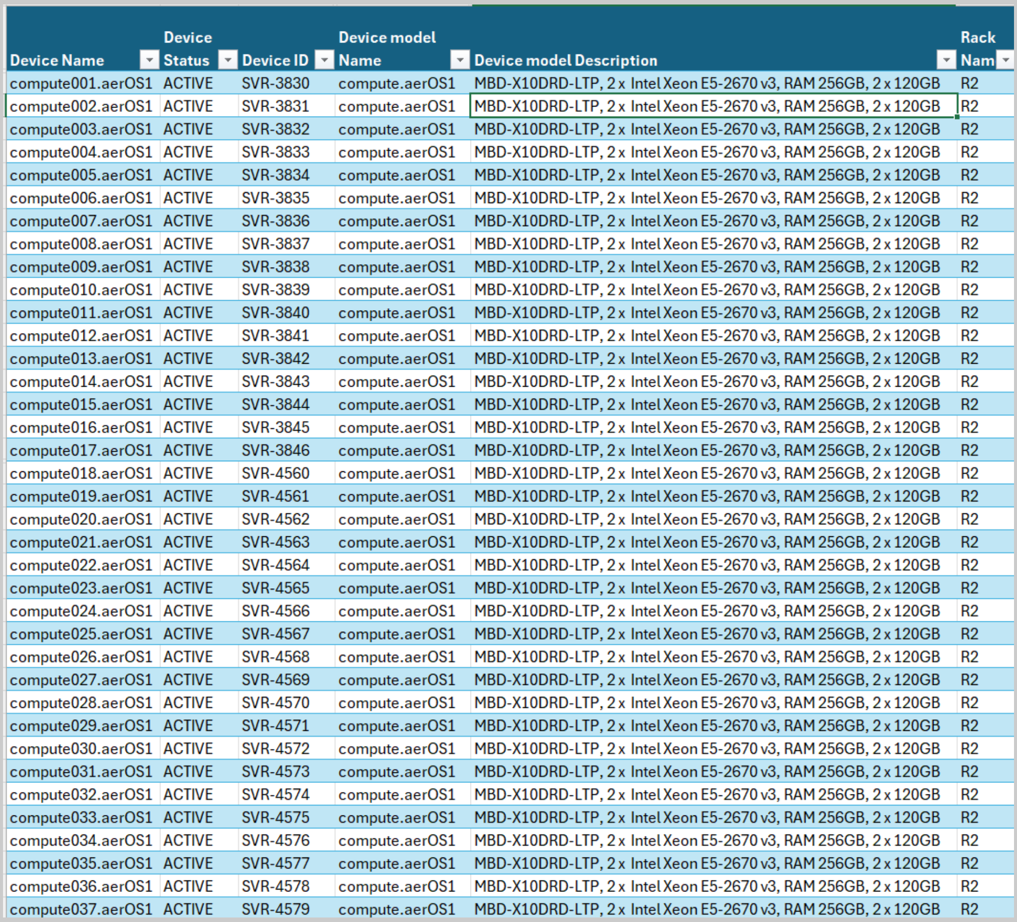
aerOS1:



Device Name	Device Status	Device ID	Device model Name	Device model Description	Rack Nam
ceph001.aerOS1	ACTIVE	SVR-3847	ceph ssd.aerOS1	Supermicro MBD-X10DRD-iTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 240GB, 22 x 960GB, 1 x CX-4 Lx Dual, 1 x Supermicro AOC-S3008L-L8E	R2
ceph002.aerOS1	ACTIVE	SVR-3848	ceph ssd.aerOS1	Supermicro MBD-X10DRD-iTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 240GB, 22 x 960GB, 1 x CX-4 Lx Dual, 1 x Supermicro AOC-S3008L-L8E	R2
ceph003.aerOS1	ACTIVE	SVR-4559	ceph ssd.aerOS1	Supermicro MBD-X10DRD-iTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 240GB, 22 x 960GB, 1 x CX-4 Lx Dual, 1 x Supermicro AOC-S3008L-L8E	R2

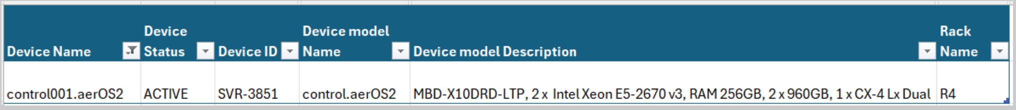


Device Name	Device Status	Device ID	Device model Name	Device model Description	Rack Nam
ceph001.aerOS1	ACTIVE	SVR-3847	ceph ssd.aerOS1	Supermicro MBD-X10DRD-iTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 240GB, 22 x 960GB, 1 x CX-4 Lx Dual, 1 x Supermicro AOC-S3008L-L8E	R2
ceph002.aerOS1	ACTIVE	SVR-3848	ceph ssd.aerOS1	Supermicro MBD-X10DRD-iTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 240GB, 22 x 960GB, 1 x CX-4 Lx Dual, 1 x Supermicro AOC-S3008L-L8E	R2
ceph003.aerOS1	ACTIVE	SVR-4559	ceph ssd.aerOS1	Supermicro MBD-X10DRD-iTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 240GB, 22 x 960GB, 1 x CX-4 Lx Dual, 1 x Supermicro AOC-S3008L-L8E	R2

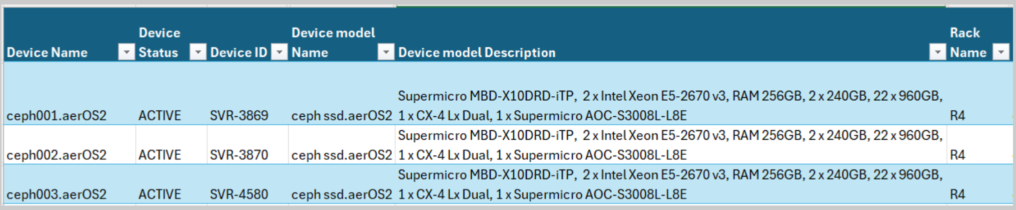


Device Name	Device Status	Device ID	Device model Name	Device model Description	Rack Nam
compute001.aerOS1	ACTIVE	SVR-3830	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute002.aerOS1	ACTIVE	SVR-3831	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute003.aerOS1	ACTIVE	SVR-3832	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute004.aerOS1	ACTIVE	SVR-3833	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute005.aerOS1	ACTIVE	SVR-3834	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute006.aerOS1	ACTIVE	SVR-3835	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute007.aerOS1	ACTIVE	SVR-3836	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute008.aerOS1	ACTIVE	SVR-3837	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute009.aerOS1	ACTIVE	SVR-3838	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute010.aerOS1	ACTIVE	SVR-3839	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute011.aerOS1	ACTIVE	SVR-3840	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute012.aerOS1	ACTIVE	SVR-3841	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute013.aerOS1	ACTIVE	SVR-3842	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute014.aerOS1	ACTIVE	SVR-3843	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute015.aerOS1	ACTIVE	SVR-3844	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute016.aerOS1	ACTIVE	SVR-3845	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute017.aerOS1	ACTIVE	SVR-3846	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute018.aerOS1	ACTIVE	SVR-4560	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute019.aerOS1	ACTIVE	SVR-4561	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute020.aerOS1	ACTIVE	SVR-4562	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute021.aerOS1	ACTIVE	SVR-4563	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute022.aerOS1	ACTIVE	SVR-4564	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute023.aerOS1	ACTIVE	SVR-4565	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute024.aerOS1	ACTIVE	SVR-4566	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute025.aerOS1	ACTIVE	SVR-4567	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute026.aerOS1	ACTIVE	SVR-4568	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute027.aerOS1	ACTIVE	SVR-4569	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute028.aerOS1	ACTIVE	SVR-4570	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute029.aerOS1	ACTIVE	SVR-4571	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute030.aerOS1	ACTIVE	SVR-4572	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute031.aerOS1	ACTIVE	SVR-4573	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute032.aerOS1	ACTIVE	SVR-4574	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute033.aerOS1	ACTIVE	SVR-4575	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute034.aerOS1	ACTIVE	SVR-4576	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute035.aerOS1	ACTIVE	SVR-4577	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute036.aerOS1	ACTIVE	SVR-4578	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2
compute037.aerOS1	ACTIVE	SVR-4579	compute.aerOS1	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R2

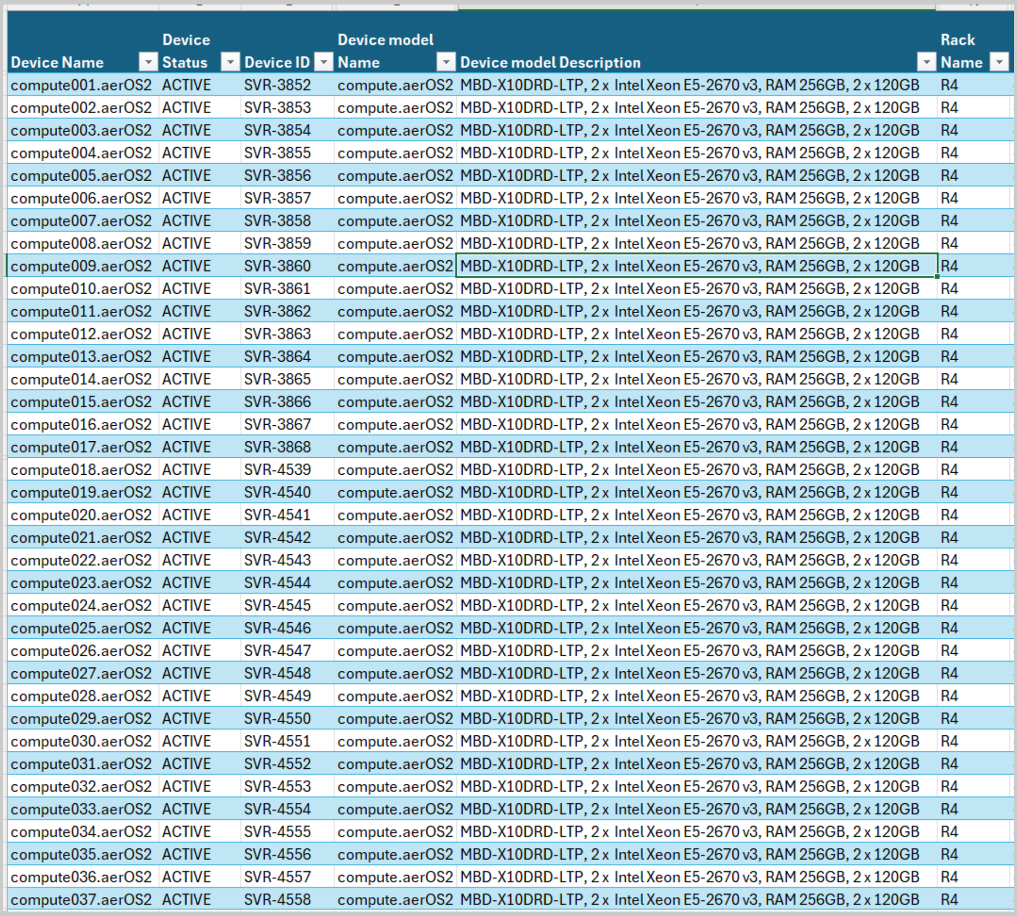
aerOS2:



Device Name	Device Status	Device ID	Device model Name	Device model Description	Rack Name
control001.aerOS2	ACTIVE	SVR-3851	control.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 960GB, 1 x CX-4 Lx Dual	R4



Device Name	Device Status	Device ID	Device model Name	Device model Description	Rack Name
ceph001.aerOS2	ACTIVE	SVR-3869	ceph ssd.aerOS2	Supermicro MBD-X10DRD-iTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 240GB, 22 x 960GB, 1 x CX-4 Lx Dual, 1 x Supermicro AOC-S3008L-L8E	R4
ceph002.aerOS2	ACTIVE	SVR-3870	ceph ssd.aerOS2	Supermicro MBD-X10DRD-iTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 240GB, 22 x 960GB, 1 x CX-4 Lx Dual, 1 x Supermicro AOC-S3008L-L8E	R4
ceph003.aerOS2	ACTIVE	SVR-4580	ceph ssd.aerOS2	Supermicro MBD-X10DRD-iTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 240GB, 22 x 960GB, 1 x CX-4 Lx Dual, 1 x Supermicro AOC-S3008L-L8E	R4



Device Name	Device Status	Device ID	Device model Name	Device model Description	Rack Name
compute001.aerOS2	ACTIVE	SVR-3852	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute002.aerOS2	ACTIVE	SVR-3853	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute003.aerOS2	ACTIVE	SVR-3854	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute004.aerOS2	ACTIVE	SVR-3855	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute005.aerOS2	ACTIVE	SVR-3856	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute006.aerOS2	ACTIVE	SVR-3857	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute007.aerOS2	ACTIVE	SVR-3858	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute008.aerOS2	ACTIVE	SVR-3859	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute009.aerOS2	ACTIVE	SVR-3860	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute010.aerOS2	ACTIVE	SVR-3861	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute011.aerOS2	ACTIVE	SVR-3862	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute012.aerOS2	ACTIVE	SVR-3863	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute013.aerOS2	ACTIVE	SVR-3864	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute014.aerOS2	ACTIVE	SVR-3865	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute015.aerOS2	ACTIVE	SVR-3866	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute016.aerOS2	ACTIVE	SVR-3867	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute017.aerOS2	ACTIVE	SVR-3868	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute018.aerOS2	ACTIVE	SVR-4539	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute019.aerOS2	ACTIVE	SVR-4540	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute020.aerOS2	ACTIVE	SVR-4541	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute021.aerOS2	ACTIVE	SVR-4542	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute022.aerOS2	ACTIVE	SVR-4543	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute023.aerOS2	ACTIVE	SVR-4544	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute024.aerOS2	ACTIVE	SVR-4545	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute025.aerOS2	ACTIVE	SVR-4546	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute026.aerOS2	ACTIVE	SVR-4547	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute027.aerOS2	ACTIVE	SVR-4548	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute028.aerOS2	ACTIVE	SVR-4549	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute029.aerOS2	ACTIVE	SVR-4550	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute030.aerOS2	ACTIVE	SVR-4551	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute031.aerOS2	ACTIVE	SVR-4552	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute032.aerOS2	ACTIVE	SVR-4553	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute033.aerOS2	ACTIVE	SVR-4554	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute034.aerOS2	ACTIVE	SVR-4555	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute035.aerOS2	ACTIVE	SVR-4556	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute036.aerOS2	ACTIVE	SVR-4557	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4
compute037.aerOS2	ACTIVE	SVR-4558	compute.aerOS2	MBD-X10DRD-LTP, 2 x Intel Xeon E5-2670 v3, RAM 256GB, 2 x 120GB	R4

KPI 2.2.7 Number of batch processing jobs successfully distributed and executed by the system

Table 99: KPI 2.2.7 Number of batch processing jobs successfully distributed and executed by the system

KPI ID number and partner resp.	**KPI 2.2.7**
KPI Name	**Number of batch processing jobs successfully distributed and executed by the system**
Description	The number of batch jobs scheduled, orchestrated and executed by aerOS continuum.
Motivation	Significant amount of batch jobs shows the ability of handling complex parallel computing tasks.

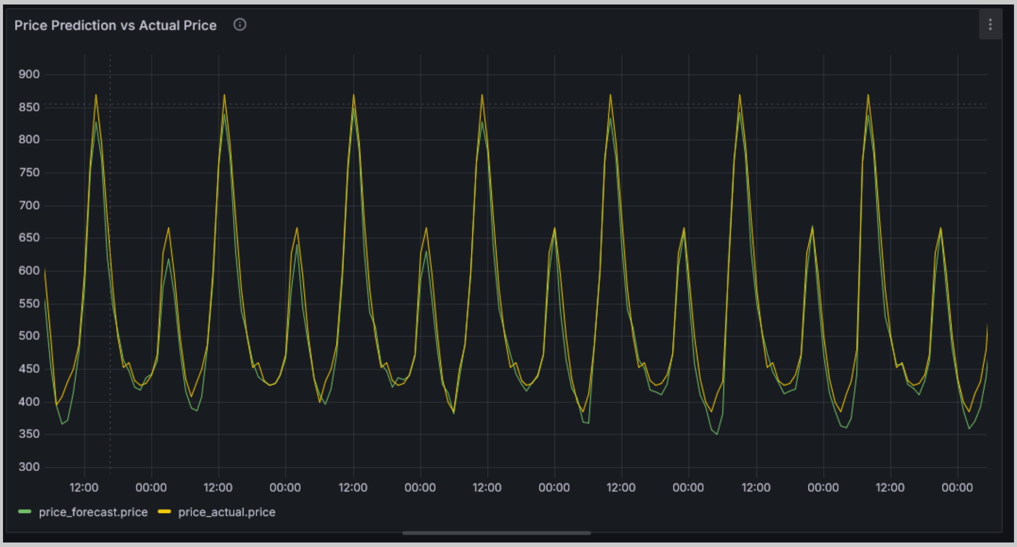
Target value	300k		
Prerequisites	Container connected to power, connected to the network. aerOS continuum installed and ready to use.		
aerOS components (task)	HLO (T3.3), LLO.		
Evaluation means	Task done will be counted based on logs and displayed on Grafana.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0***	***N/A***	***475.718***
Outcome elaboration (M38)	Scenario 1 processed over 475k cloud masks from Sentinel-2 products (SenSei2 model). One cloud mask is a one batch job. Processing is still ongoing aiming to process full Sentinel-2 archive for Poland area. Masks are publicly available at: https://s3.waw4-1.cloudferro.com/swift/v1/aeros-cloudmask-public/ 		

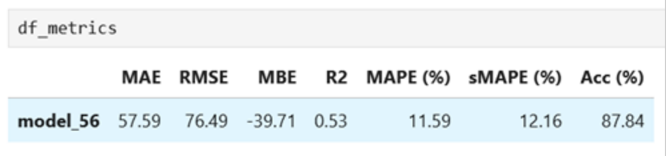
```
eouser@jumper:~$ date
Tue Oct 14 09:08:56 UTC 2025
eouser@jumper:~$ s5cmd --endpoint-url=https://s3.waw4-1.cloudferro.com ls s3://aeros-cloudmask-public/* | wc -l
475718
```

KPI 2.2.8 Precision of the Future Price prediction algorithm

Table 100: KPI 2.2.8 Precision of the Future Price prediction algorithm

KPI ID number and partner resp.	**KPI 2.2.8**		
KPI Name	**Precision of the Future Price prediction algorithm**		
Description	Precision of the price value predicted by running MLOps microservice compared to the actual price value published by the energy exchange the next day.		
Motivation	Significant deviation in predicted and actual value eliminates the usability of the microservice.		
Target value	85%		
Prerequisites	Containers infra operational, Electrum microservice for price estimation, aerOS runtime working, access to TGE and PSE work platform.		
aerOS components (task)	HLO and LLO (T3.3)		
Evaluation means	Price from TGE will be compared with estimated price of microservice. Grafana will be utilized for visualizing the estimations.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**

<table>
<tr><td>Measured value (% achieved)</td><td><i>0%</i></td><td><i>N/A</i></td><td><i>85%-90% (measured 87,74%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">The regression model used for energy price prediction shows good performance. The average prediction error is about 57.6 units, and the model tends to systematically underestimate prices by nearly 40 units. On a relative scale, the errors are around 11–12% (MAPE/sMAPE), giving an overall accuracy of ~88%, which is fairly strong in the context of volatile energy markets. However, the higher RMSE compared to MAE suggests the model occasionally struggles with sharp price spikes, which happens to be the challenge in energy related forecasting. After gathering enough data model should be considered for revision of NN architecture. R² of 0.53 points that model explains over half of the variability in prices. Given the input values are past.



df_metrics
<table>
<tr><th></th><th>MAE</th><th>RMSE</th><th>MBE</th><th>R2</th><th>MAPE (%)</th><th>sMAPE (%)</th><th>Acc (%)</th></tr>
<tr><td>model_56</td><td>57.59</td><td>76.49</td><td>-39.71</td><td>0.53</td><td>11.59</td><td>12.16</td><td>87.84</td></tr>
</table>
</td></tr>
</table>

Pilot 3 High performance computing platform for connected and cooperative mobile machinery

KPI 2.3.1.a (and KPI 2.3.2.b) Performance and connectivity capabilities improvement (single vehicle)

Table 101: KPI 2.3.1 Performance and connectivity capabilities improvement (single vehicle)

KPI ID number and partner resp.	**KPI 2.3.1.a and KPI 2.3.2.b**
KPI Name	**Performance and connectivity capabilities improvement (single vehicle)**
Description	This complex KPI includes 2 measures of capabilities of a single vehicle:

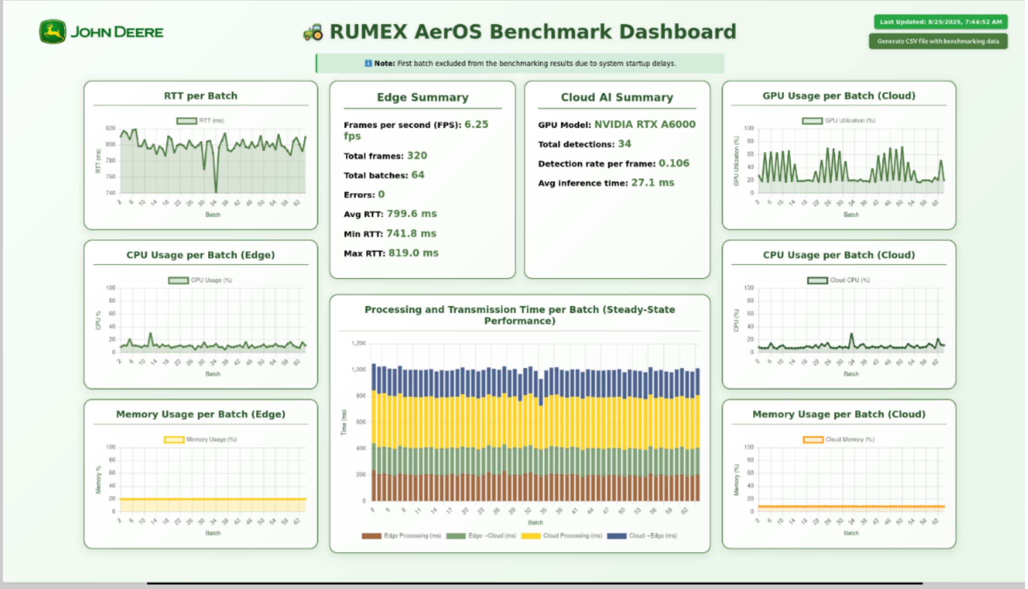
	– KPI 2.3.1.a Performance without using AI-supported application(s), where the improvement should be more than 20% – KPI 2.3.1.b Performance of the connectivity with temporary network infrastructure, meaning that high bandwidth connectivity e.g., 5G, should be available in rural environment to achieve the connectivity in so called dead areas with GPS (i.e., no connectivity at the moment)		
Motivation	For suggested KPI 2.3.1.a: As applications become more complex, they require more computing capabilities on the edge device. Mobile machinery for agriculture and construction applications poses hard challenges to developers of computers, because of the rugged environment and conditions in which they must operate. Measuring computing capabilities gives an indication on the innovation and engineering efforts expended in making the computer suitable for the aforementioned use-case. For suggested KPI 2.3.1.b: Connectivity is needed in order to realize the edge to cloud continuum which is the research topic in aerOS. Measuring the availability and sustained speed of network connectivity gives an indication of efforts spent in realizing the needed infrastructure both at the edge and at the cloud.		
Target value	For performance: GPU: 12.6 FP16 TFLOPS; CPU: SPEC int 2k6: 22, SPEC int rate: 140 Gflops. For connectivity: 4G/5G network available.		
Prerequisites	• Assembly and test of the prototype HW platform to be used in the pilot. • Integrating and testing of the required OS and libraries. • Availability of required interfaces between HW platform and other components with the target prototype vehicle		
aerOS components (task)	Networking (T3.1), Self-* (T3.5), Manageability (T4.6).		
Evaluation means	For performance: The performance is evaluated by integrating the TTControl platform, HPCP prototype extended with the NVIDIA-based packages, running aerOS software on prototype John Deere machines and executing the lab and field tests for John Deere's scenarios, while at the same time monitoring the computing resources utilization such as CPU, GPU and memory. Only with the execution of John Deere's applications, e.g. the sustainability impact can be measured. For connectivity: The connectivity using the temporary network will be tested by the aerOS SW and John Deere's applications (e.g. running operational instructions) on John Deere's prototype machines.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***For performance: GPU: 2x128 GFLOPS FP 16 CPU: 26000 DMIPS.***	***For performance: GPU: 12.6 FP16 TFLOPS; CPU: SPEC int 2k6: 22, SPEC int rate: 140 Gflops.***	***For performance: GPU: 12.6 FP16 TFLOPS; CPU: SPEC int 2k6: 22, SPEC int rate: 140 Gflops.***

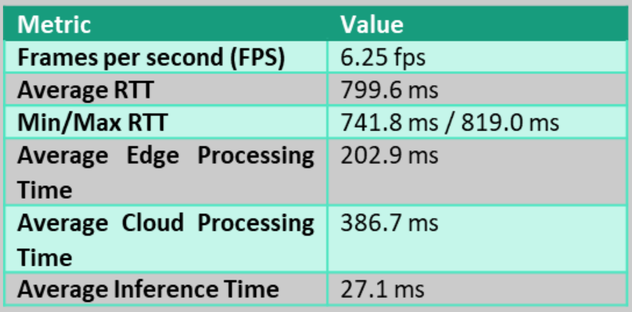
	For connectivity: No network available.	***For connectivity: 4G / 5G network available***	***For connectivity: 4G / 5G network available***
Outcome elaboration (M38)	The HW prototype are available for the laboratory use and testing and are currently used by both pilot partners TTControl and John Deere. The expected performance is achieved when executing the target SW in a laboratory setup.		

KPI 2.3.2 Swarm of vehicle performance improvement

Table 102: KPI 2.3.2 Swarm of vehicle performance improvement

KPI ID number and partner resp.	**KPI 2.3.2**
KPI Name	**Swarm of vehicle performance improvement**
Description	Performance using AI-supported application(s) to monitor and optimize the integration of AI-based solutions to enhance vehicle efficiency, and overall performance. This KPI helps to identify areas for improvement, to fine-tune the AI algorithms, and ensure a seamless operation experience for end users.
Motivation	Applications based on Artificial Intelligence methodologies (Deep neural networks) will be developed to process images coming from sensors on the mobile machinery (cameras). The metric of frames per second indicates how efficient the AI algorithms are as well as how powerful the hardware is that has been developed for the use on the mobile machine (which is subject to the same constraints mentioned in KPI 2.3.1 with respect to accommodating powerful processing in the challenging environment of agriculture and construction machines). By leveraging aerOS, the goal is to improve this frame rate by at least 20%, enabling faster processing and subsequently increasing the tractor's operation speed. This improvement is possible due to more frequent updates on the field status, resulting from the higher frame rate.
Target value	Target value: 6 FPS pro Camera and 18 km/h
Prerequisites	Finalization of the preparation and setup of two electric prototype tractors and the associated implements. • Integration of High-Performance ECUs in tractors to convert them to IEs. • Establishment of a 4G/5G private network on the test field. • Setup of on-premises and cloud IEs such as computing nodes and VMs. • Ensuring the aerOS runtime is integrated and fully operational.
aerOS components (task)	aerOS Basic Services (Data Fabric (T4.2), Federated Orchestration (HLO/LLO) (T3.3), Management Portal (T4.6), Services and configuration APIs (T3.2), Self-* modules (T3.5), aerOS Auxiliary Services (AI distributed inference)
Evaluation means	Lab and field tests will be conducted to assess the AI's performance in ensuring accurate and efficient field work operations. Multiple AI models, such as a model

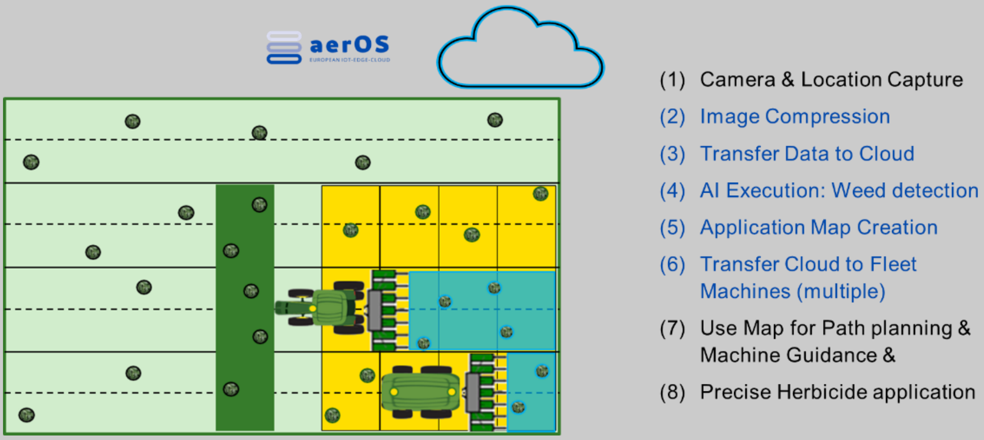
<table>
<tr><td></td><td colspan="3">designed to optimize tillage, will be employed to enhance different aspects of agricultural practices. The performance of the IEs, the prototype machines, the AI models, and the network infrastructure, such as achieved machine speed, compute resource utilization and required computation time, will be tracked by the embedded control software of the prototype machines and applications across the different IEs. The system's ability to adapt to real-time environmental and operational changes will be tested, along with scalability assessments to determine the capacity of aerOS to handle varying farm sizes and complexities. Impact evaluations will measure improvements in resource utilization efficiency.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>The baseline frame rate of 4 frames per second (FPS) per camera represents the current processing capacity for the exemplary task in the use case.</i></td><td><i>N/A</i></td><td><i>During the lab and field testing it was proved to increase the FPS to 6.25 by ensuring the field operating speed of 20km/h.</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">A monitoring framework was established to evaluate key performance parameters, including frames per second (FPS), image processing time, and round-trip latency. This setup enabled controlled testing of various AI model configurations and image resolutions within a laboratory environment. By leveraging the aerOS Edge-Cloud Continuum, the system was successfully optimized to achieve a 20% increase in FPS, enhancing real-time processing capabilities.

RUMEX AerOS Benchmark Dashboard
Last Updated: 8/25/2025, 7:44:52 AM
Note: First batch excluded from the benchmarking results due to system startup delays.
Edge Summary
Frames per second (FPS): 6.25 fps
Total frames: 320
Total batches: 64
Errors: 0
Avg RTT: 799.6 ms
Min RTT: 741.8 ms
Max RTT: 819.0 ms
Cloud AI Summary
GPU Model: NVIDIA RTX A6000
Total detections: 34
Detection rate per frame: 0.106
Avg inference time: 27.1 ms
 </td></tr>
</table>

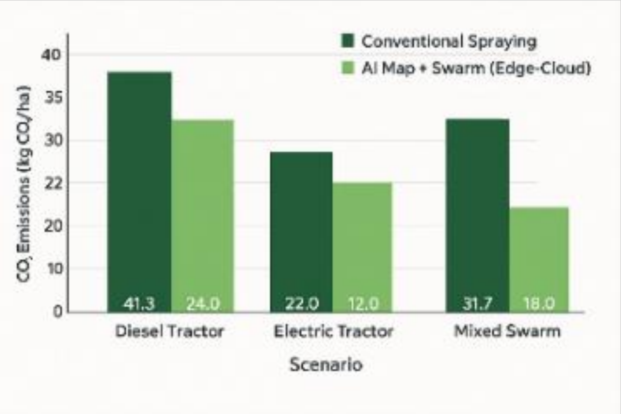
<table>
<tr><td></td><td>
<table>
<tr><th>Metric</th><th>Value</th></tr>
<tr><td>Frames per second (FPS)</td><td>6.25 fps</td></tr>
<tr><td>Average RTT</td><td>799.6 ms</td></tr>
<tr><td>Min/Max RTT</td><td>741.8 ms / 819.0 ms</td></tr>
<tr><td>Average Edge Processing Time</td><td>202.9 ms</td></tr>
<tr><td>Average Cloud Processing Time</td><td>386.7 ms</td></tr>
<tr><td>Average Inference Time</td><td>27.1 ms</td></tr>
</table>
<p>Following the successful verification of the laboratory configuration, the system was integrated into our field operations. During this phase, the same performance parameters, such as frames per second (FPS), image processing time, and round-trip latency were systematically measured under real-world conditions. This allowed us to validate the consistency and reliability of the setup outside the controlled lab environment and confirm that the optimizations achieved during testing translated effectively to operational use.</p></td></tr>
</table>

KPI 2.3.3 CO2 emissions reduction thanks to platooning

Table 103: KPI 2.3.3 CO2 emissions reduction thanks to platooning

KPI ID number and partner resp.	**KPI 2.3.3**
KPI Name	**CO2 emissions reduction thanks to platooning**
Description	CO2 indicators to measure and track the CO2 emissions and subsequent reduction due to the utilization of electric tractors and the aerOS services. Here in particular for the cultivating/grubbing activity during stubble cultivation.
Motivation	The motivation behind the CO2 emissions reduction KPI in the aerOS project is to quantify and assess the environmental impact of deploying the aerOS solution and transitioning from diesel-powered tractors to electric tractors. Climate change and environmental conservation are increasingly important global concerns, and the reduction of CO2 emissions is a crucial step towards addressing these challenges.
Target value	A reduction of 80% - 17,9 kg CO2/ha
Prerequisites	• Finalization of the preparation and setup of two electric prototype tractors and the associated implements. • Integration of High-Performance ECUs in tractors to convert them to IEs. • Establishment of a 4G/5G private network on the test field. • Setup of on-premise and cloud IEs such as computing nodes and VMs. • Ensuring the aerOS runtime is integrated and fully operational.
aerOS components (task)	aerOS Basic Services (Data Fabric (T4.2), Federated Orchestration (HLO/LLO) (T3.3), Management Portal (T4.6), Services and configuration APIs (T3.2), Self-* modules (T3.5), aerOS Auxiliary Services (T4.3 AI distributed inference)

<table>
<tr><td>Evaluation means</td><td colspan="3">Lab and field tests will be conducted to evaluate capabilities of aerOS in optimizing resource utilization and real-time adaptation of machine operations. These evaluations focus on optimizing resource use and dynamically adjusting operations to changing conditions. The impact on reducing CO2 emissions through sustainable practices is measured by monitoring the power consumption and operational performance of the prototype machines during field and overall operations. The data is tracked by the embedded control software of the machines, ensuring operational quality remains high.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>89,31 kg CO2/ha</i>
<i>(33,7 l Diesel/ha)</i></td><td><i>N/A, because this KPI was defined for M34.</i></td><td><i>Thanks to the implementation of the aerOS components we could measure the following results for 40% CO2 reduction for diesel and electric tractors in a swarm environment.</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">For the evaluation, a spatially accurate prescription map was generated using the aerOS edge-cloud continuum and AI-based image analysis (e.g., from satellite, drone, or tractor-mounted cameras) that identifies:
• Weed or pest hotspots
• Crop health variability
• Soil moisture or nutrient zones
This map is then used to control spraying intensity and location for multiple machines The following setup was conducted:

The following data was captured on the machines to evaluate the KPI:</td></tr>
</table>

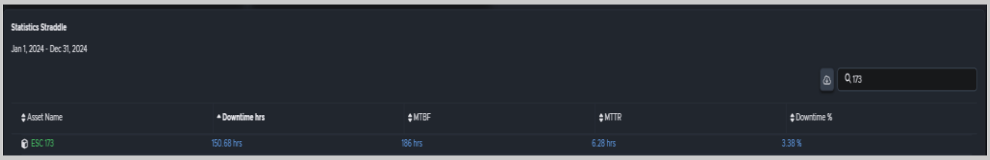
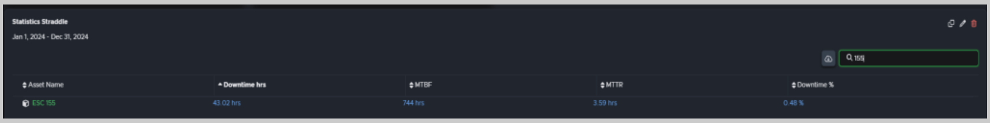
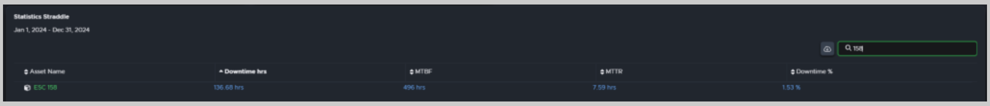
	 Assuming a baseline CO_2 emission of 41.3 kg/ha for diesel tractor spraying: With GPS guidance there was an achievement of 20% reduction: 41.3×0.80=33.0 kg CO_2/ha | 41.3-33.0 kg CO_2/ha → Savings: ~8.3 kg CO_2/ha With basic overlap reduction due to capturing the already deployed herbicides there was an achievement of 2–7% reduction: 41.3×0.93=38.4 kg CO_2/ha | 41.3-38,4 kg CO_2/ha → Savings: ~2.9 kg CO_2/ha With swarm coordinated ground spraying and task distribution a reduction of 15% was achieved: 41.3×0.85=35.1 kg CO_2/ha | 41.3-35,1 kg CO_2/ha → Savings: ~6.2 kg CO_2/ha

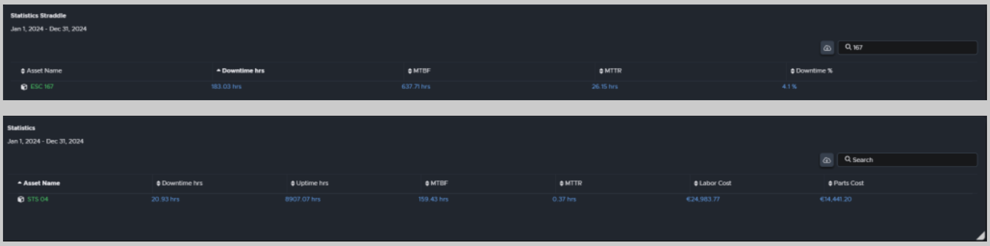
Pilot 4 Smart edge services for the port continuum

KPI 2.4.1 Reduction of CHE idle time due to failures

Table 104: KPI 2.4.1 Reduction of CHE idle time due to failures

KPI ID number and partner resp.	**KPI 2.4.1**
KPI Name	**Reduction of CHE idle time due to failures**
Description	The preventive maintenance tool and approach used in EUROGATE Container Limassol plans the maintenance task according to some number of working hours. This sub-optimal approach is frequently not enough for removing any unexpected failure of Container Handling Equipment (CHE) components, and undesired idle times at operational hours occur. aerOS predictive maintenance models are expected to reduce these IDLE times.
Motivation	It will show how the predictive maintenance on the edge service to be deployed in the project provides a relevant benefit to EUROGATE operational efficiency.
Target value	20-30%
Prerequisites	All sensors are deployed in the CHEs under tests, and their maintenance associated data is acquired and collected for AI modelling. Access to the CMMS system of EUROGATE is also required.

aerOS components (task)	OpenAPI (T3.2), AAA (T3.4), Context Broker (T4.2), Data Fabric (T4.2)		
Evaluation means	A report of the idle time of CHEs under study in the project due to maintenance tasks will be extracted for the first two years of the project from the Computerised Management System (CMMS) of EUROGATE. Once the predictive maintenance service from aerOS is deployed, an analysis between the original idle times with the preventive maintenance and the new ones with the predictive maintenance will be carried out. If the idle time of the use cases of the pilot is reduced at least 20%, the KPI will be considered as fulfilled.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***Total 2023 downtime of 4 straddle Carriers: 900h*** ***Total 2023 downtime of 2 STS: 297.70h***	***Q1-Q2 2024 downtime of 4 straddle Carriers: 514h*** ***Q1-Q2 2024 downtime of 2 STS: 69.1h***	***Q1-Q2 2025 downtime of 4 Straddle Carriers: 403h*** ***Q3 2025 downtime of 2 STS: 46 hours*** ***Straddle carriers 21.5% less downtime hours*** ***STS 31.3% less downtime hours*** ***Average 26.4% less downtime hours***
Outcome elaboration (M38)	5 different AI models have been implemented for predictive maintenance: 4 for the straddle carriers deployed during first quarter of 2025 (anomalies on the hydraulic system, engine, brakes, and parking brakes overtemperature), and 1 for the STS crane (trolley wire rope elongation) in third quarter of 2025. To evaluate the potential lower downtime hours due to their use, abnormal situations with respect to the previous year were obtained from EUROGATE's CMMS system and compared with the reports since the models were in place. As it can be seen in the table below, the use of predictive maintenance over the straddle carriers helps on reducing the downtime hours from 514h during Q1-Q2 2024, to 403h during Q1-Q2 2025, leading to a reduction of 21.5%. In parallel, the use of predictive maintenance over the STS cranes helps on reducing the downtime hours from 67h during Q3 2024, to 46h during Q3 2025, leading to a reduction of 31.3%. Hence, the use of all the AI models in EUROGATE premises, lead to an average reduction of 26.4% in CHEs downtime hours.  Statistics Straddle Jan 1, 2024 - Dec 31, 2024 173 Asset Name Downtime hrs MTBF MTTR Downtime % ESC 173 150.68 hrs 186 hrs 6.28 hrs 3.38 %  Statistics Straddle Jan 1, 2024 - Dec 31, 2024 155 Asset Name Downtime hrs MTBF MTTR Downtime % ESC 155 43.02 hrs 744 hrs 3.59 hrs 0.48 %  Statistics Straddle Jan 1, 2024 - Dec 31, 2024 158 Asset Name Downtime hrs MTBF MTTR Downtime % ESC 158 136.68 hrs 496 hrs 7.59 hrs 1.53 %		

Statistics Straddle
Jan 1, 2024 - Dec 31, 2024

🔍 167

Asset Name	Downtime hrs	MTBF	MTTR	Downtime %
ESC 167	183.03 hrs	637.71 hrs	26.15 hrs	4.1 %

Statistics
Jan 1, 2024 - Dec 31, 2024

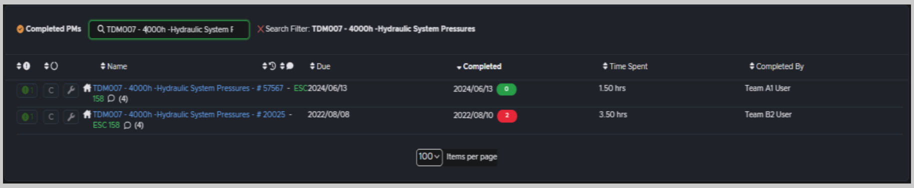
🔍 Search

Asset Name	Downtime hrs	Uptime hrs	MTBF	MTTR	Labor Cost	Parts Cost
STS 04	20.93 hrs	8907.07 hrs	159.43 hrs	0.37 hrs	€24,983.77	€14,441.20

KPI 2.4.2 Increase on detection of equipment malfunctions (from manual to automatic)

Table 105: KPI 2.4.2 Increase on detection of equipment malfunctions (from manual to automatic)

KPI ID number and partner resp.	**KPI 2.4.2**		
KPI Name	**Increase on detection of equipment malfunctions (from manual to automatic)**		
Description	The maintenance of EUROGATE CHEs is based on preventive inspection. The predictive maintenance service will be able to detect equipment malfunctions more precisely.		
Motivation	It will show how the predictive maintenance on the edge service to be deployed in the project provides a relevant benefit to EUROGATE operational efficiency.		
Target value	30-40% with respect to 2023		
Prerequisites	All sensors are deployed in the CHEs under tests, and their maintenance associated data is acquired and collected for AI modelling. Access to the CMMS system of EUROGATE is also required.		
aerOS components (task)	OpenAPI (T3.2), AAA (T3.4), Context Broker (T4.2), Data Fabric (T4.2), Explainability service (T4.3)		
Evaluation means	A comparative analysis between the manual equipment malfunctions reported for the four CHEs under study in the project versus the automatic ones provided by the predictive maintenance service will be conducted. If the proper identification of malfunctions is increased at least 30%, the KPI will be considered as fulfilled		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***In 2023: 30 unplanned failures detected (manual), 0 predictive (automatic)***	***Q1-Q2 2024: 15 unplanned failures detected (manual), 0 predictive (automatic)***	***Q1-Q2 2025: 20 unplanned failures detected (manual), 8 predictive (automatic) = 28 detected → +86% detections***
Outcome elaboration (M38)	In this KPI, due to the lack of enough time for monitoring the use of STS failures, only the PdM models of the straddle carriers has been used for the assessment. In that sense, thanks to them, abnormal hydraulic situations were spotted, and EGCTL maintenance		

	scheduled maintenance tasks for the associated CHEs. Consequently, from the 15 unplanned failures detected in Q1-Q2 2024, aerOS has led to the detection of 28 failures, both manual and automatic, leading to an increase of 86% during Q1-Q2 2025 with respect to the same period in previous year. Before and After Time required for 4000h Maintenance Service  Completed PMs TDM007 - 4000h -Hydraulic System F Search Filter: TDM007 - 4000h -Hydraulic System Pressures Name · Due · Completed · Time Spent · Completed By TDM007 - 4000h -Hydraulic System Pressures - # 57567 - ESC 158 (4) · 2024/06/13 · 2024/06/13 0 · 1.50 hrs · Team A1 User TDM007 - 4000h -Hydraulic System Pressures - # 20025 - ESC 158 (4) · 2022/08/08 · 2022/08/10 2 · 3.50 hrs · Team B2 User 100 Items per page Hydraulic Issues detected by observation.

KPI 2.4.3 Increase of number of actual damaged containers (manually reported by staff vs automatic system-reports)

Table 106: KPI 2.4.3 Increase of number of actual damaged containers (manually reported by staff vs automatic system-reports)

KPI ID number and partner resp.	**KPI 2.4.3**
KPI Name	**Increase of number of actual damaged containers (manually reported by staff vs automatic system-reports)**
Description	When loading/discharging containers to/from vessels by Ship-to-Shore (STS) cranes, a manual inspection by the port stevedores is carried out in order to confirm there are no wrong seals and damages generated during the manoeuvre. These reports will be more accurate if an automatic system which makes use of cameras and computer vision functionalities is deployed by aerOS.
Motivation	It will show how the Computer Vision (CV) on the edge service to be deployed in the project provides a relevant benefit to EUROGATE business.
Target value	30-40%
Prerequisites	All cameras are deployed under STS crane operations, their video streams are received in an AV server, and the accurate enough CV models are available and in execution. In addition, access to the ERP system data of EUROGATE is also need for data comparison.
aerOS components (task)	Context Broker (T4.2), Data Fabric (T4.2), HLO and LLO (T3.3), Model Reduction service (T4.3), Management portal (T4.6)
Evaluation means	A historical analysis of EUROGATE customers complaints, which demand penalties for not complying with the SLAs will be collected for the first two years of the project from the ERP system of EUROGATE. A comparison between the number of penalties received before and after the CV service from aerOS is deployed will be carried out. If the proper detection of actual damaged containers leads to a reduction of 30% of complaints procedures, the KPI will be considered as fulfilled.

<table>
<tr><th>Measurement period</th><th>Baseline</th><th>M24 (Deliverable D5.5)</th><th>M38 (Deliverable D5.6)</th></tr>
<tr><td>Measured value (% achieved)</td><td>350 damaged containers reported by terminal staff + 30 damaged containers not reported and claimed</td><td>Q1-Q4 2024: 530 damaged containers reported by terminal staff + 36 damaged containers not reported and claimed (not using CV models)</td><td>Q1-Q3 2025:396 damaged containers reported by terminal staff + 37 damaged containers not reported and claimed (not using CV models).
Oct 17-24, 2025: 11 damaged containers reported by terminal staff. 60 damaged containers reported by CV. Increase: 445%</td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">
<p>During the period from October 17, 2025, to October 24, 2025, around 12 thousand containers were discharged or loaded from two cranes at the Port of Limassol. The terminal staff manually checked the containers for damage during discharge related to side bends, dents, or holes, which stemmed from violent or negligent handling. The terminal staff identified 11 containers with damage.</p>
<p>The CV models tracked the containers using live video feed from two cameras installed on each crane and were able to identify 60 containers with real damages (bends/dents or holes). At the same time, however, the CV models also incorrectly reported 449 containers as having damage. The table below summarizes the results:</p>
<table>
<tr><th></th><th colspan="3">Bend/Dent</th><th colspan="3">Hole</th><th>Grand Total</th></tr>
<tr><td>Crane</td><td>TP</td><td>FP</td><td>Total</td><td>TP</td><td>FP</td><td>Total</td><td></td></tr>
<tr><td>STS 4</td><td>12</td><td>42</td><td>54</td><td>4</td><td>6</td><td>10</td><td>64</td></tr>
<tr><td>STS 5</td><td>35</td><td>371</td><td>406</td><td>9</td><td>30</td><td>39</td><td>445</td></tr>
<tr><td>Grand Total</td><td>47</td><td>413</td><td>460</td><td>13</td><td>36</td><td>49</td><td>509</td></tr>
</table>
<p>Our first observation is that the CV models running on the STS 5 crane have a much higher false positive (FP) rate compared to STS 4. The models were trained from video footage originating only from STS 4, and the camera angles between the two cranes are slightly different, which seems to negatively impact the models. Retraining the models with video footage from STS 5 should alleviate this issue. Moreover, we identified two major triggers of FPs. The first one is the design of some logos that appear on the side of the containers, which the models mistake for bends. The second one is shadows from different objects (e.g., straddle carriers, trucks), which the models mistake as either bends or holes. Additional training with these scenarios should further alleviate the issue.</p>
<p>Despite the high false positive rate, the CV models were able to identify a significantly higher number of damaged containers than the technical staff. Reviewing the cases that have been missed, the oversight of reporting damages by terminal staff does not seem to pertain to a specific type of damage but rather to fluctuations in personnel attention, which tends to decrease during specific times of the day.</p>
</td></tr>
</table>


msc
MSMU 497361 5
45G1
CAUTION HIGH TRAILER


08-06-2024 Sat 18:54:31
stinis
MEDITERRANEAN SHIPPING CO
msc
QC4 Gantry Cam1


APZU 337144 7
22G1
MAX. GROSS
30.480 KGS
67.200 LBS
TARE
2.220 KGS
4.890 LBS
NET
28.260 KGS
62.310 LBS
CU.CAP.
33.2 CU.M.
1.170 CU.FT.
AMERICAN BUREAU OF SHIPPING
CIMC

KPI 2.4.4 Performance evaluation metrics of regression AI models (R2)

Table 107: KPI 2.4.4 Performance evaluation metrics of regression AI models (R2)

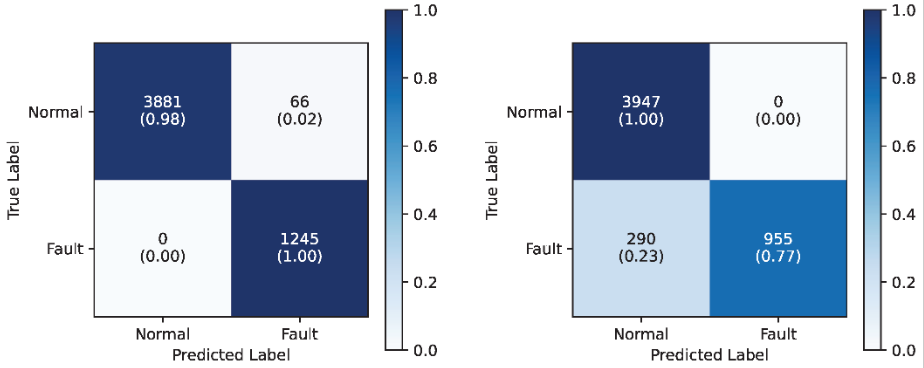
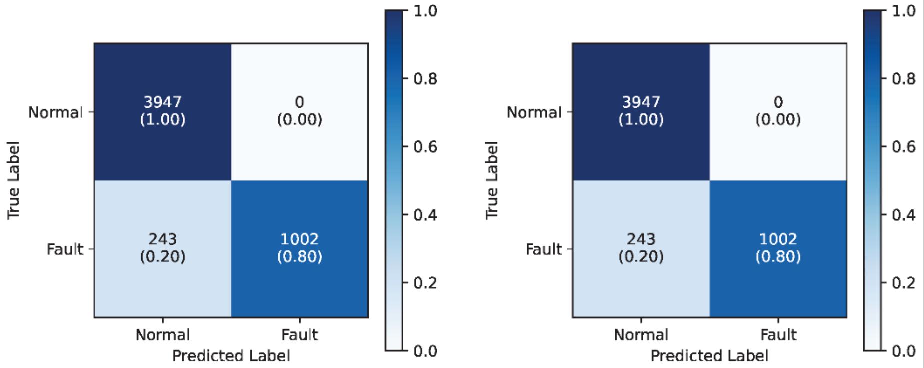
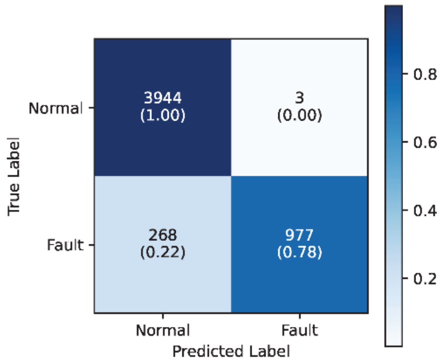
KPI ID number and partner resp.	**KPI 2.4.4**
KPI Name	**Performance evaluation metrics of regression AI models (R2)**
Description	Different regression AI models will be developed and deployed at the edge nodes of the Port Continuum domain. These models shall be accurate enough to predict equipment malfunctions before any failure occur. In regression models, R-square (R2) corresponds to the squared correlation between the observed outcome values and the predicted values by the model. The Higher the R-squared, the better the model.
Motivation	An accurate regression model should be provided in order to replace the current preventive maintenance for the new one developed in the project.
Target value	0.8

<table>
<tr><td>Prerequisites</td><td colspan="3">Large amount of time-series data from the CHEs shall be collected for ML model training. A Python script in charge of accuracy validation shall also be available.</td></tr>
<tr><td>aerOS components (task)</td><td colspan="3">Data Fabric (T4.2), Explainability service (T4.3)</td></tr>
<tr><td>Evaluation means</td><td colspan="3">From the different CHEs’ telemetry dataset collected, a portion of them will be used for validation purposes. Python-based Jupyter notebooks will be used for evaluating the R2 metric of the developed model against these validation datasets. As long as the model surpassed R2>=0.8, the KPI will be fulfilled.</td></tr>
<tr><td>Measurement period</td><td>Baseline</td><td>M24 (Deliverable D5.5)</td><td>M38 (Deliverable D5.6)</td></tr>
<tr><td>Measured value (% achieved)</td><td><i>N/A</i></td><td><i>N/A</i></td><td><i>83.3% (110%)</i></td></tr>
<tr><td>Outcome elaboration (M38)</td><td colspan="3">
<p>Most mechanical actuators inside straddle carriers are hydraulic, including the parking brake, steering, and spreader. For this reason, monitoring and predicting the status and health of the hydraulic system are crucial for the operation of the machine. Some events, such as a leak or a clogged fitting, can have a noticeable impact on the pressure of the whole system or the duty cycle of the pump. Thus, by monitoring the duty cycle of the pump (which under normal operation only fires in pulses), some of those issues were detected before causing significant damage. In that sense, faced with the lack of high-quality labels, an unsupervised anomaly detection method was developed. The dataset consisted of time series for the four hydraulic signals measured every 100 ms from June to October 2024, totalling 4 GB. This approach generated a list of potentially anomalous timestamps, each assigned a likelihood score indicating their rarity as anomalies, as well as involving a domain expert that manually reviewed a subset of flagged timestamps over a continuous 5-month period (June–October 2024).</p>
<p>In the table below, the binary metrics lead to a model with low recall due to a large quantity of false positives but a high precision. In that sense, for this particular case study, the most relevant metric is precision since the cost of a false negative is significantly higher than that of a false positive.</p>
<p>These results demonstrate that our unsupervised technique effectively identifies meaningful anomalies even in the absence of labelled training data.</p>
<table>
<tr><td colspan="4">Confusion Matrix</td><td colspan="2">Binary classification matrix</td></tr>
<tr><td colspan="4">Predicted</td><td><i>Metric</i></td><td><i>Value</i></td></tr>
<tr><td></td><td><i>Positive</i></td><td><i>Negative</i></td><td><i>Total</i></td><td><i>Accuracy</i></td><td>91.8%</td></tr>
<tr><td><i>Positive</i></td><td>5</td><td>1</td><td>6</td><td><i>Precision</i></td><td>83.3%</td></tr>
<tr><td><i>Negative</i></td><td>40</td><td>453</td><td>493</td><td><i>Recall</i></td><td>11.1%</td></tr>
<tr><td>Total</td><td>45</td><td>454</td><td>499</td><td><i>Specificity</i></td><td>99.8%</td></tr>
</table>
</td></tr>
</table>

KPI 2.4.5 Performance evaluation metrics of regression AI models (MAE/RMSE) for predictive maintenance of CHEs

Table 108: KPI 2.4.5 Performance evaluation metrics of regression AI models (MAE/RMSE) for predictive maintenance of CHEs

KPI ID number and partner resp.	**KPI 2.4.5**		
KPI Name	**Performance evaluation metrics of regression AI models (MAE/RMSE) for predictive maintenance of CHEs**		
Description	Different regression AI models will be developed and deployed at the edge nodes of the Port Continuum domain. These models shall be accurate enough to predict equipment malfunctions before any failure occur. Similarly to R2, other commonly used evaluation metric for regression models is Mean Average Error or Root Mean Squared Error (MAE/RMSE). Both metrics refer to refers to the mean of the absolute values of each prediction error on all instances of the test dataset. The lower the MAE/RMSE, the better the model.		
Motivation	An accurate regression model should be provided in order to replace the current preventive maintenance for the new one developed in the project.		
Target value	20%		
Prerequisites	Large amount of time-series data from the CHEs shall be collected for ML model training. A Python script in charge of accuracy validation shall also be available.		
aerOS components (task)	Data Fabric (T4.2), Explainability service (T4.3)		
Evaluation means	From the different CHEs' telemetry dataset collected, a portion of them will be used for validation purposes. Python-based Jupyter notebooks will be used for evaluating the MAE/RMSE metric of the developed model against these validation datasets. As long as the error of the model <20%, the KPI will be considered as fulfilled.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***N/A***	***F1-score: 98.0%***
Outcome elaboration (M38)	The aim was to develop ML models to predict inverter overtemperature faults caused by fan failures, clogged filters, and cooling issues. Real-world telemetry data for one years and for 15 straddle carriers at the EUROGATE Container Terminal Limassol were collected for training and validation purposes, such as inverter, motor, and engine temperatures, speed, torque, hydraulic pressure, and various error flags. Past failure incidents were identified from the Computerized Maintenance Management System (CMMS) and data records were manually labelled as normal or faulty. Five ML models were tuned, trained, and tested, namely, Artificial Neural Network (ANN), Decision Tree (DT), Random Forest (RF), Extreme Gradient Boosting (XGBoost), and Gaussian Naive Bayes (GNB). From all models, the ANN model achieved the highest performance with 98.7% accuracy and 98.0% F1-score, while RF and XGBoost also delivered strong results with same accuracy of 95.32% and 93.0% F1-score. SHAP analysis confirmed that inverter and motor temperatures were the most influential predictive features, supported by ambient temperature and engine error slots. Overall, the outcome demonstrates that ML models, particularly ANN, can reliably identify		

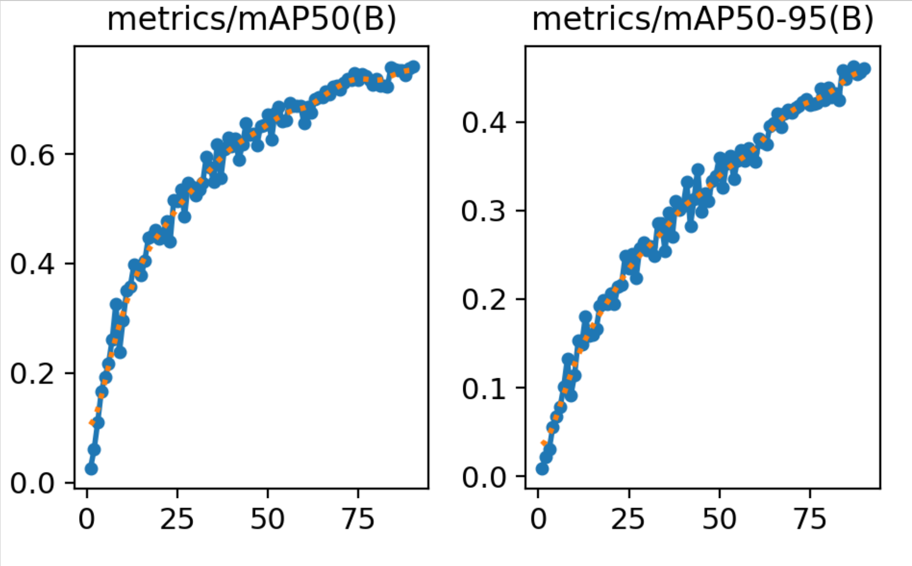
	inverter faults in advance, significantly reducing the risk of unplanned downtime in port operations. Hence, this KPI has been fulfilled.  (a) ANN (b) DT  (c) RF (d) XGBoost  (e) GNB

KPI 2.4.6 Performance evaluation metrics of classification AI models (accuracy) for damaged containers

Table 109: KPI 2.4.6 Performance evaluation metrics of classification AI models (accuracy) for damaged containers

KPI ID number and partner resp.	**KPI 2.4.6**
KPI Name	**Performance evaluation metrics of classification AI models (accuracy) for damaged containers**

Description	Different classification AI models will be developed and deployed at the edge nodes of the Port Continuum domain. These models shall be accurate enough to detect and classify damages identified at containers' surfaces. Accuracy is one of the most common metrics used for the evaluation of these classification models. AI accuracy is the degree to which an AI system produces correct outputs or predictions based on the given inputs or data. Therefore, if the AI system classifies damages, its accuracy is the percentage of images that the model correctly labels as dents, etc.		
Motivation	An accurate classification model should be provided in order to replace guarantee the proper detection and classification of surfaces across the loaded/unloaded containers.		
Target value	60%		
Prerequisites	Large amount of video streams recorded needed for CV models ML training shall be available, especially with damages visible on containers' surfaces as part of the data set.		
aerOS components (task)	N/A		
Evaluation means	From the different videos collected from the cameras deployed in the dock area, a portion of them will be used for validation purposes. TensorFlow evaluation tool that offers various libraries for model validation, testing, and evaluation will be used. As long as the accuracy of the model > 60%, the KPI will be considered as fulfilled.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***mAP50: 53%***	***mAP50: 75%***
Outcome elaboration (M38)	Two models were developed, one for container detection and one for damage detection. Both models utilize a pre-trained YOLOv12 model. To prepare the data for the models, we collected videos taken throughout the year 2024, at different times of day and night, and under different weather conditions. We converted these videos into frames, which were manually labelled with containers and damages to create two datasets. We recorded three main types of container damage that are of interested to the container terminal, namely holes, bents, dents, and normal wear. The containers dataset contains 2470, 598, and 786 images for training, validation, and testing, respectively, while the damage dataset includes 1442, 182, and 140 images for training, validation, and testing, respectively. The container detection model achieved very strong validation results with a mAP50 score of 99%. Note that mAP50 measures the mean average precision at an intersection over union (IoU) threshold of 0.5. The damage detection model achieved a mAP50 score of 75%, which is higher than the desired 60% score. Hence, this KPI has been fulfilled. Validation results for damage detection:		

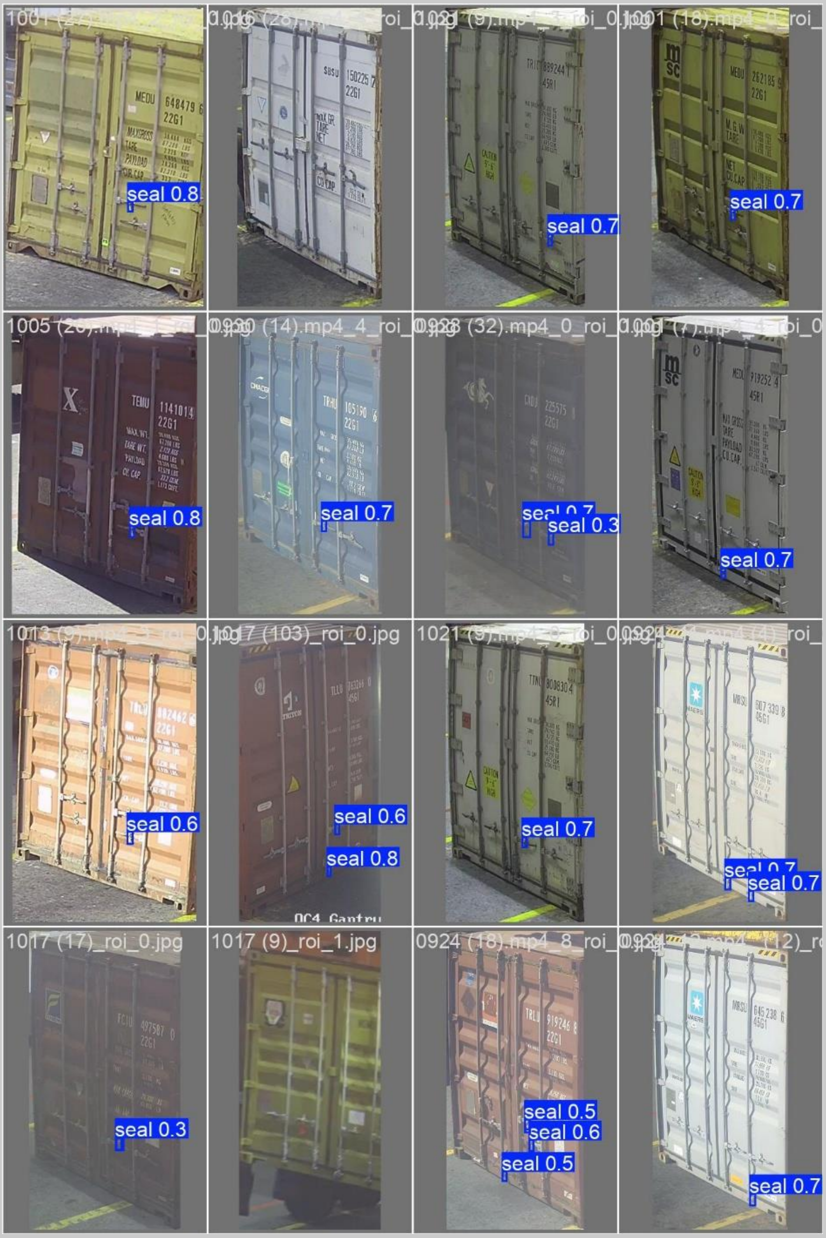
	 mAP50 and mAP50-95 for damage detection: 

KPI 2.4.7 Performance evaluation metrics of classification AI models (F1) for damaged seals

Table 110: KPI 2.4.7 Performance evaluation metrics of classification AI models (F1) for damaged seals

KPI ID number and partner resp.	**KPI 2.4.7**
KPI Name	**Performance evaluation metrics of classification AI models (F1) for damaged seals**
Description	While accuracy is often used as a primary indicator of the quality and effectiveness of an AI system, there are other metrics like precision and recalls that help to evaluate the quality of a model. This KPI will evaluate the F1 score. F1 balances the trade-off between precision and recall, which can vary depending on the model and the data.

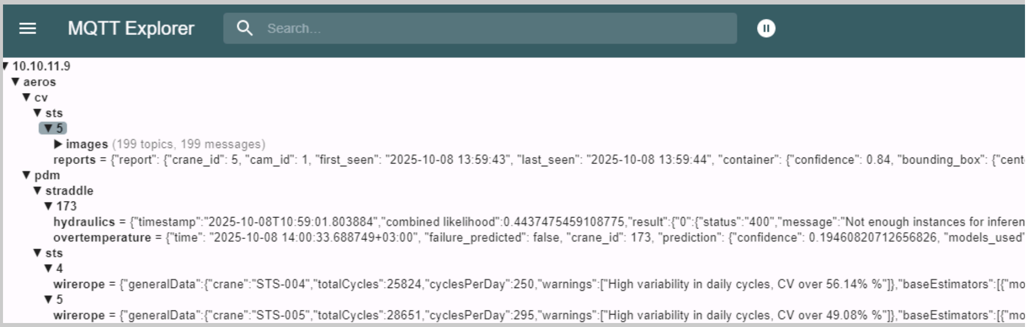
Motivation	An accurate classification model should be provided in order to replace guarantee the proper detection and classification of surfaces across the loaded/unloaded containers.		
Target value	60%		
Prerequisites	Large amount of video streams recorded needed for CV models ML training shall be available, especially with wrong or damaged seals included in the data set.		
aerOS components (task)	N/A		
Evaluation means	From the different videos collected from the cameras deployed in the dock area, a portion of them will be used for validation purposes. TensorFlow evaluation tool that offers various libraries for model validation, testing, and evaluation will be used. As long as the F1 of any of the model developed in the project is > 60%, the KPI will be considered as fulfilled.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***N/A***	***mAP50: 86%***
Outcome elaboration (M38)	Two models were developed, one for container door detection and one for seal detection. Both models utilize a pre-trained YOLOv12 model. To prepare the data for the models, videos taken throughout the year 2024 were collected at different times of day and night, and under different weather conditions. These videos were converted into frames, which were manually labelled with container doors and seals to create two datasets. The container doors dataset contains 215, 61, and 22 images for training, validation, and testing, respectively, while the seal dataset includes 422, 54, and 31 images for training, validation, and testing, respectively. The container door detection model achieved strong validation results with a mAP50 score of 96%. Note that mAP50 measures the mean average precision at an intersection over union (IoU) threshold of 0.5. The seal detection model achieved a mAP50 score of 86%, which is higher than the desired 60% score. Hence, this KPI has been fulfilled. Validation results for seal detection:		

KPI 2.4.8 Number of models executed on edge nodes

Table 111: KPI 2.4.8 Number of models executed on edge nodes

KPI ID number and partner resp.	**KPI 2.4.8**
KPI Name	**Number of models executed on edge nodes**
Description	This KPI will evaluate the scalability capabilities of the models that are going to be developed in the port continuum pilot of the project. Since the goal is to have as lightweight as possible AI models, the way to confirm that approach is by confirming that these developed models can perform their inference process properly at the edge, without requiring high computational resources.

Motivation	The IEs / nodes that are being used in Port Continuum pilot do not provide high processing capabilities. Frugal and lightweight AI models shall be developed in order to guarantee that they are run under these low-processing conditions.		
Target value	5		
Prerequisites	The Infrastructure Elements of Pilot 4 are commissioned and available for ML models deployment.		
aerOS components (task)	Orchestration (T3.3), Self-* (T3.5), Model reduction service (T4.3), Manageability (T4.6)		
Evaluation means	The Pilot 4 models will be deployed in the edge IEs of the port continuum (either the IEs of the predictive maintenance use case, or the IEs attached to the cameras of the damaged detection through CV use case). Logs from the OpenCV instance running on these IEs will be collected, proving if the new models are deployed and under successful execution. As long as 5 models in total are running, the KPI will be fulfilled.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***N/A***	***N/A***	***10 (200%)***
Outcome elaboration (M38)	5 different AI models have been developed for the Predictive Maintenance use case scenario. In addition, another 5 AI models have been developed for the Computer Vision use case scenario. Thus, there are 10 AI-based models developed in the pilot, and all of them have been deployed in the IEs (either Siemens IoT gateways or Jetsons). All of them send their outputs through the MQTT data topics configured for the pilot. A proof of their associated MQTT messages is shown in the following screenshot.  		

```
▼ 10.10.11.9
  ▼ aeros
    ▼ cv
      ▼ sts
        ▼ 5
          ▶ images (199 topics, 199 messages)
          reports = {"report": {"crane_id": 5, "cam_id": 1, "first_seen": "2025-10-08 13:59:43", "last_seen": "2025-10-08 13:59:44", "container": {"confidence": 0.84, "bounding_box": {"cent
    ▼ pdm
      ▼ straddle
        ▼ 173
          hydraulics = {"timestamp":"2025-10-08T10:59:01.803884","combined likelihood":0.4437475459108775,"result":{"0":{"status":"400","message":"Not enough instances for inferen
          overtemperature = {"time": "2025-10-08 14:00:33.688749+03:00", "failure_predicted": false, "crane_id": 173, "prediction": {"confidence": 0.19460820712656826, "models_used"
      ▼ sts
        ▼ 4
          wirerope = {"generalData":{"crane":"STS-004","totalCycles":25824,"cyclesPerDay":250,"warnings":["High variability in daily cycles, CV over 56.14% %"]},"baseEstimators":[{"mo
        ▼ 5
          wirerope = {"generalData":{"crane":"STS-005","totalCycles":28651,"cyclesPerDay":295,"warnings":["High variability in daily cycles, CV over 49.08% %"]},"baseEstimators":[{"mo
```

Pilot 5 Energy efficient, health safe and sustainable smart buildings

KPI 2.5.1 Energy use reduction

Table 112: KPI 2.5.1 Energy use reduction

KPI ID number and partner resp.	**KPI 2.5.1**

KPI Name	**Energy use reduction**		
Description	20% Energy use reduction, using frugal AI and real-time processing in aerOS rather than in the cloud.		
Motivation	Energy consumption is a significant operational cost factor that all enterprises seek to reduce. Furthermore, energy efficiency is a strategic sustainability target for most enterprises, and especially for MNOs that maintain many sites.		
Target value	20% reduction of the daily baseline consumption.		
Prerequisites	IoT Domain ready, AI deployment for inference complete.		
aerOS components (task)	T3.1 (Networking), T3.3 (Orchestration), T3.5 (Self-*), T4.1 (Semantic Translation & Annotation), T4.2 (Data Fabric), T4.3 (Frugal AI), T4.6 (Manageability)		
Evaluation means	Energy utilization is selectively measured by the pilot using smart metering devices and related data are collected and stored for post-processing. Furthermore, energy-related AI forecasting is expected through open calls to produce accurate approximation of future values for comparison.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***Baseline measurements vary per room, but some indicative consumptions to be reported without the aerOS optimisation range from 40Kwh - 150Kwh.***	***N/A***	***100%***
Outcome elaboration (M38)	With the aerOS intelligence, the power consuming devices, such as an air conditioner are used less time, since the Forecaster upon evaluating the health score of a room, requests through the actuator the necessary adaptations (e.g. on/off). The KPI results are presented in detail in the P5-BP1-VA27 Energy Use Reduction validation activity described in the relevant subsection.		

KPI 2.5.2 Edge processing performance gains

Table 113: KPI 2.5.2 Edge processing performance gains

KPI ID number and partner resp.	**KPI 2.5.2**
KPI Name	**Edge processing performance gains**
Description	Edge processing and IoT performance gains, by evaluating the performance characteristics of the solution.

Motivation	An extensive number of IoT sensors are deployed in the Smart Buildings ecosystem, generating/processing huge amount of data that are only valid for the location they originate from, yielding their transmission and collection for central processing meaningless and wasteful. The distinctive infrastructure characteristics of each building rationalize the autonomous and decentralized decision-making at the edge with the use of the aerOS nodes intelligence, and the effects are instantaneous and tactile.		
Target value	The measurement of the Edge processing performance gains is a composite KPI that can be approximated by collecting the following sub-KPIs 1. Exhibit average E2E Communication Latency < 100 ms for the aerOS nodes deployed locally (in the edge), measured through ping tools. 2. Demonstrate the gains of KubeEdge vs. K8s deployments utilising light devices at the far-edge gaining 20 % less memory resources consumption comparing the cluster reported average measurement values. • Demonstrate the gains of KubeEdge for service resilience, measuring the service recovery time under various disruptive conditions showcasing 90% increase in recovery time (KubeEdge vs. K8s)		
Prerequisites	IoT Domain Ready, 2 aerOS IE running and aerOS runtime working		
aerOS components (task)	T3.1 (Networking), T3.3 (Orchestration), T3.5 (Self-*)		
Evaluation means	Use of aerOS self-* capabilities for nodes monitoring and measurement tools through network protocols (e.g., ping)		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***The pilot is implemented on premises and dedicated networks already and typical values monitored include:*** ***Latency: 2-3 ms*** ***Memory: 1.5 Gbps.***	***100%*** ***Latency of communication between the pilot5 aerOS nodes (ms): Average: 0.919 ms*** ***Memory utilization when deploying IoT Application in a KubeEdge node: 730 Mbytes*** ***Time to recover IoT application when master node is down***	***Already achieved in M24 and reported in D5.5***
Outcome elaboration (M38)	The outcome measures the deployment leveraging the aerOS KubeEdge infrastructure and is detailed in D5.5 while analytic information is also presented in validation activity P5-BP1-VA28 Edge Processing Performance Gains. Comparing the K8s versus KubeEdge deployments of a pilot component efficient use of Memory (from 1.5 Gbps for K8s to 730Mbytes for KubeEdge). Significant impact on service resilience is also demonstrated as KubeEdge nodes continue operation event if the master node or the communication link to the master node is lost.		

KPI 2.5.3 5G capabilities to execute security and privacy functions

Table 114: KPI 2.5.3 5G capabilities to execute security and privacy functions

KPI ID number and partner resp.	**KPI 2.5.3**		
KPI Name	**5G capabilities to execute security and privacy functions**		
Description	Development of VNFs/CNFs in the 5G network to be integrated in aerOS to execute certain security and privacy functions will be evaluated		
Motivation	Leveraging niche network technologies and the 5G capabilities is an important tool to enhance the secure and reliable communication of the IOT system as well as to enhance the end-users' interactions.		
Target value	2		
Prerequisites	2 IEs setup complete, aerOS runtime working5G connectivity		
aerOS components (task)	T3.1 (Networking), T3.2 (APIs), T3.3 (Orchestration, HLO), T3.4 (Cybersecurity components)		
Evaluation means	List of CNFs deployed within aerOS domains will be provided. Monitoring capabilities of K9s tools will be used to export screenshots demonstrating CNFs deployment.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0***	***N/A***	***100%*** ***2 5G VNFs deployed over aerOS***
Outcome elaboration (M38)	Detailed elaboration of the outcome is provided as part of the P5-BP2-VA1 5G E2E deployment validation with VNFs over aerOS (UERANSIM) validation activity described in the respective subsection.		

KPI 2.5.4 Service availability

Table 115: KPI 2.5.4 Service availability

KPI ID number and partner resp.	**KPI 2.5.4**
KPI Name	**Service availability**
Description	The aerOS automation responds to failures by instantly re-deploying failed nodes with minimum interruption time.
Motivation	Due to the distributed characteristics of the smart buildings IoT deployment, with vast number of sensors managed by nodes locally deployed per room and building it is

	important that automation systems ensure that all nodes are running with minimum interruption time.		
Target value	99.99% in the service window of operations		
Prerequisites	At least one IEs setup complete, aerOS runtime working		
aerOS components (task)	T3.1 (Networking), T3.3 (Orchestration), T3.5 (Self-*), T4.6 (Manageability)		
Evaluation means	The availability of the pilot's IE and service application measured through the node's uptime in the service window period of 1 month for at least 3 consecutive months following the final installation of all the aerOS meta-OS intelligence. The service window is defined to be the actual expected window of operation, that exclude known maintenance periods.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***Manual operation***	***100%*** ***99,9999% in the service window a period of one month for at least one pilot node.*** ***Uptime: 25 days in the service window of 1 month***	***Already achieved in M24 and reported in D5.5***
Outcome elaboration (M38)	The outcome measures the service uptime of an aerOS node and is already reported in D5.5. Detailed information is available as part of the P5-BP1-VA29 Service Availability within the aerOS IE validation activity in the relevant subsection.		

KPI 2.5.5 Service creation / scalability

Table 116: KPI 2.5.5 Service creation / Scalability

KPI ID number and partner resp.	**KPI 2.5.5**
KPI Name	**Service creation / scalability**
Description	Demonstrate the capability of dynamic provisioning of the service as well as scaling in and out of buildings
Motivation	As new rooms, floors, buildings, sites are added in the Smart Buildings ecosystem per enterprise, it is important that the process to incorporate these is dynamic, transparent, and easy.
Target value	< 10 min end-to-end

Prerequisites	aerOS runtime working		
aerOS components (task)	T3.1 (Networking), T3.3 (Orchestration), T3.5 (Self-*)		
Evaluation means	Measure the time-to-deploy one IoT GW (a core pilot-5 service) leveraging the aerOS orchestration capabilities using the aerOS self-* capabilities as well as the OS system commands (e.g. time) to retrieve the clock time of start and end deployment.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***Manual***	***100%*** ***Time-to-deploy: 34 secs***	***Already achieved in M24 and reported in D5.5***
Outcome elaboration (M38)	The outcome measures the time to deploy an aerOS node using basic system/log tools and is already reported in D5.5. Detailed information is available as part of the P5-BP1-VA30 Service Creation / Scalability validation activity P5-BP1-VA30 Service Creation / Scalability		

KPI 2.5.6 Services directly managed by the aerOS orchestrator

Table 117: KPI 2.5.6 Services directly managed by the aerOS orchestrator

KPI ID number and partner resp.	**KPI 2.5.6**		
KPI Name	**Services directly managed by the aerOS orchestrator**		
Description	Number of services/workloads directly managed by the FOM and deployed along the IEs		
Motivation	Efficient use of available computational resources and dynamic migration of workloads to maximise performance is enabled through the operations of federation automation as developed by aerOS. All application components and services of the smart buildings pilot must be managed by the federation orchestration (HLO/LLO) capabilities so that to always operate on the most appropriate Infrastructure Element at a given time.		
Target value	3		
Prerequisites	3 IEs hosting distinct pilot services (IoT GWs) complete, aerOS runtime working		
aerOS components (task)	T3.1 (Networking), T3.3 (Orchestration) , T4.6 (Manageability)		
Evaluation means	Exhibit the management of 3 pilot services through the aerOS monitoring (self-*) dashboards.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**

Measured value (% achieved)	***0***	***Exhibit the operation of 3 pilot5 services in the aerOS-capable infrastructure (K8s/KubeEdge) (100%)***	***Already achieved in M24 and reported in D5.5***
Outcome elaboration (M38)	The outcome exhibits the management of 4 pilot components through the aerOS portal, and detailed information is provided as part of the P5-BP1-VA26 Pilot Services Created, Managed and Operated by aerOS Orchestrator validation activity presented in the respective subsection.		

KPI 2.5.7 Improvement of air quality

Table 118: KPI 2.5.7 Improvement of air quality

KPI ID number and partner resp.	**KPI 2.5.7**		
KPI Name	**Improvement of air quality**		
Description	Reduction of CO2 levels (or other gasses) because of using frugal AI and real-time processing in aerOS to achieve an efficient distribution of workers in the office.		
Motivation	Health safety at office buildings is a societal requirement following the pandemic.		
Target value	A typical acceptable target is set to be 400-600 ppm per room for the demo, average > 20% improvement. Especially for the rooms of the pilot, and the specific demo situation, the target is set to me to reduce the max CO2 lower than 1000 ppm in all cases.		
Prerequisites	IoT Domain ready, AI deployment for inference complete, IoT Actuation finalized		
aerOS components (task)	T3.1 (Networking), T3.3 (Orchestration), T3.5 (Self-*), T4.1 (Semantic Translation & Annotation), T4.2 (Data Fabric), T4.6 (Manageability)		
Evaluation means	The evaluation can be achieved by measuring the ppm values from the sensors of a room with a certain number of employees for the first half of the day. For the second half of the day, activate the aerOS system and observe the improvements in the ppm values. Provisionally this can be extended to measuring the ppm values in a specific room over the course of one week, and assuming that the exact conditions can be recreated andmeasured with the aerOS intelligence activated to compare the results.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***Relative value per room. Spike values in the range of 1200-1500 ppm are measured.***	***N/A***	***100%*** ***For all rooms of the pilot, max CO2 is less than 1000 ppm at all times.***

			Significant improvement in air quality with the deployment of the pilot.
Outcome elaboration (M38)	Detailed results are presented as part of the P5-BP1-VA31 Improvement of Air Quality validation activity presented in the respective subsection.		

KPI 2.5.8 Number of AI models used/adapted for the pilot

Table 119: KPI 2.5.8 Number of AI models used/adapted for the pilot

KPI ID number and partner resp.	**KPI 2.5.8**		
KPI Name	**Number of AI models used/adapted for the pilot**		
Description	Number of AI models which has been used in the pilot or specifically adapted to its requirements.		
Motivation	The pilot is addressing a wide range of parameters that need to be optimised, from health-related indicators to energy consumption metrics. Due to this diversity, many AI models need to be evaluated, and through the appropriate configuration and calibration the most suitable models to be identified and used.		
Target value	6 models in total for the AI part of the components Forecasting and Health-Energy		
Prerequisites	aerOS runtime working, IoT Sensors deployed, collect and persistently store data		
aerOS components (task)	T4.3 (Frugal AI), T4.2 (Data Fabric)		
Evaluation means	Can be deducted by the number of trained AI models saved in the Pilot 5 database		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***No AI is used***	***4 AI models adapted for forecasting (70%)***	***7 (117%)***
Outcome elaboration (M38)	The KPI has been successfully achieved through the development of seven AI models based on XGBoost Regressor. These models address both environmental monitoring and energy estimation, as outlined below: 1. Temperature forecasting 2. Humidity forecasting 3. CO_2 forecasting 4. PM1 forecasting 5. PM2.5 forecasting 6. PM10 forecasting		

	7. Energy consumption prediction Evidences were provided in the deliverable D5.5 [1] and in Section 2 in this deliverable D5.6.

Overall pilots engagement

KPI 2.6.1 Validation of aerOS in different use cases (KVI-6.1)

Table 120: KPI 2.6.1 Validation of aerOS in different use cases (KVI-6.1)

KPI ID number and partner resp.	**KPI 2.6.1**		
KPI Name	**Validation of aerOS in different use cases (KVI-6.1)**		
Description	Confirm that the aerOS platform has been validated with the committed number of use cases		
Motivation	The consortium has specific commitments as described in the DoA to validate the aerOS platform with the predefined use cases. More may be derived during the project execution and discussions.		
Target value	*>5*		
Prerequisites	Pilot needed per use case must be implemented and running.		
aerOS components (task)	Depending on the use cases, pilots 1-5 (T5.2).		
Evaluation means	Coordination with the pilots to confirm use case examination and validation.		
Measurement period	***Baseline***	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*0*	***N/A***	***5***
Outcome elaboration (M38)	aerOS has been successfully validated in all five pilots (Pilot 1 - Pilot 5) and in all fifteen Open Call use cases, achieving the target of >5 validations. Key outcomes of validation include proven improvements in edge processing per-formance, energy ef-ficiency, service availability and 5G capabilities. Detailed validation results regarding the integration in pilots are elaborating in Section 4 "aerOS Pilot KPIs" of this deliv-erable.		

KPI 2.6.2 Enable fast-track development of new use cases through external partners (e.g., open call third parties) based on aerOS' Open-Source Software components and tools from O1 (KVI-6.2)

Table 121: KPI 2.6.2 Enable fast-track development of new use cases through external partners (e.g., open call third parties) based on aerOS' Open Source Software components and tools from O1 (KVI-6.2)

KPI ID number and partner resp.	**KPI 2.6.2**		
KPI Name	**Enable fast-track development of new use cases through external partners (e.g., open call third parties) based on aerOS' Open Source Software components and tools from O1 (KVI-6.2)**		
Description	The project has commitments for two open calls allowing new partners to join the Consortium and develop applications and/or aerOS components		
Motivation	It should be measured that the open calls were successful, and the anticipated number of new use cases has been reached.		
Target value	14		
Prerequisites	Open calls announced and new use cases selected.		
aerOS components (task)	Required components per pilot depending on the new use cases		
Evaluation means	Open calls organized and executed successfully, and evaluation of the new use cases completed and validated.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	***0***	***7***	***15***
Outcome elaboration (M38)	The project successfully conducted two Open Calls where 72 proposals in total were submitted (38 in OC#1, 34 in OC#2). Fifteen proposals were awarded and successfully completed, exceeding the target of 14 new use cases. OC#1 delivered 7 and OC#2 delivered 8 use cases, covering various domains as described in detail in KPIs 3.10.1 "Successful conduction of Open Calls" and 3.10.2 "KPI 1.10.2 # of stakeholders deploying aerOS". All projects successfully completed their objectives, providing valuable feedback for Meta-OS improvements, and several contributed public datasets for dissemination.		

KPI 2.6.3 Identification of new application domains to deploy aerOS architecture (KVI-6.3)

Table 122: KPI 2.6.3 Identification of new application domains to deploy aerOS architecture (KVI-6.3)

KPI ID number and partner resp.	**KPI 2.6.3**
KPI Name	**Identification of new application domains to deploy aerOS architecture**
Description	Analysis of potential new application domains (out of the ones already tackled by aerOS pilots) where aerOS benefits would be clear.

Motivation	Scalability and uptake potential of aerOS.		
Target value	3		
Prerequisites	Some pre-requisites identified are: - Architecture is complete (D2.7) - Open Calls (round #1 and #2) are selected and have started validating aerOS' components - T6.4 has performed several Business Analysis tools, revealing new application domains where such a Meta-OS would be of interest.		
aerOS components (task)	The intention is to detail here the most relevant aerOS components that would be transferrable to further domains other than the currently covered in aerOS pilots. Up to now M24, the most relevant components that are being required by Open Call projects, and by other initiatives such as EUCEI and the project SAFE-6G are: - Self-* tool suite (T3.5) - Orchestration (HLO and LLOs) (T3.3) - Data Fabric and Federation (T4.2) - AAA and cybersecurity around all those (T3.4)		
Evaluation means	A report will be done (included in D5.6) referring to domains coming from Open Calls, other identified, and the potentialities of aerOS adoption in those sectors.		
Measurement period	**Baseline**	**M24 (Deliverable D5.5)**	**M38 (Deliverable D5.6)**
Measured value (% achieved)	*N/A*	*1 (33%)*	*10*
Outcome elaboration (M38)	This KPI was already advanced in D5.5 (M24), however, the it has been successfully achieved by the end of the project, covering a full list of application domains at this point, resulting in a total of 10: 1. 6G telco operators and security (MEC and control plane): a. The project SAFE-6G is using aerOS Meta OS as the baseline for the description and deployment of virtualized network security functions over heterogeneous MEC and (other radio related) edge and cloud resources. b. The project MATRIARCH (Open Call of 6G-BRICKS action) used aerOS to valide the implementation of end user apps over test 6G infrastructure. 2. Energy: the recently started O-CEI project is commencing to install aerOS across various large scale pilots that have as common goal improving energy flexibility in complex IoT-edge-cloud scenarios. The 8 Open Call projects funded under the second round of cascade funding organised by aerOS had as one of the requirements to validate the Meta OS in different application domains (not covered by aerOS). Since all of those projects were		

successfully finalised, it is safe to confirm that aerOS can be adopted in 8 new relevant application domains:

3. IoT Sensors Monitoring in Digital Twin scenario
4. Dev/Ops
5. Security of Critical Infrastructure
6. Fire prevention in Forests
7. Vehicle damages inspection
8. AI in safety monitoring
9. Sports Analytics at the edge in real time
10. Railway Status Analysis

SYNCHRO	**SensorsReport on aerOS:** SensorsReport.com on aerOS
ExcID	**SecCon:** Transparency technologies for a secure continuum
virtualangle	**SecureOrch:** Security Orchestration for Critical Infrastructure through aerOS
ONESOURCE	**FireGuard:** Forest Intelligence, Response & Environment Governance for Unified Alert and Risk Detection
UNIVERSITY OF WEST ATTICA	**MOTION++:** iMage and lidar fusiOn for vehicle exTerior InspectiON ++
Pleg.ma labs	**safeOS:** A self-healing AI solution for real-time safety monitoring using a Meta-OS paradigm
TUS	**SPARTA:** SPort Analytic Toolkit for Real-Time AeROS Applications
thinger.io	**AIRCATT:** Artificial Intelligence-based early detection of railway cracks by combining aerOS and Thinger.io technologies

D. Appendix D. Requirements Coverage

D1. Technical requirements

In the next pages there is the analysis of the technical requirements defined by the technical partners of aerOS Consortium at the first stage of the project. In the last two columns it can be appreciated how those have been covered and where the evidences of achievement can be found.

Table 123: Technical Requirements

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-1	INFRASTRUCTURE	Homogeneous seamless integration of diverse edge technologies	Seamlessly integrate various edge technologies into a homogeneous continuum	Continuum	Interoperability Accessibility	NF	M	Seamless integration of different edge technologies achieving homogeneity across the continuum	Common homogenous access agnostic to the edge technology	Yes	This TR is covered by the aerOS Network and Compute fabric component	Deliverable D2.7 (section 5.1.1)
TR-2	INFRASTRUCTURE	IoT edge-cloud continuum resources	Computing and storage resources can be located anywhere in the network, defining an expanded network compute fabric that spans over (any fragment of) the entire path between (constrained) devices and cloud(s)	Continuum	Availability Accessibility	NF	M	Computing and storage resources can be located anywhere in the network; IoT, Edge, Cloud	Accessibility to orchestratable resources at any domain	Yes	This TR is covered by the aerOS Network and Compute fabric component	Deliverable D2.7 (section 5.1.1) Deliverable D5.6 (KPI 1.1.2)
TR-3	APPLICATIONS	Hyper-distributed applications support	Support future hyper-distributed applications, delivering intelligence on demand (when/where needed)	App Continuum	General	NF	M	Support of scalable distributed applications execution where parts of application run in different domains of the continuum delivering effectively required intelligence	Distributed application constructed as a service chain where each scalable component runs at the aggregated continuum	Yes	The aerOS decentralized orchestration system allows for the installation and deployment of a service chain of scalable	Section 5.4.3 of D2.7, section 4.2 of D3.3. KPIs 1.3.2, 1.3.3 and 1.3.4 of this very document.

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
											component within the continuum working in harmony.	
TR-4	META-OS	Meta-operating system for the IoT edge-cloud continuum	Provide a common meta operating system for the IoT edge-cloud continuum, which will be able to orchestrate hyper-distributed containerized applications over a heterogeneous and segmented/federated IoT edge-cloud continuum	App Continuum	General	NF	M	Provide a meta-operating system to make possible scalable distributed containerized applications constructed as service chains to effectively execute over offered resources across the continuum. This requires orchestration of distributed application execution over distributed continuum resources.	Scalable distributed applications constructed as service chains effectively execute over offered resources across the continuum	Yes	The aerOS decentralized service orchestration system allows to deploy services with distributed components across the continuum. It also establishes an overlay network to interconnect them.	Section 5.4.2 of D2.7, sections 4.1 and 4.3 of D3.3, and section 3.6 of D4.3. Moreover, the video of MVP demostrator (https://www.youtube.com/watch?v=UV4mnN4CrwI&t=325s).
TR-5	META-OS	Meta-operating system host environment	The proposed aerOS meta-operating system will be hosted in a flexible and fully-orchestrated containerization based environment	Continuum	General	NF	M	A containerization environment provides the required flexibility to host a distributed meta-operating system where applications will be executed across the continuum.	Applications execute in a fully orchestrated containerization environment hosting the distributed across the continuum meta-operating system.	Yes	aerOS is built on top of container management frameworks and IEs must have the capacity to run containerised workloads. In addition, the orchestrated user services	Sections 5.4.1, 5.5.1 and 5.5.3 of D2.7, and chapter 3 of D5.2.

ID	Refers to	Name	Description	Domain	Category	Type	Prior-ity	Rationale	Acceptance Cri-teria	Covered	Explana-tion	Evidences
											are deployed as containers in the selected IEs.	
TR -6	META-OS	Meta-operating system deployment - Portability	The envisioned meta operating system will be deployable on different levels of the architecture, across the IoT edge-cloud continuum. It will consist of containerized S/W modules that can be executed on top of any operating system of any component of the architecture providing typical services of an operating system, e.g., abstractions, low-level element control, commonly-used functions or message-passing between processes	Continuum	General	NF	M	Enable components at each part of the continuum to provide orchestratable resources for apps and services deployment.	aerOS can be executed on top of any operating system of any component of the architecture.	Yes	aerOS Meta-OS has been successfully installed and tested in a wide range of computing nodes.	KPI 1.5.2 of this document (D5.7: section 3.5.2). Implementation of use cases described in D5.4 (section 2).
TR -7	META-OS	Meta-operating system orchestration and AI enabler	aerOS will have ability to be executed in different infrastructural components of the IoT edge-cloud continuum enabling distributed AI and orchestration of services across IoT edge-cloud continuum.	Continuum	Performance Availability Data quality	NF	M	Provide an environment for deploying distributed AI mechanisms as well as hyper distributed containerized applications and services.	Distributed AI mechanisms as well as hyper distributed containerized applications and services are efficiently deployed over the continuum.	Yes	Strongly coupled with TR-4 when it comes to distributed containerized services. Moreover, aerOS enables the deployment of	Sections 5.4.2, 5.5.6 and 5.6.1 of D2.7, Sections 4.1 and 4.3 of D3.3, and sections 3.3 and 3.6 of D4.3. In addition, videos of Pilot 4 (https://www.youtube.com/w

ID	Refers to	Name	Description	Domain	Category	Type	Prior-ity	Rationale	Acceptance Cri-teria	Covered	Explana-tion	Evidences
											distributed AI applications across the continuum.	atch?v=VEamfNDdjP0) and Pilot 5 (https://www.youtube.com/watch?v=PpVjaunb6Ek).
TR-8	META-OS	Meta Operating System modularity	aerOS will be implemented as containerized modules, executed on top of any operating system of an Infrastructure Element (IE) of the IoT edge-cloud continuum, e.g., a smart device, IoT gateway, edge node or network component	Continuum	General	NF	M	Take advantage of containerized technology for ease of deployment, isolation, performance, personalization and expandability.	aerOS is implemented as containerized modules and executed on top of any operating system of an Infrastructure Element of the continuum.	Yes	Strongly coupled with TR-5. aerOS is built on top of container management frameworks and its basic and auxiliary services are only available as container images. In addition, IEs must have the capacity to run containerised workloads.	Sections 5.4.1, 5.5.1 and 5.5.3 of D2.7, and chapter 3 of D5.2.

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-9	NETWORK	Network programmability in the IoT edge-cloud continuum	aerOS should leverage the powerful toolset of openAPIs on network exposure and APIs for network management and orchestration to fully enable programmability feature in the IoT edge-cloud continuum	Continuum	General	NF	M	Enable smart networking capabilities for performance, availability, resilience, security.	Programmability is supported across the continuum.	Partially	aerOS has not focused in network services (e.g., NFV slicing…) but in orchestrating clod-ntive workloads, however this TR is covered by the aerOS networking capabilities component	Deliverable D3.3 (section 4.1.3.3)
TR-10	INFRASTRUCTURE	Dynamic resources	Resources available in the compute continuum, are geo-distributed and migrate over time while some of them are part of a dynamic infrastructure	Continuum	Availability	C	M	Support and exploit the dynamicity of the environment.	Seamless access to resources	Yes	This TR is covered by aerOS self-orchestration and HLO flexibility.	Deliverable D2.7 (section 5.1.1)
TR-11	DATA	Data autonomy	aerOS will handle data generated by heterogeneous sources and support data processing tasks performed within the system towards supporting data autonomy	Continuum	Security Privacy Availability Data quality	NF	M	Support the heterogeneity of the environment and the data sources to provide for data autonomy.	Data autonomy is realized across the continuum.	Yes	The TR is covered by the use of Semantic Annotator and Semantic Translator components	Deliverable D3.3 (sections 3.1.1 and 3.1.2)

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-12	APPLICATIONS	Usage requirements	High level intents are defined by users (i.e., developers, service consumers, data providers, administrators) specifying needs, in terms of QoS and geo-scope. By leveraging these views, aerOS will orchestrate services in an intelligent and automatic manner	Continuum	General	NF	M	Provide an efficient and agnostic to the complexity of the operations way to fully exploit the benefits of the continuum.	High level intents are defined by users and efficiently reflected in deployments.	Yes	This TR is covered by the combined usage of the aerOS management portal with the orchestrators.	Section 3.6 of D4.3, section 7.1 of D2.7
TR-13	DATA	Distributed data management	aerOS will allow for distributed data management to make user-side applications more intelligent and proactive, and to provide foundation for hyper-distributed applications and services, closer to data sources and event-generating processes without sacrificing aggregated data analysis and insights	Continuum	Performance Availability Data quality	NF	M	Efficiently support distributed data management to facilitate intelligent hyper-distributed applications and services.	Distributed data management is illustrated and it is exploitable by applications and services.	Yes	The TR is covered by the use of Data Fabric	Deliverable D4.3 (section 3.2)
TR-14	INFRASTRUCTURE	Federation	aerOS will leverage concept of services as a "unifying abstraction", across resources (i.e. any physical or virtual IoT edge-cloud continuum resource, from device to far-edge, edge or cloud); across multiple infrastructure domains and service levels,	Continuum	General	C	M	Enable sharing of resources across multiple domains.	Resources across distinct administrative domains and from edge-to-cloud are accessible in a unified way.	Yes	The TR is covered by the aerOS Federator component	Deliverable D2.7 (section 5.5.8) Deliverable D5.6 (KPI 1.3.2, 1.3.3, 1.7.1, 1.8.1 and 1.8.3) And in MVP and MVPv2 videos, among others.

ID	Refers to	Name	Description	Domain	Category	Type	**Priority**	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
			supporting federation									
TR-15	META-OS	Self* mechanisms	Several aspects of aerOS will utilize the (semi)autonomous approaches, in particular these will include mechanisms for self-adaptation and self-healing of the Infrastructure Elements, based on self-observation	Continuum	Performance Availability Maintainability	NF	S	Support self-adaptation and self-healing of infrastructure elements to automate processes and reduce complexity.	self-adaptation and self-healing of infrastructure elements is realized.	Yes	Self-* modules of aerOS provide a powerful and varied toolset that satisfies this requirement in all IEs of the continuum.	Section 5.5.5 of D2.7 and section 4.5 of D3.3.
TR-16	META-OS	Security mechanisms of an Infrastructure Element	Nodes connected to the aerOS continuum shall be able to scan their own internal network, through (semi)autonomous mechanisms, for unwanted network situations such as DDoS attacks, etc.	Continuum	Security	NF	S	Enables adding an extra layer of security to the nodes of the continuum to detect and prevent network attacks directed towards the Infrastructure Elements	The self-security element has been realised and network threats to the node have been successfully prevented	Yes	The Self-security module continuously monitors the internal network of its host node to detect and respond to potential attacks.	Section 5.5.5 of D2.7 and section 4.5 of D3.3.
TR-17	META-OS	Mechanisms for recovery and (re)configuration of a continuum node	Nodes connected to the aerOS continuum that have sensors connected to them shall be able to analyse the health status, the data sent by the sensors and the configurations	Continuum	Performance Availability Maintainability	NF	S	Enables automated (re)configuration and curation of Infrastructure Elements to reduce process complexity and increase process automation	Self-healing and self-configuration mechanisms have been realised and the Infrastructure Elements are able to recover from abnormal states of	Yes	Self-healing module enables IEs to autonomously detect failures or abnormal states and	Section 5.5.5 of D2.7 and section 4.5 of D3.3.

ID	Refers to	Name	Description	Domain	Category	Type	**Prior-ity**	Rationale	Acceptance Cri-teria	Covered	Explana-tion	Evidences
			applied to recover these sensors from anomalous states or incorrect configurations, in case the sensors are not working properly						operation or incorrect or inappropriate configurations.		recover affected hardware and software components, ensuring resilience and continuity of operation.	
TR-18	META-OS	Communication interface of the Infrastructure Elements	Computing nodes connected to the aerOS continuum shall have a communication system (API) between the node and the surrounding environment to send/receive data and apply operations	Continuum	Security Accessibility	NF	M	Allows a connection to be established with the Infrastructure Elements of the aerOS continuum to apply actions on them and allow them to send/receive information about certain actions	The self-API mechanism has been realised and it is possible to establish a connection to the Infrastructure Elements	Yes	The Self-API module is deployed on each IE to allow external access to the underlying self-* modules.	Section 5.5.5 of D2.7 and section 4.5 of D3.3.
TR-19	META-OS	Adaptability of the Infrastructure Elements	Through self-monitoring, the nodes of the continuum will be able to detect possible future anomalies, applying prevention and anticipation techniques to avoid service interruptions and remain operational and available for as long as possible	Continuum	Availability Reliability	NF	S	The node will be able to anticipate possible errors or lack of resources in the near future in order to avoid them and remain operational	The self-optimisation and adaptation mechanism has been realised and it is possible to keep the node operational longer and in a better state of health	Yes	The Self-optimization/adaptation module anticipates potential scenarios when the IE would like to act upon and dynamically adjusts the self-awareness sampling frequency to optimize monitoring and data dissemination.	Section 5.5.5 of D2.7 and section 4.5 of D3.3.

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-20	META-OS	Applicability	The aerOS approach will be generic and directly applicable to any vertical, cross-vertical business process, and several different physical or virtual platforms	App Continuum	General	NF	M	Efficiently support a wide range of diverse vertical use cases.	Diverse use cases may be implemented using the aerOS solution.	Yes	The aerOS Meta-OS has been successfully deployed and tested in pilots and open calls that belong to different vertical sectors.	D5.4, D5.6 and videos of the pilots available in the YouTube channel of the project (https://www.youtube.com/@aeros-project).
TR-21	SECURITY	Cross-layer cybersecurity	aerOS should introduce a holistic cross-layer solution for cybersecurity, while supporting federated and distributed data governance.	App Continuum	Security Privacy Availability Data quality	NF	M	Provide a holistic security solution across each continuum and along federated continuums.	A cross-layer, cross-domain cybersecurity solution is implemented.	Yes	aerOS implements a security solution among distributed domains using a variety of open-source technologies while maintaining data governance.	Section 6 of D2.7, section 4.4 of D3.3 and sections 3.2 and 3.5 of D4.3
TR-22	SERVICES	Multi-domain services orchestration	aerOS should efficiently orchestrate services in a heterogeneous continuum of resource federation, as opposed to single-domain orchestration (where the orchestrator has full control over resources; while multi-domain orchestration requires coordination across domains	Continuum	General	NF	M	Efficiently orchestrate services in a heterogeneous continuum of resources federation.	Services are efficiently orchestrated in a heterogeneous continuum of resources federation.	Yes	The aerOS decentralized service orchestration system allows to deploy services with distributed components across the continuum. Domains are federated.	Sections 5.4.2 and 5.4.3 of D2.7, sections 4.1 and 4.3 of D3.3 and section 4.6 of D4.3.

ID	Refers to	Name	Description	Domain	Category	Type	**Priority**	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-23	INFRASTRUCTURE	Infrastructure resilience	aerOS should adapt to abrupt network changes, with the orchestrator rerouting traffic and resources allocation within IoT edge-cloud continuum	Continuum	Availability Performance	NF	M	Efficiently adapt to the dynamicity of the environment to provide uninterruptible services deployed on the continuum.	Infrastructure proves resilient to the dynamicity of the environment.	Yes	The TR is covered by the High-Level Orchestrator, the Low-Level Orchestrator and the self-orchestrator components	Deliverable D2.7 (section 5.4.2) Deliverable D3.3 (section 4.5) Deliverable D5.6 (KPI 1.1.1, 1.3.2, 1.5.4 and 1.5.5)
TR-24	INFRASTRUCTURE	Network infrastructure resilience	aerOS must adapt to unexpected changes or errors in the network. The aerOS orchestration systems, together with the (semi)autonomous self-orchestrator mechanism of the nodes, must be able to reroute network traffic and resource allocation to other Infrastructure Elements of the computing continuum	Continuum	Performance Availability Reliability	NF	M	Efficiently adapt to the dynamicity of the environment to provide uninterruptible services deployed on the continuum	Infrastructure proves resilient to the dynamicity of the environment	Yes	The TR is covered by the High-Level Orchestrator, the Low-Level Orchestrator and the self-orchestrator component, as well as self-scaling module in K8s-only environments.	Deliverable D2.7 (section 5.4.2) Deliverable D3.3 (section 4.5) Deliverable D5.6 (KPI 1.1.1, 1.3.2, 1.5.4 and 1.5.5)
TR-25	INFRASTRUCTURE	Resource availability	aerOS should make sure there is always appropriate amount of resources available per infrastructural element	Continuum	Availability Performance	NF	M	Eliminate the possibility of resources starvation at each infrastructure element.	Scarcity of resources is efficiently avoided.	No	Since aerOS has not focused on resources provisioning (e.g., OpenStack, OpenNebula, Terraform, Ansible…)but in services orchestartion, this requirement was not any longer aligned with aerOS Meta OS.	

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-26	INFRASTRUCTURE	Resource availability of a computing node	(Semi)autonomous aerOS self-scaling mechanisms must always ensure that enough resources are available on the computing nodes connected to the continuum	Continuum	Availability Performance	NF	M	Eliminate the possibility of resources starvation at each infrastructure element	Self-scaling of Infrastructure Elements is realised and scarcity of resources is efficiently avoided	Partially	Although resource starvation avoidance was not tackled, the self-scaling component allow that this TR is partially covered by the node's self toolsuite	Deliverable D3.3 (section 4.5) Deliverable D5.6 (KPI 1.5.3, 1.5.4 and 1.5.5)
TR-27	INFRASTRUCTURE	Proximity of resources	aerOS should support delivering computing resources close to edge devices, considering communication latency	IoT Edge	Performance Availability Data quality	NF	M	Support closer to the edge deployment of services and app components.	Close to the edge services and app components deployments are achieving low latency solutions.	Yes	The TR is covered by the High-Level Orchestrator and the Low-Level Orchestrator components, which can deploy components at the edge	Deliverable D5.6 (validation activity P3-BP1-VA23 and P5-BP1-VA28)
TR-28	DATA	Context awareness	aerOS should support the ability to provide information about (edge) , considering heterogeneity as a drawback to deal with	IoT Edge	General	NF	M	Enable context awareness as the ability of the heterogeneous IoT devices to gather information about their environment at any given time and adapt behaviors accordingly.	IoT devices are able gather information about their environment at any given time and adapt behaviors accordingly.	Yes	The TR is covered by the use of node's self and monitoring tools, and the Data Fabric	Deliverable D3.3 (section 4.5) and D4.3 (section 3.2)
TR-29	INFRASTRUCTURE	Architecture modularity	aerOS architecture should be modular, to fully exploit encapsulated functionalities including new modules easily	Continuum	General	NF	M	Built a flexible and easily expandable architecture.	Architecture follows a modular design.	Yes	The TR is covered by the use of aerOS's own architecture, which	Deliverable D2.7 (section 5)

ID	Refers to	Name	Description	Domain	Category	Type	**Priority**	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
											was designed from the outset to be modular so that "blocks" could be added and removed as needed	
TR-30	INFRASTRUCTURE	Support of containerized infrastructure technologies	It must be possible for the aerOS system to be installed in any computational environments supporting any containerized environment and demonstrate integration support to most prevailing open source containers engines, being clustered or not.	Continuum	General	NF	M	The aerOS should be installed in enterprise-level computational infrastructures with minimum constraints	Demonstrate the installation of aerOS in containerized environments	Yes	The TR is covered by the use of aerOS's own architecture, which was designed from the outset to be installed in different containerised environments	Deliverable D2.7 (section 5.3.1 and 5.4) Deliverable D5.6 (KPI 1.1.2, 1.5.2 and 1.8.1)
TR-31	INFRASTRUCTURE	Basic Infrastructure Manageability	The aerOS infrastructure must comply and execute centrally defined (enterprise-level) policies (e.g. scheduled patch upgrades, security hardening)	Continuum	General	NF	M	The aerOS system must follow well-defined policies for configuration management across the whole continuum	Demonstrate the infrastructure configuration policies definition and execution	Yes	The TR is covered by the TOSCA specification of services deployment	Deliverable D2.7 (section 5.4.2) Deliverable D3.3 (section 4.1) MVPv2 orchestration video
TR-32	INFRASTRUCTURE	aerOS Monitoring	The aerOS system must include a monitoring infrastructure and depict health to relevant stakeholders	Continuum	General	NF	M	The aerOS system must ensure best-performance operation	Existence of monitoring dashboard	Yes	The TR is covered by the use of aerOS Management Portal component	Deliverable D2.7 (sections 5.2 and 5.5.8)

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-33	INFRASTRUCTURE	aerOS monitoring mechanisms	(Semi)autonomous mechanisms based on self-observation will be used to determine the health status and performance of Infrastructure Elements connected to the aerOS computing continuum	Continuum	Performance Availability Reliability	NF	M	The aerOS system must guarantee optimum performance, high availability and reliability	Self-awareness, self-diagnose and self-realtimeness of Infrastructure Elements is realised	Yes	The TR is covered by the node's self-* and monitoring components	Deliverable D3.3 (section 4.5) Deliverable D5.6 (KPI 1.5.1, 1.5.2, 1.5.3, 1.5.4, 1.5.5, 1.5.6 and 1.5.7)
TR-34	INFRASTRUCTURE	Infrastructure management automation	The management of the aerOS infrastructure should be as automated as possible, with minimum manual intervention	Continuum	General	NF	M	With a huge number of infrastructure elements, manual actions are a prohibiting factor as they relate to an error-prone and time consuming process	Depict the utilization of automation engines for configuration support	Yes	The TR is covered by the node's self-* and monitoring component and the semi-automatic and fully automatic orchestration modes	Deliverable D3.3 (section 4.5) Deliverable D5.6 (KPI 1.5.3, 1.5.4, 1.5.5, 1.5.6 and 1.5.7)
TR-35	APPLICATIONS	Declarative applications requirements	There should be a declarative way to specify the application's infrastructural requirements and consumed services	Continuum	Flexibility Extensibility	NF	M	Ensuring infrastructure and service specification consistency and simplicity across the continuum	A blueprint should be sufficient to specify all the required infrastructure and service components across the continuum	Yes	This TR is covered by the usage of TOSCA in the management portal to deploy a service.	Section 3.6.1 of D4.3 as well as section 3.8 of this deliverable.
TR-36	NETWORK	Services visibility across virtual network links	Once virtual links are established between infrastructure components, services should be visible across the link	Network	Accessibility	NF	M	Ensuring services access across virtual networks	Interaction between services across the architecture should be possible and configurable	Yes	The TR is covered by the aerOS cross-domain network overlay component (WireGuard tunnelling)	Deliverable D2.7 (section 4.1)

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-37	INFRASTRUCTURE	Automated workload execution on IoT Devices	The architecture should allow automated setup and upgrade of IoT devices through workload specification	IoT	Automation	NF	S	Ensuring automation on the edge devices	A blueprint should be sufficient to accurately specify the workloads to be executed in all or part of the edge devices	Yes	The TR is covered by the High-Level Orchestrator, the Low-Level Orchestrator and the Implementation Blueprint components	Deliverable D2.7 (section 5.4.2) Deliverable D5.6 (KPI 1.1.1 and 1.3.2)
TR-38	DATA	(semi) real-time data analysis support	Support analyzing live data in a timely manner and give a response back with the required/suggested action	Continuum Network	Performance Maintainability	NF	S	Depending on the situation: timely data analysis could improve the quality of work and automating some of the tasks that are currently handled manually	Data analysis /decision making through aerOS's edge	Yes	The TR is covered by the use of communication services and APIs	Deliverable D3.3 (section 4.2)
TR-39	NETWORK	Low latency communication between system components	System latency should be monitored to ensure a low latency communication between deployed application components	Edge	Performance Reliability	NF	S	Observing overall system latency	Defining a tolerable overall system latency	Partially	Although latency has not been a priority for WP3 basic or auxiliary services, the TR is covered by the aerOS Data Fabric component	Deliverable D5.6 (KPI 1.2.9)
TR-40	NETWORK	Low latency communication between system components	System latency should be controlled from the (semi)autonomous self-monitoring systems of the Infrastructure Elements and kept below an acceptable limit to ensure communication between all	Continuum	Performance Availability Reliability	NF	S	Monitoring of the overall system latency to keep it below the limit and to meet the requirements of the applications or services deployed on the nodes	Defining a tolerable overall system latency	Partially	Latency is not directly measured nor monitored through aerOS basic services	Latency measurements have been done, and they are tolerable according to the tests over Orion-LD in Deliverable D4.3 (section 3.2.1) Deliverable

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
			components of the deployed application or service									D5.6 (KPI 1.4.7)
TR-41	DATA	Syntactic interoperability	"Out of the box" aerOS data infrastructure should provide support for the most commonly used data formats: JSON, XML, CSV, …	App Continuum	Usability	F	S	aerOS data pipelines shall support most commonly used data formats	Data pipelines support most commonly used data formats	Yes	The TR is covered by the use of Semantic Translator, Semantic Annotator and Data Fabric	Deliverable D4.3 (sections 3.1 and 3.2)
TR-42	DATA	Extensible syntactic interoperability	aerOS syntactic interoperability solution should allow for user defined extensions. Hence, it should have a modular and "parametrized" architecture.	App Continuum	General	F	M	aerOS data-level interoperability mechanisms shall be extensible	aerOS syntactic interoperability solution allows for user defined extensions	Yes	The TR is covered by the use of Semantic Translator, Semantic Annotator and Data Fabric	Deliverable D4.3 (sections 3.1 and 3.2)
TR-43	DATA	Composable data topologies	aerOS should offer mechanisms for defining (compound) data sources and creating data-flow topologies based on streams.	App Continuum	General	F	M	aerOS shall provide mechanisms for defining data sources and data flows	Mechanisms for defining data sources and data flows are provided	Yes	The TR is covered by the use of Data Fabric	Deliverable D4.3 (section 3.2)
TR-44	DATA	Reactive data streams handling	Stream processing mechanisms should be created using tools and techniques ensuring "reactivity", i.e., allowing asynchronicity with non-blocking back pressure.	Continuum	Usability	NF	S	aerOS stream processing should follow the Reactive Streams principles	Stream processing mechanisms are provided	Yes	The TR is covered by the use of Semantic Translator component	Deliverable D4.3 (section 3.1.2)
TR-45	DATA	Semantic data annotation	aerOS should be able to semantically annotate "raw" data to enable/empower its semantic interoperability	App Continuum	Usability	F	S	aerOS shall provide mechanisms for semantic annotation of "raw" data	Mechanisms for semantic annotation of "raw" data are provided	Yes	The TR is covered by the use of Semantic Annotator component	Deliverable D4.3 (sections 3.1.1 and 3.2)

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
			mechanisms.								and Data Products	
TR-46	DATA	Streaming semantic annotation	Since most data handled by aerOS will have a streaming nature, the semantic annotation mechanisms should offer full support for data streams.	App Continuum	General	F	M	aerOS semantic annotation should be capable of processing data streams	Semantic annotation is capable of processing data streams	Yes	The TR is covered by the use of Semantic Annotator component	Deliverable D4.3 (section 3.1.1)
TR-47	DATA	Semantic interoperability	Because of the high heterogeneity of aerOS deployments, it should use an interoperability solution based on semantics and semantically annotated data and data flows.	Continuum	Data quality	NF	M	aerOS shall enable semantic interoperability	Semantic interoperability is provided	Yes	The TR is covered by the use of Semantic Translator and Semantic Annotator components	Deliverable D4.3 (section 3.1.1 and 3.1.2)
TR-48	DATA	Semantic translation	To achieve interoperability, aerOS shall employ efficient semantic translation mechanisms, e.g., based on an enhanced version of the Inter Platform Semantic Mediator (IPSM) semantic translator.	App Continuum	General	F	M	aerOS semantic interoperability shall utilize semantic translation	Semantic interoperability utilizes semantic translation	Yes	The TR is covered by the use of Semantic Translator component	Deliverable D4.3 (section 3.1.2)
TR-49	DATA	Core data models/ontologies	aerOS Core data models, as the basis for the aerOS internal data flow/exchange should be based on a set of carefully selected ontologies.	Continuum	Data quality	NF	S	Whenever feasible, aerOS Core data models design should be based on well established ontologies	aerOS Core data models design is based on well established ontologies	Yes	The TR is covered by the use of the aerOS continuum ontology	Deliverable D4.3 (section 3.1.3.2)
TR-50	AI	AI task execution in the continuum	Providing execution environment utilizing	App Continuum	General	F	M	aerOS shall enable commissioning and execution of AI jobs using	AI tasks must be executed in the continuum on the	Partially	All mentioned pilots	Pilot 1 Requirement 10 as reported

ID	Refers to	Name	Description	Domain	Category	Type	**Prior-ity**	Rationale	Acceptance Criteria	Covered	Explana-tion	Evidences
			heterogenous IE is one of the objectives of aerOS. Among tasks that can be executed in an aerOS deployment should be AI-related tasks.					resources available in the continuum.	IE that matches the user-defined requirements.		under Evidences deploy aerOS services that execute AI models for different tasks, but the concept jobs has not been embraced since it has not been necessary.	in D5.4 – 2.1.2 Pilot 4 as reported in BP2 Pilot 5 as reported in BP1-DA10, DA11
TR-51	AI	Support for non-centralized data processing	aerOS shall be able to process data at the edge and decide which information needs to be transmitted to a central cloud server for further storage and processing.	App Continuum	General	C	M	aerOS shall support AI tasks without centralized data processing. Where required, transmission of information to the cloud shall be kept to a minimum from the point of view of bandwidth, as well as security.	A selected AI task must be executed in such a way that training data does not leave a local device, e.g. federated learning implementation.	Yes	All pilots exhibit cases where centralized applications have been moved to edge devices	Pilot 1.1 edge devices as reported in D5.4 – 2.1.1 (Table 2) Pilot 2 edge devices as reported in D5.4 – 2.2.1 (Table 6) Pilot 3 edge devices as reported in D5.4 – 2.3 (Table 8) Pilot 4 edge devices as reported in D5.4 – 2.4 (Table 20) Pilot 5 edge devices as reported in D5.4 – 2.5 (Table 23)

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-52	AI	Models deployed in the continuum	aerOS shall support models deployed in different places in the continuum doing predictions. It is the responsibility of a model developer to minimize the model size.	App Continuum	General	F	M	AI in aerOS can be envisioned as a process of model training using the available infrastructure or using ready models to support internal or pilot-related functionalities. Predictions may be done on IE "in different places" in the continuum including resource-restricted devices.	At least one IE has running service with deployed model that is used for prediction based on data sent to or available at this IE.	Yes	Both Pilot 1 and 5 utilize local data for prediction and/or reporting for quality control and personal health applications	Pilot 1 Requirement 10 as reported in D5.4 – 2.1.2. Pilot 5 as reported in BP1-DA10, DA11
TR-53	AI	User requirements for AI tasks	Users shall be able to specify requirements related to execution of AI tasks according to a pre-established data model.	App Continuum	Usability Data models	F	M	To be able to effectively use the continuum and execute tasks on resources that best match requirements users shall be able to provide information about the task.	Values for at least 5 parameters describing an AI task to be executed can be specified by the user.	Partiallu	Although there is support for 40+ training parameters comprising nearly all commonly used training options, the variety of 5 parameter user requirements has not been necessary in pilots.	AI Local Executor Parameter Data Model in Deliverable 4.3 – 3.3.1.1
TR-54	AI	AI-related data models adaptability and extendibility	Data models that shall support AI requirements and workflow definitions should be extendable and adaptable to new cases.	App Continuum	Data models	NF	S	Large amount of possible use cases / algorithms / models / data may be utilized with aerOS architecture which requires the possibility to describe them using aerOS metadata.	There is a possibility to add new attributes to the existing data models without need to change the already implemented logic for the attributes that existed in the model before.	Yes	Instances of ontology changes are documented without them impacting reliant services in the HLO	From data model proposed in Deliverable 2.7 and 4.2 to a final model reported in Deliverable 3.3 and 4.3

ID	Refers to	Name	Description	Domain	Category	Type	**Prior-ity**	Rationale	Acceptance Criteria	Covered	Explana-tion	Evidences
TR -55	AI	Compliance with frugal AI paradigm	When possible and required by the user, execution of AI tasks should be aligned with the concept of frugality.	App Continuum Edge	General	NF	M	aerOS is designed for edge-cloud continuum which requires mechanisms that can deal with nodes (IEs) with limited resources. This shall be considered when designing and implementing AI-based mechanisms especially close to the edge by applying frugality where necessary.	For AI implemented in aerOS (internal or external) at least one frugal technique for AI shall be present.	Yes	Frugality in the form of model reduction has been tested, using quantization, pruning, and distillation	Model reduction techniques applied in Deliverable 4.3 – 3.3.2
TR -56	AI	Support for data frugality	aerOS shall provide mechanisms for tackling data frugality (small amount of training data and/or labels).	App Continuum Edge	General	NF	M	aerOS is designed for edge-cloud continuum which requires mechanisms that can deal with nodes (IEs) with limited data available for model training. This shall be considered when designing and implementing AI-based mechanisms especially close to the edge by applying frugality where necessary.	For AI implemented in aerOS (internal or external) at least one frugal technique directed at limited data for AI shall be present.	Yes	Self-optimization uses only subsequent, in-the-moment observations for its model and does not require any volume of data for training, eliminating the need for costly data storage	Self-optimization repository (https://gitlab.aeros-project.eu/wp3/t3.5/self-optimization) and as reported in D3.3 – 4.5.1
TR -57	AI	AI tasks orchestration	aerOS shall provide mechanisms for AI tasks orchestration to provide reliability.	App	Reliability	NF	M	AI tasks (e.g. training of models) can be executed in a distributed way (e.g. federated learning) which requires orchestration including: task execution and monitoring. This should increase the reliability provided by the system.	The AI tasks can be executed in a distributed way. aerOS mechanisms are used to provide reliability.	Yes	The AI Task Controller and Local Executor work in tandem to distribute decentralized steps of an ML workflow according to standardized	AI Local Executor and Task Controller via Flower in Deliverable 4.3 – 3.3.1.1

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
											technologies	
TR-58	AI	AI tasks monitoring	aerOS shall provide mechanisms for task execution monitoring.	App	Usability Reliability	NF	S	AI tasks can be long running and the aerOS-based system shall provide means to check the current status.	There is a possibility to check the status of AI tasks that was commissioned to be executed in aerOS-based system.	Yes	Reporting for AI workflow status is present in the AI Task Controller GUI	AI Task Controller GUI in Deliverable 4.3 – 3.3.1.1
TR-59	AI	Reliable AI task execution	aerOS shall react to changes in the environment to provide a reliable AI execution environment.	App Continuum	Reliability	NF	M	The edge-cloud continuum can include resources with various capabilities, connectivity and stability. There shall exist mechanisms to adapt the AI task execution to current state of the environment to increase the reliability.	The AI task that was commissioned to be executed in aerOS-based system can be finished even if some unpredicted changes in the environment happened (for which aerOS has some mitigation and reaction techniques).	Partially	Resilience techniques for AI workloads and executions are shared with common tasks, where aerOS provides intelligent self-scaling. However, this relies on self-* tools and is not tailored for AI pipelines. or stable	AI Local Executor and Task Controller via Flower in Deliverable 4.3 – 3.3.1.1 Orchestration workflow as reported in Deliverable D3.3 – 4.3 Self-* capabilities as reported in D3.2 – 4.5.1
TR-60	AI	AI tasks user requirements to resource matching	aerOS shall be responsible for matching user requirements to capabilities of resources available	App	General	NF	M	User shall have the possibility to define AI task-related requirements and, on the other hand, IEs in the continuum have their configurations and capabilities. These	User can commission AI tasks execution to the continuum without knowledge of underlying	Yes	Users of the AI execution service do not need to specify any	AI Task Controller GUI in Deliverable 4.3 – 3.3.1.1

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
			in the continuum.					information shall be matched to select best place to execute the job in the continuum using aerOS orchestration.	physical infrastructure or selecting specific IEs.		information or selection regarding the physical IEs where tasks are executed	
TR-61	AI	AI task description	User shall be able to specify characteristics of the AI task covering algorithm, required data and resource configuration restrictions.	App	Data models	NF	S	User shall have the possibility to define AI task-related requirements with respect to a predefined data model.	aerOS provides a mechanism and data models to define AI tasks to be commissioned to an aerOS-based system.	Partially	Data models and serving infrastructure allows connection to common sources, however Orion usage or data definition is not necessarily tailored for AI tasks.	Context broker in Deliverable D4.3 – 3.2.1. Data product serving in Deliverable 4.3 – 3.2.3.2
TR-62	AI	Internal and external AI support	aerOS components shall allow for execution of AI tasks originating from internal use cases (supporting aerOS mechanisms) and external use cases (originating from applications).	App Continuum	General	C	M	aerOS shall enable to execute AI tasks specific to pilot applications but may as well use AI to enhance its internal mechanisms.	There is at least one scenario for internal AI and one for external AI in aerOS.	Yes	Applications in Pilot 1, 4 and 5 among others demonstrate execution of AI tasks	Pilot 1 Requirement 10, Pilot 4 as reported in BP2, Pilot 5 as reported in BP1-DA10, DA11 as external Deliverable 3.3 – 4.3.1.2 detailing internal HLO AI systems

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-63	AI	Explainability support	aerOS shall support explainability of models.	App Continuum Edge	General	F	M	To enhance the trustworthiness of the whole solutions selected AI should be explainable or interpretable.	At least one AI application scenario includes explainability or interpretability.	Yes	Explainability is supported for the High Level Orchestrator's AI model, an internal operator Models for health safety used in Pilot 5 are also auditable	Deliverable 4.3 – 3.3.3 Pilot 5 as reported in Deliverable 5.6 – 6.5
TR-64	DATA	Data cataloguing	Metadata about the available data sources and the data they provide	Continuum	Data quality	F	M	Data consumers within aerOS (either users or aerOS internal services) need a way for discovering the data that are available in aerOS continuum	Standard interface that exposes the data catalog towards data consumers	Yes	The TR is covered by the use of Data Fabric	Deliverable D4.3 (section 3.2)
TR-65	DATA	Data collection	Automated ingestion of data in the data infrastructure	Continuum	Accessibility	F	M	Data infrastructure must orchestrate and automate the collection of data from their data sources on behalf of data consumers	Mechanisms for collecting data which will be implement based on the nature of the target data source	Yes	The TR is covered by the use of Data Fabric	Deliverable D4.3 (section 3.2)
TR-66	DATA	Data privacy labeling	Annotation of sensitive data such as Personal Identifiable Information (PII)	Continuum	Privacy Security	F	C	Definition of data governance policies for data access must consider sensitive data	Mechanism for annotation of data as sensitive	Partially	The TR is covered by the use of Semantic Annotator component and Data Security, but it has not been required to be validated in pilots.	Deliverable D4.3 (sections 3.1.1 and 3.2.5)

ID	Refers to	Name	Description	Domain	Category	Type	**Priority**	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-67	DATA	Data provenance	Metadata about history of data in a data pipeline	Continuum	Data quality Security	F	S	Tracing the history of data throughout its life cycle is needed for providing trust in the data	Mechanism for collecting provenance information about data	Yes	The TR is covered by the use of Data Catalog, Data Security and Data Product components	Deliverable D4.3 (section 3.2)
TR-68	DATA	Context-aware data access	Definition of data access policies based on context information related to the target data and the data consumer	Continuum	Privacy Security	NF	S	Data access policies must be defined with a fine-grained level based on context information	Access control mechanism for defining data access policies based on the role of the consumer and the context of the target data	Partially	The TR is covered by the use of Data Security and Data Product component, however the security checks list could be enhanced and demosnstrated in pilots.	Deliverable D4.3 (section 3.2)
TR-69	DATA	Distributed data management	Management of data across different data infrastructure instances	Continuum	General Performance Standards	NF	M	Distributed data management throughout the continuum is needed in order to scale and adapt in dynamic data exchange scenarios	Standard mechanisms and interfaces to enabled distributed data infrastructures	Yes	The TR is covered by the use of Data Fabric	Deliverables D4.3 (section 3.2)
TR-70	DATA	Data integration	Combination of data from different heterogenous data sources	Continuum	Accessibility Standards	NF	M	Data consumers will greatly benefit from having a holistic view of data across the continuum	Mechanisms to adapt collected data into a unified data model based on the semantics of the data	Yes	The TR is covered by the use of Semantic Translator, Semantic Annotator and Data Fabric components	Deliverable D4.3 (sections 3.1.1, 3.1.2 and 3.2)

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR -71	DATA	Data-as-a-product	Management of data as a product that can be easily shared among consumers	Continuum	Accessibility Standards	NF	M	Exposing data following the Findable, Accessible, Interoperable, Reusable (FAIR) principles to enable interoperability among data consumers in the continuum	Standard interfaces and standard data models to facilitate interoperability	Yes	The TR is covered by the use of Data Fabric	Deliverable D4.3 (section 3.2)
TR -72	SECURITY	Cybersecurity tools	Implementation of cybersecurity tools that will support the DevPrivSecOps procedures of aerOS	ALL	Security Privacy	NF	S	The cybersecurity tools are essential for DevPrivSecOps	Sufficient cybersecurity tools to support the DevPrivSecOps procedures	Yes	4 tools have been included aerOS implements security tools necessary for the DevPrivSecOps such as KrakenD, Keycloak and OpenLDAP.	Section 4.4 of D3.3 and 3.5 of D4.3. KPI 1.4.2 and 1.4.3 of this document.
TR -73	SECURITY	Privacy-preserving functions	Deployment of functions that aim at the protection of privacy by protecting sensitive data from unauthorized access	ALL	Privacy	NF	S	Protection of sensitive data from unauthorized access	Mechanisms for the protection of sensitive data from unauthorized access are realized	Yes	aerOS implements AAA mechanisms to prevent unauthorized access to sensitive data.	Section 4.4 of D3.3 and 3.5 of D4.3.
TR -74	SECURITY	Trust establishment	Employment of mechanisms to establish trust within aerOS ecosystem	ALL	Security Trust	NF	S	Trust establishment in aerOS	Trust score calculation and trust management are realized	Yes	aerOS implements an IE trust calculator component that when combined with IOTA guarantees trust manageme	Section 6.3 of D2.7 and 3.5 of 4.3. KPI 1.4.8 and 1.4.10 of this document.

ID	Refers to	Name	Description	Domain	Category	Type	**Priority**	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
											nt.	
TR-75	SECURITY	Cybersecurity policies	Establishment of cybersecurity policies to define who can do what and when	ALL	Security	NF	S	Cybersecurity policies to enhance and maintain the security of aerOS	Policies that define who can do what and when are realized	Yes	This TR is achieved by the implementation of OpenLDAP and Keycloak with KrakenD.	Section 4.4 of D3.3 and 5.6 of D2.7, also D2.5.
TR-76	SECURITY	Security and management of identity and access in aerOS	Integration of Keycloak with OpenLDAP provides a comprehensive solution for secure authentication, authorization and user management	ALL	Security Trust	NF	M	Keycloak with OpenLDAP provides a robust, flexible, and comprehensive security solution. It combines Keycloak's modern identity management and access control features with OpenLDAP's powerful directory services.	The synchronization between Keycloak and OpenLDAP is crucial for maintaining data integrity and ensuring that authentication and authorization processes run smoothly.	Yes	Keycloak and OpenLDAP successfully integrated with one another in the aerOS security framework.	Section 4.4 of D3.3 and 6.3 of D2.7. KPI 1.4.4 and 1.4.5.
TR-77	SECURITY	Security features to protect API's	High performance API Gateway that provides several features to enhance the security of aerOS API infrastructure. The implementation is based on KrakenD	ALL	Security Privacy	NF	M	KrakenD, as an API Gateway, provides a range of security features to protect APIs from various threats.	The effectiveness in securing an API infrastructure depends on proper configuration and maintenance of these features.	Yes	KrakenD fully implemented in all aerOS domains.	Section 4.4 of D3.3 and 6.3 of D2.7. KPI 1.4.7.
TR-78	SECURITY	Distributed Trust Management	Dynamic Trust management for devices utilizing DLT technologies and MQTT protocol	ALL	Security Trust	NF	S	Distributed management of trust within aerOS	Continuous attestation of trust for all devise onboarding the system or roaming between different domains	Yes	IOTA implemented in aerOS to manage and increase trust in the	Section 3.5 of D4.3 and KPI 1.4.8, 1.4.9 and 1.4.10.

ID	Refers to	Name	Description	Domain	Category	Type	**Priority**	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
									and revoking access to untrusted devices		continuum with the trust calculator component .	
TR-79	SECURITY	Authentication and authorization accounting	Establishment of logs that record, based on the policies, who did what and when	ALL	Maintainability	NF	M	It is a must having an accounting system so as to register every single authentication and authorization within the aerOS continuum. This will help traceback and debug unexpected behaviors that pose conflict with the expected cybersecurity policies.	An accounting system (probably accessible via graphical frontend) that registers every authentication and authorization grant and deny, is realized.	Yes	This TR is fulfilled by the Keycloak UI as well as the KrakenD logging system.	Section 4.4 of D3.3 and KPI 1.4.4.
TR-80	SECURITY	User interface for cybersecurity control	GUI to set cybersecurity policies and to check the access logs	ALL	Maintainability Accessibility	NF	M	It is a must having a graphical user interface to easily adjust security and privacy policies; and also to check whether incidents take place	Intuitive, user-friendly user interface that enables policy configuration and access log review	Yes	This TR is achieved by the Keycloak UI..	Section 4.4 of D3.3 and 3.5 of D4.3
TR-81	SECURITY	DevSecOps coding environment	A set of plugins and guidelines for the desired programming IDE that lead to the production of safe and clean code	ALL	General	NF	S	Prior to taking any SAST/DAST testing, the coding environment should be aware of the security, privacy and efficiency constraints. Sets of plugins should guide the developer to the optimal code regarding security and efficiency.	Defined set of plugins and guidelines to adopt during development	Partially	This TR is achieved by the Keycloak UI. Section 4.4 of D3.3 and 3.5 of D4.3. The tools implemented in aerOS all had the DevPrivSecOps procedures in mind	Section 3.2.8.3 of D2.1 and 2.1 of D5.7. The entire D2.5. Although this is achieved, DevPrivSecOps is not an automated process incorporated in aerOS Meta-OS (since it was not foreseen but could be done)

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-82	DEVELOPMENT	Code repository	Source Code Management tools where the different project developments will be uploaded	ALL	Maintainability	NF	M	Source Code Management tools where the different project developments will be uploaded	A code repository and Source Code Management tools are provided	Yes	The TR is covered by the use of GitLab repository	Deliverable D2.5 (section 4.2) and the GitLab repository of the project (https://gitlab.aeros-project.eu/)
TR-83	DEVELOPMENT	Continuous Integration and Continuous Delivery	Deployment of a Continuous Integration / Continuous Delivery (CI/CD) pipeline	ALL	Maintainability	NF	S	Deployment of the Continuous Integration / Continuous Delivery pipelines in the project repository	CI/CD pipelines are deployed resulting to secure and private code within the project.	Yes	The TR is covered by the use of GitLab repository and its full support to incorporate CI/CD pipelines	Deliverable D2.5 (sections 4.2 and 4.2.1) and the GitLab repository of the project (https://gitlab.aeros-project.eu/)
TR-84	DEVELOPMENT	Static and dynamic code test	SAST/DAST pipeline	ALL	Maintainability	NF	S	Implementation of automatic code testing in the development and operation phases	Static and dynamic code testing to identify vulnerabilities and ensure quality, design safety and functional code.	Yes	The TR is covered by the use of GitLab repository, its full support to incorporate CI/CD pipelines, and Semgrep, SonarQube and ZAP	Deliverable D2.5 (sections 4.2, 4.2.1, 4.3.2, 4.3.3 and 4.6) and the GitLab repository of the project (https://gitlab.aeros-project.eu/)
TR-85	DEVELOPMENT	Security analysis in Software Development Life Cycle	Security threat modeling	ALL	Security	NF	S	Security threat modelling to detect weaknesses in the aerOS platform and define mitigation actions	Security-preserving tools in application development and infrastructure to create security-by-design solutions and a secure aerOS platform.	Partially	Although threat modelling has not been the direct purpose of any aerOS task, the TR is partially covered by the use of aerOS	Deliverable D2.5 (sections 3 and 4)

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
											DevPrivSecOps metodology and the Cookbook of the metodology	
TR-86	DEVELOPMENT	Privacy analysis in Software Development Life Cycle	Privacy Threat analysis	ALL	Privacy	NF	S	Privacy threat modelling to detect weaknesses in the aerOS platform and to define mitigation actions.	Privacy-preserving tools in application development and infrastructure to create privacy-by-design solutions and a secure platform.	Partially	Partially covered with SAST/DAST Analysis in DevPrivSecOps cookbook.	Deliverable D2.5 (sections 3 and 4)
TR-87	ANALYTICS	Embedded Analytics Tool (EAT) Platform	Development of a lightweight platform for the deployment and testing of analytical functions.	App	General Performance Robustness	F	M	A flexible platform for the deployment and execution of analytical functions will allow both network operators and customers to instantanciate a variety of features such as context aware decision making for orchestration and management, or insights into existing data	The demonstration of one or more functions producing a noticeable change in network/customer operations	Yes	Pilot 1 utilizes EAT functions for data gathering, which in turn enables AI applications	Pilot 1 as reported in D5.4 – 2.1.1
TR-88	ANALYTICS	EAT Interfaces	Development of interfaces for the triggering of analytical functions, data retrieval and communication with actuators	App	Accessibility	F	M	The establishment of interfaces between the EAT and other project components such as Data Fabric, High Level Orchestrator or AI will increase the accessibility of EAT allowing for the component to contribute analytics to a variety of processes if needed. These interfaces enable customers to create functions specific to their needs	The demonstration of EAT functions utilising a variety of interfaces to solve a task. Full use of interfaces would include the triggering a function by an aerOS component, retreival of data through the data fabric and the communication of results to another aerOS	Yes	Pilot 1 utilizes EAT functions for data gathering enabling AI applications Self-orchestrator utilizes EAT internally for message delivery to	In application in Pilot 1 as reported in D5.4 – 2.1.1 Internally in self-orchestrator, reported in D3.3 – 4.5.1

ID	Refers to	Name	Description	Domain	Category	Type	Prior-ity	Rationale	Acceptance Cri-teria	Covered	Explana-tion	Evidences
									component. reporting the result to the orginal triggering component.		other aerOS components	
TR-89	ANALYTICS	EAT Functions	Support for analytical function design through templating	App	Accessibility	NF	S	As functions intend to be leveraged by both network operators and customers, supports must be provide to enable users to create their own functions. This will be addressed to detailed support documentation and the use of templating for function creation	The demonstration of functions created by project partners specific to their use cases	Yes	Pilot 1 scenarios were able to develop EAT functions using aerOS-provided templates	Pilot 1 as reported in D5.4 – 2.1.1
TR-90	ANALYTICS	EAT Visualisation	Support for visualisation of in-function metrics	App	Accessibility	F	S	The visualisation of network metrics is an important feature for both network operators and customers. Analytics can involve non intuitive processes and results. However by visualising these results insights can become more obvious, especially to non technical users.	The visualisation of in-function metrics through easily accessible dashboards	Yes	EAT Functions such as the explainability service integrate directly with Grafana dashboards, to allow users to easily parse and digest them in visual form	Deliverable D4.3 – Section 3.4.1
TR-91	ANALYTICS	EAT aerOS Utility Function	The design and implementation of 3 analytical functions, these generalised functions will provide stratified sampling, anomaly detection and data drift	App	Performance Reliability Data Quality	F	S	The visualisation of network metrics is an important feature for both network operators and customers. Analytics can involve non intuitive processes and results. However by visualising these results insights can become more obvious, especially to non technical users.	The triggering of the utility function by another project component to solve a task, for example triggering a stratified sampling feature for Frugal AI	Yes	Over 9 production-ready functions available in the repository	EAT Internal gitlab page (https://gitlab.aeros-project.eu/wp4/t4.4/embedded-analytics-tool/-/tree/main/functions) as reported in D4.3 – 3.4.1

ID	Refers to	Name	Description	Domain	Category	Type	**Priority**	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-92	DATA	Data product ownership	Support for mechanisms to enable owners of data products to expose their data products in the continuum.	Continuum	Accessibility Standards Security Data Quality	F	M	Based upon the definition of a data product in aerOS. Only owners of data should be allowed to onboard their data products in the data fabric. This will ensure high quality data and accountability.	Secure interface for authorized data owners to onboard data products.	Yes	The TR is covered by the use of Data Security component	Deliverable D4.3 (section 3.2.5)
TR-93	DATA	Ontology development methodology	Methodology for the development of ontologies that to enable data integration in the knowledge graph of the continuum.	Continuum App	Data quality Accessibility Standards	NF	M	Following a common, standard methodology will help aerOS users to develop ontologies for their use cases.	Procedure defined and tools for their implementation identified.	Yes	The TR is covered by the use of Linked Open Terms (LOT) methodology	Deliverable D4.3 (section 3.1.3.1)
TR-94	DATA	Data pipeline orchestration	Orchestration of data pipelines to enable the integration of data sources in the knowledge graph.	Continuum App	Accessibility	F	M	Abstract aerOS data owners from the underlying complexities of data engineering.	The demonstration of the creation of a data pipeline based on the requirements specified by the data owner.	Yes	The TR is covered by the use of the Data Product Manager component	Deliverable D4.3 (section 3.2.3)
TR-95	NETWORK	Secure networking connectivity	Ensure cross domain private and secure communication	Continuum App	Security Accessibility	F	M	aerOS domains expose services for resources' orchestration and sharing domain status data, and also host application workloads. Communication both among aerOS services and among workloads should be private and secured.	TLS for exposed domains endpoints and VPN for cross domain clustering	Yes	The TR is covered by the TLS encryption for the exposed communications and WireGuard tool for the VPN connections	Deliverables D2.7 (section 5.1.1) Deliverable D3.3 (section 4.1) Deliverable D5.6 (KPI 1.1.6, 1.3.1, 1.3.4, 2.1.6, 2.1.8 and 2.1.9)
TR-96	NETWORK	3GPP NEF integration	aerOS should expose data from 3GPP APIs regarding access networks	Edge Network Application IoT	Automation Development Availability	F	C	aerOS integrates IoT resources for which the access network can provide data which could enable the development of applications	aerOS service providing NEF APIs as defined by 3GPP	Yes	The TR is covered by the 5G native APIs (3GPP NEF and CAPIF)	Deliverable D3.3 (section 4.1.1.2) Deliverable D5.6 (KPI 1.1.3)

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
TR-97	AI	Explainability support on-demand	Explainability is an additional computational overload and it should be configurable when to enable it for internal aerOS use cases.	App Continuum	General	F	M	Explainability/interpretability adds computational overhead to AI which may be omitted in scenarios where explainability is not required.	There is a possibility to enable or disable explainability for internal aerOS use cases.	Yes	The explainability service for the HLO is configurable via EAT and can be disabled	Deliverable 4.3 – 3.3.3
TR-98	AI	Explainability results should be available for the aerOS administrator	Explainability is an information directed at the aerOS administrator and should be presented to him using appropriate interface.	App	General	F	M	Explainability should enable the administrator to understand aerOS decisions and monitor the operations.	There is an interface available which can be used to view explainability results.	Yes	The HLO explainability service can be directly reported to Grafana dashboards	Deliverable 4.3 – 3.3.3
TR-99	AI	Model reduction support	Frugality mechanisms proposed in aerOS research should include methods to reduce the AI model, e.g. by pruning.	App	General	F	M	Model reduction is one of the most popular technique to minimize AI models, therefore it should be research among aerOS approaches.	Results of evaluation of model reduction applicability to at least one aerOS scenario should be available.	Yes	Application scenarios for at least 3 different kinds of models with 2 different reduction techniques	Deliverable 4.3 – 3.3.2.1
TR-100	DEVELOPMENT	aerOS DevPrivSecOps guidelines	aerOS secure and privacy by design development guide	ALL	Security Privacy	NF	M	Security and privacy implementation guidelines for aerOS developers. This will allow the project developers to generate privacy and security by design code.	1 document for the implementation of the methodology	Yes	The TR is covered by the use of aerOS DevPrivSecOps methodology and the Cookbook of the metodology	Deliverable D2.5 (sections 3 and 4) and the project documentation (https://docs.aeros-project.eu/en/latest/methodology/index.html)
TR-101	SERVICES	Service availability and reliability	Availability and reliability metrics of domains for aerOS services	Continuum Network	Availability Reliability Performance	NF	S	Service providers have specific availability and reliability requirements to provide the best service experience. To help with the selection of domains, the aerOS continuum should provide the metrics on the	Each service provider has access to the availability and reliability metrics of the provided service.	Partially	Service metrics are limited to active, stopped, including the IE where it is	Deliverable D3.3 – Section 3 (Orchestration)

ID	Refers to	Name	Description	Domain	Category	Type	Priority	Rationale	Acceptance Criteria	Covered	Explanation	Evidences
								availability and reliability for a service.			running and the original description (TOSCA)	
TR-102	SERVICES	Communication of distributed services in real-time	The aerOS continuum should provide real-time communication capabilities to services with underlying network technologies, such as TSN or 5G.	Continuum Network	Performance	F	S	A service provider that provides multiple services may need real-time communication between services. This includes guaranteed latency, low jitter, etc. The aerOS continuum should support low latency communication between distributed services.	Services can use underlying real-time capable networking hardware to achieve real-time communication.	No	There has not been the need (and was not the focus) to measure and guarantee certain jitter and latency thresholds met.	-

D2. User and system requirements (pilot-related)

In the next pages there is the analysis of the user and system requirements defined by the pilot teams at the first stage of the project. Those written in blue correspond to the new ones identified in the period M9-M18 of the project (February-2024), whereas the black-coloured font respond to those existing since the first identification exercise (May-2023). In the last two columns it can be appreciated how those have been covered and where the evidences of achievement can be found.

D.2.1 Pilot 1 - Data-Driven cognitive production lines

Table 124: Functional (F) and non-functional (NF) requirements of Pilot 1.

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov Degree	Evidences
R-P1-1	Real time data management and response	F	System	S	aerOS should be able to efficiently acquire and process data from a variety of devices (sensors…), in order to offer a fast response to actuators and rapidly act to avoid possible deviations.	aerOS must offer a real-time response when a parameter deviation is detected	Yes, different devices from your pilots	D3.3 – orch, D5.4 – Activity + youtube video..
R-P1-2	Computing resources (cloud & edge)	NF	System	M	aerOS should be able to host all the computational workload required in an industrial environment	aerOS must guarantee that all decisions are taken applied before it is too late on the production line,	Yes	Portals included inD5.4

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov Degree	Evidences
R-P1-3	Low latency communication between edge devices and with cloud	NF	System	M	aerOS should be able to guarantee a federated organization of devices while ensuring low latency communications among them. All on-field IoT devices must be able to intercommunicate rapidly, and they also must be able to communicate rapidly with the cloud or any other agent taking the intelligent decisions.	even faster than what the current scenario is able to deliver	Yes, P1.1	KPI of P1.1 where the time of accessing CO2 data is measured
R-P1-4	Secure communications between edge devices and with the cloud	NF	System	M	aerOS should be able to guarantee secured communications, following the main cybersecurity standards in all communications between any given devices	At any given point in the communication of two devices, the information must be indecipherable and must remain integral (not modified by a third party).	Yes	D3.2, D3.3, demo about cybersecurity tools
R-P1-5	Compatibility among heterogeneous devices and industrial machinery	NF	System	M	Production lines are made up of a variety of machinery which makes for a great heterogeneity of devices. This heterogeneity takes place within a given production line and accross different production lines in different factories. To really foster aerOS continuum, all those devices must be able to intercommunicate and understand each other in spite of its heterogeneity.	aerOS continuum and Industrial OT protocol interoperability	Yes	P1.2 OPC UA, P1.4 ROS1 + HTTP – in deliverable D5.4
R-P1-6	Interoperability of the technology, which enables a various kind of data, IoT-Devices and interfaces.	NF	System	M	As a supplement to R-P1-5, aerOS should be built interoperability so that a wide variety of protocols, formats and interfaces are possible	Common Formats for Data exchange industries are covered (OPCUA, REST-API etc.)		P1.1 Gaia-X connector, P1.4 + production data floorplan – D5.4 , NGSI-LD in the whole pilot
R-P1-7	Support for various types of devices, even at different levels	NF	System	M	In production lines various types of devices such as machine tools, AGV's, 3D-printers, sensors, actuators, complete Systems etc. are present. A continuum which is able to connect these on common platform enables completely new possibilities	It must be possible to connect various types of devices on different levels. For example, AGV's, 3D-printers and an ERP-system.	Yes	AGV (P1.3, P1.4), Moving vehicle (demo MVPv1, v2), Order Manager for the line (almost ERP)
R-P1-8	Real time dashboarding of processed and/or collected data	F	System	M	It is able to display processed and/or collected data in some kind of dashboarding tool.	Provision of Simple charts with the time on the X-Axis and the according value one Y-Axis. In addition simple bar chart for comparisons would be great.	Yes	P1.1 – D5.4 - P1-BP1-IA14 activity
R-P1-9	Integration with Existing Systems	NF	User	M	aerOS must be integrated seamlessly with existing systems, such as Enterprise Resource Planning (ERP) and Manufacturing Execution Systems (MES), to streamline production workflows and data exchange.	aerOS integration should not cause disruptions or conflicts in the production line.	Yes	Video of SIEMENS pilot , Video of P1.4

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov Degree	Evidences
R-P1-10	Ease of re-configuration	NF	System	S	aerOS should be easy to re-configure in the case it is needed to apply changes to the production plant	The end user should be able to make minor changes to his system and trained to re-configuration	Yes	D3.2, D3.3, D5.5 validation – Management Portal – continuum P1.2 – M3 – services deployed – can switch and reconfigure machines from a file – D5.4 P1.3 – SIEMENS low-code
R-P1-11	Virtual Test simulation	F	System	S	The end user should be able to test the functionality of the aerOS system before going online	The end user can test the platform before integrating in the real production plant		D5.5 integration sections, installation manual
R-P1-12	AI/ML models choice	F	System	M	The system should allow to select the best AI/ML model that better produces predictions	The system allows to select among some AI/ML model that better produces predictions	Deployment AI/ML modeº	Random Forest in P1.4
R-P1-13	AI/ML model setup	F	System	M	AI/ML capabilities should have an intuitive HMI to help user to easily configure the system	AI/ML capabilities have an intuitive HMI to help user to easily configure the system (e.g. providing interesting dataset for the purpose, provide interested output of prediction based on data model)	Partially	HMI of M3 connected to aerOS – D5.4, and D5.3
R-P1-14	Efficient Task Rescheduling and Allocation	NF	System	M	aerOS should be capable of real-time monitoring of production line status and AGV/robotic arm availability. Based on this data, it should efficiently reschedule and allocate tasks to available AGVs and robotic arms, ensuring optimal use of resources. The system should be able to dynamically adjust task assignments in response to real-time changes in the production environment.	aerOS must demonstrate the ability to adjust task schedules and allocations within a minimal response time, showing a marked improvement in resource utilization and operational efficiency compared to the current scenario.	Yes	MVP1 demo, MVP2 reorchestration

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov Degree	Evidences
R-P1-15	Advanced Object Recognition and Handling	F	System	M	AI vision software should enable the robotic arm module to recognize a wide range of objects. This involves accurately identifying object characteristics (size, shape, weight) and determining the appropriate handling strategy. The system should ensure precise and safe manipulation of objects, whether for assembly, sorting, or storage purposes.	The AI vision software and robotic arm must accurately identify and handle at least 95% of objects presented, with minimal errors in object recognition and handling. The system should also demonstrate a significant reduction in manual intervention for object handling tasks.	No	
R-P1-16	Implementation of Time-Sensitive Networking (TSN) for Synchronized Operations	F	System	M	TSN to enable deterministic communication, ensuring that data packets are delivered with low latency and minimal jitter. This implementation is crucial for the synchronized operation of AGVs, robotic arms, sensors, and other connected devices. TSN will ensure that time-critical tasks are executed in a precisely coordinated manner, essential for maintaining the integrity and efficiency of the production process.	The system must demonstrate that it can maintain synchronized operations across various devices with a timing accuracy better than 1 millisecond. The rate of communication delays or synchronization errors should be negligible, ensuring seamless and uninterrupted production workflows.	Yes	Deliverable D5.4 – Section 2.1.3.2.1
R-P1-17	Support for "on-demand" real-time critical service operation and configuration	F	System	S	Production modular and flexible zero defect manufacturing functions require the configuration and operation of manufacturing assets (scanning sensors, IoT sensors, drives, controls) as well as timely implementation of command control protocols that could be triggered "on-demand"	Metrology equipment critical services and computing resources activated on a flexible manner	Yes	D5.4 – AC setup configuration KPI-2.1.2 – digital service programming P1.2 video

D.2.2 Pilot 2 - Containerised edge computing near renewable energy sources

Table 125: Functional (F) and non-functional (NF) requirements of Pilot 2.

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P2-1	Scheduling with real-time adjustments support	F	System	M	aerOS should react to changing context and conditions and adopt application and job execution accordingly	Adjust execution parameters for the scheduled task. Check the result.	Partially	User can delete their own tasks Deliverable D5.5 and Pilot 2 video in Youtube

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P2-2	Shifting computing tasks across time	F	System	M	aerOS should react to changing circumstances and use predictions of heavy the workload will be and what type of energy will be available to create task queues and adapt the execution environments for specific tasks	Demonstrate scheduled task reschedule after adding new task with higher priority or specific demand.	Partially	There is no prioritising system but the shifting is performed via the re-orchestration in nodepool auto-scaling scenarios Deliverable D5.4 and Pilot 2 video in Youtube
R-P2-3	Support for execution of user applications/jobs	NF	System	M	aerOS should support execution of applications delivered by end user using the provided infrastructure	User uploads custom docker image and uses it to schedule task.	Yes	Support custom user's repo, as it can be seen in Integration activities of D5.4 and Section 4 of this document (D5.6).
R-P2-4	Application/job conditions definable by the user	F	System	S	User should be able to define how fast they need the results, **where (topologically and geographically)** processing should be performed and what should be **the renewable energy** usage rate for their processing.	Schedule the task with desired execution parameters. Check parameters after schedule.	Partially	User can define efficiency, green energy consumption, which resources (node, nodepool) – MVPv2 flow #5
R-P2-5	Support for movable workload in batches	F	System	M	Important characteristic of a task in this service is its limited execution time. In order to efficiently populate the system we need to have workload that is movable and in batches. It comes with an additional advantage and requirement: efficient usage of available cloud resources.	Demonstrate launching the Task split in batches	Yes	1 batch = 1 product to calculate In Scenario1, as it can be observed in the Gitlab repository of pilot 2: https://gitlab.aeros-project.eu/wp3/t3.3/llo-k8s-operator-sdk/-/commits/pilot2/nodepool-ie
R-P2-6	Meta-operating system deployment Portability	F	System	M	AerOS should be able to integrate and orchestrate multiple near containerized edge data centres across different networks.	Demonstrate IE integration within aerOS IE	Yes	2 domains inside CF network ,1 central domain in CF infrastructure, 1 Electrum domain over internet – Deliverable D5.4
R-P2-7	IAM	F	System	M	aerOS should be able to define users and assign different roles.	IAM and role management based on existing industry-accepted standard	Yes	Pilot2 uses role continuum_administrator and other roles - D4.2 and D4.3
R-P2-8	Traceability	F	System	M	aerOS should log actions during the scheduling and the execution of each task.	aerOS endpoint can provide information on request	Yes	All operations are logged, as observed in deliverables D4.2 and D4.3
R-P2-9	Tenant separation	NF	System	C	Multiple tenants should be able to deploy non-supervised applications in the same physical location without risk to their activities or to the system	aerOS can create secure, separate environments on a single node	Partially	No support for name's separation but containers provide some separation out-of-the-box - https://gitlab.aeros-project.eu/wp3/t3.3/autoscaler-monitor
R-P2-10	Security rules and policies	F	System	M	aerOS should support security rules and policies required	Create a security policy and aerOS and a security rule.	Yes	Yes, deliverables D3.2 and D3.3 and the roles and continuum evidences in deliverable D5.4.
R-P2-11	Interoperability	NF	System	M	aerOS should be able to seamless run the Task on different underlying hardware.	Run the same Task on two different hardware environments.	Yes	Different hardware and different architecture (proessors) – KPIs of T3.5, self-^toolusite in this document

D.2.3 Pilot 3 - High Performance Computing Platform for Connected and Cooperative Mobile Machinery to improve CO2 footprint

Table 126: Functional (F) and non-functional (NF) requirements of Pilot 3.

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P3-1	(semi) Real-time data analysis	F	System	S	Analysing a given data in a timely manner and give a response back with the required/suggested action.	An analysis of this requirement will be addressed in the KPI evaluation task when implemented algorithms are tested on the suggested platform.	Yes	KPI D5.6 – frames/s in image processing
R-P3-2	Low latency communication between system components	F	System	S	The main point of interest here is the integration of TTControl's HW (non-John Deere device) with all the other John Deere devices.	Defining a tolerable overall system latency	Integration totally covered	Not monitoring latency
R-P3-3	Compatibility between different types of devices in the built system	NF	System	M	The main point of interest here is the integration of TTControl's HW (non-John Deere device) with all the other John Deere devices.	Achieving a system with components that can fully communicate with one another	Yes	D5.6 , validation activities
R-P3-4	Compatibility between the built system and the overall architecture of aerOS	NF	System	S	The main point of interest here is the integration of TTControl's HW (non-John Deere device) with all the other John Deere devices.	aerOS must demonstrate the ability to perform distributed computations dealing with large data in real-time. This will be analysed in the KPI evaluation task.	Yes	D5.4 development/integration of aerOS
R-P3-5	Local processing of data flow	F	System	S	The main point of interest here is the integration of TTControl's HW (non-John Deere device) with all the other John Deere devices	Able to process the provided data in time and result in actions to take accordingly	Yes	D5.6 validation activities + one KPI Pilot 3 video

D.2.4 Pilot 4 - Smart edge services for the Port Continuum

Table 127: Functional (F) and non-functional (NF) requirements of Pilot 4.

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P4-1	Develops aerOS IE that integrates data telemetry from cranes into aerOS Data continuum	F	User	M	In order to benefit from aerOS AI capabilities, it is necessary to feed aerOS with data from the cranes in being monitored in the port	Telemetry from cranes is stored in aerOS data stores	Yes	Orion and for telemetry om deliverable D5.4.
R-P4-2	Integration of TOS with aerOS	F	User	M	In order to exploit the data generated by the TOS, aerOS should implement the mechanisms to retrieve and storage these data	Alerts generated for configured cranes is available in aerOS data storage	No	No permission and not relevant (Section 2.4 in D5.4)

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P4-3	Integration of CMMS into aerOS	F	User	M	In order to enrich the predictive maintenance scenario, it is necessary that aerOS provides connectivity with the maintenance system used by the terminal to import relevant information for the AI models.	Maintenance jobs and logs are available in aerOS data storage	Yes	Orion and for telemetry om deliverable D5.4.
R-P4-4	Integration of IPTV camera streams in aerOS	F	User	H	In order to scan containers from their 6 sides, it is necessary that aerOS integrates feeds from at least 3 video cameras placed at strategic places in the terminal. These cameras should use the IPTV protocol for a better compliance.	Video streams integrated in aerOS	Yes	Setup activities in D5.2. Other verification activity in D5.6
R-P4-5	Monitor Trolley Wire Rope Enlargement	F	User	M	In order to predict when the wires in a crane must be replaced due to wear out, it is necessary to develop an AI model that, using telemetry from the crane, can provide an alert on when the wire's elongation can lead to a failure.	Model trained and in execution for IEs in STS cranes	Partially	D5.6 Section 2 validation activities.
R-P4-6	Motor Filter Condition	F	User	M	In order to predict when bearings in the engine need replacement, aerOS will develop an AI model that, using engine telemetry, can generate an alert before the engine fails	Model trained and in execution for IEs in STS cranes	No	No. It was discarded due to evolution of the pilot – See more justification in Section 7 of the main document body
R-P4-7	Motor Bearings Condition	F	User	M	In order to avoid engine load degradation, aerOS should provide an AI model that detects and predicts motor degradation by comparing the shared load between master and follower engines in the cranes	Model trained and in execution for IEs in STS cranes	No	
R-P4-8	Motor load sharing from Hoist	F	User	L	In order to predict wear out cables in straddle carriers, aerOS should provide an AI model that uses telemetry and detects disparity between the deviation of the 2 cylinders	Model trained and in execution for IEs in STS cranes	No	
R-P4-9	Tensioning Aux Cylinder Pressure Monitoring	F	User	L	In order to predict the efficiency of the generator engine, aerOS will provide an AI model that uses telemetry to detect low efficiency and predict the required maintenance	Model trained and in execution for IEs in straddle carriers	No	
R-P4-10	Generator engine efficiency	F	User	M	In order to predict when bearings in the engine need replacement, aerOS will develop an AI model that, using engine telemetry, can generate an alert before the engine fails	Model trained and in execution for IEs in STS cranes	No	
R-P4-11	Genset vibrations	F	User	L	In order to discriminate the source of vibrations in the genset, aerOS will provide an AI model that uses telemetry from the straddle carrier and can discriminate whether the vibrations come from the genset or from the engine injectors due to wear out.	Model trained and in execution for IEs in straddle carriers	Partially	Compensated with overtemperature of engine inverters D5.4, Sensors paper
R-P4-12	Inclination issues	F	User	M	In order to prevent risky situations due to inclination of the straddle carrier, aerOS will provide an AI model that uses telemetry and detects dangerous situations, discriminating the source of the abnormal inclination	Model trained and in execution for IEs in straddle carriers	Partially	

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P4-13	Hydraulic system	F	User	L	In order to prevent failures in the hydraulic system, aerOS will provide an AI model that monitors telemetry from straddle carriers and detects episodes of pressure instability that can lead to a malfunction	Model trained and in execution for IEs in straddle carriers	Yes	D5.4 development activities (Section 2.4) Sensors Special Issue, it is published
R-P4-14	Container plate identification	F	User	M	aerOS will provide frugal AI services based on computer vision that will identify the container plate number when the crane is handling the container in the cargo area	Plate numbers MUST be identified using computer vision independently of their location in the container	Yes	AI model published in paper. MQTT broker (not Orion).
R-P4-15	Detection of damaged containers	F	User	M	In order to detect possible damaged containers while they are operated, aerOS should be able to detect different structural damages on the container's surfaces.	Algorithm trained and in execution	Yes	D5.4. Video of Pilot 4 in Youtube Paper (2): FedcSiS
R-P4-16	Detections of holes in containers	F	User	M	In order to ensure safety of workers and machinery, containers with severe damages and holes may be detected using video streams	Algorithm trained and in execution	Yes	Section 2.4 of deliverable D5.4 Paper (2): FedcSiS
R-P4-17	Detection of wrongly sealed containers	F	User	M	In order to have traceability of sealed containers, aerOS will develop a CV module that can detect when a container carries a seal or not	Algorithm trained and in execution	Yes	Section 2.4 of deliverable D5.4.
R-P4-18	Frames per second processed by CV algorithms	NF	User	M	In order to reduce bandwidth and storage size, CV algorithms should be capable of performing inference training as fast as possible	10 frames per second	Partially	2 frames per second + 2 cameras Section 4 of this document incudes performance limitation from HW that is being used plus the fact of EGCTL local networking issues
R-P4-19	Maximize evaluation metrics for AI models for PdM	NF	User	M	In order to provide trustable data for final users, models will be evaluated with their corresponding metrics in order to assure trustable predictions	A quantitative analysis in terms of R2 and other evaluation metrics will be addressed in KPI evaluation task	Yes	Section 2.4 of delvierable D5.4 + papers (Sensors)

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P4-20	Maximize evaluation metrics for AI models for CV	NF	User	M	In order to provide trustable data for final users, models will be evaluated with their corresponding metrics in order to assure trustable predictions	A quantitative analysis in terms of accuracy and other evaluation metrics will be addressed in KPI evaluation task	Yes	Deliverable D5.4 + Sensors paper

D.2.5 Pilot 5 - Energy Efficient, Health Safe & Sustainable Smart Buildings

Table 128: Functional (F) and non-functional (NF) requirements of Pilot 5

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P5-1	Cloud storage capacity	NF	System	M	Exhibit that the pipeline is working through the generation of the appropriate ML models.	Exhibit that the pipeline is working through the generation of the appropriate ML models.	No	FL not prioritised due to other goals. Demonstrated in aerOS elsewhere.
R-P5-2	Support any IoT sensor type and protocol	NF	System	M	The pilot5 platform shall support or be able to support any sensor, any sensor platform and any access technology (WiFi, 2G/3G/4G, NB-IoT, LoRaWAN, sigfox, etc.), so as the sensors can be deployed in any environment (indoors, outdoors, fixed, mobile, wearable).	Demonstrate a multi-sensor IoT network deployments, including various sensors and access technologies.	Yes	Success - Demonstrated by using in pilot – D5.4 Section 2.5 and pilot video
R-P5-3	Automatic service recovery upon system or network loss	NF	System	M	The pilot5 platform should survive network loss, or system outage and all devices must be automatically restored in the event of failure.	Demonstrate that the sensors and smart building applications automatically reconnect and resume operation upon a network or system failure	Yes	Success - Demonstrated through KPI 5.2 Edge Performance Gains
R-P5-4	IoT Data Collection and processing fully automated, reliably transferred in a configurable manner	NF	User	M	The pilot5 data collection should be automated with no human intervention and the user shall be capable of defining the interval between consecutive measurements. Moreover, data must be stored locally in case of communication disruption and be loaded in bulk mode to the backend/cloud.	Demonstrate the automated uploading and storage of measurements at the aerOS cloud infrastructure	Yes	Automations in Home Assistant – Integration activities in Section 2.5 of Del. D5.4

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P5-5	IoT system automatic configuration management	NF	User	M	Updates, bug fixes, enhancements associated with sensors' capabilities shall be done without the user's intervention. The user should be notified about the "context" of those changes.	Demonstrate Over-The-Air (OTA) updates.	Partially	Updates are done from kubernetes But not for sensors. User is not notified
R-P5-6	User-friendly monitoring of system health and remote management	F	User	M	Event Handling/Alarm process, Visualisation dashboards customised per user needs are necessary.	Generate various monitoring dashboards and control buttons based on the pilot5 user profiles.	Yes	Can be seen in Grafana – D5.4 integration
R-P5-7	Scalability to Support Mass Deployments	NF	System	S	The pilot5 platform shall be capable of integrating mass sensor deployments (in a step-wise approach) without compromising its performance (e.g. delays in measurements storage or delays in data retrieval).	Demonstrate that adding a new smart building is efficient with no unnecessary steps.	Yes	Demonstrated through KPI 5.5 Service Creation /scalability
R-P5-8	Data Analytics & Decision Making at the Edge	F	User	M	By considering the metrics received by a vast range of sensors, the employee's data, historical data on energy consumption, CO2 emissions per office segments, historical data on employees' routine/preferences, the aerOS pilot 5 intelligence system (i) shall select the appropriate room and most suitable seat(s), and instantly direct the employee to pick from the alternative seats proposed through the use of a Mobile App. (ii) Shall exploit the sensors data to actuate appropriately the ventilation, heating and air-condition systems as well as control luminosity.	Exhibit intelligent decisions beyond the automation capabilities of the existing IoT systems	Yes	Yes - demonstated with demo and KPI 5.8 in Section 4 of this document
R-P5-9	APIs for 3rd Parties/Stakeholders	F	System	M	The cloud infrastructure shall offer an API for third parties and/or stakeholders/customers enabling access to their own datasets and/or statistics.	Demonstrate the existence of such an API	Yes	Development activities in D5.4, Section 2.2.5 in D5.6

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P5-10	Gateways and Base Stations Heterogeneity	NF	System	M	The gateways shall be able to provide a common set of services to the same type of IoT end devices, regardless of the gateway capabilities. In the same way that various protocols and sensor types will be onboarded in end devices, there will also be different types of gateways to serve either the same set of IoT devices (e.g., gateways that support the same IoT communication protocol but with different capabilities), or a different set of them (e.g., gateways that support different communication protocols). Also, the support of deployments in unlicensed-bands via the functionalities of 3GPP-based networks could be examined (i.e., N3IWF).	Demonstrate multi-gateway deployment (in the scope of the pilot, two or more gateways should suffice).	Yes	NGSI-LD AP, which is exposed and accessible via KrakenD, enables data to be read and incorporated in NGSI-LD format. Additionally, data can be inserted to deployed services. In the case of pilot 5, MQTT data is integrated via a MQTT broker (can be seen in MVP1, MVP2 and deliverable D5.4).
R-P5-11	Web app for end user-system interaction	F	User	M	The web app will have a twofold role. It will act as a virtual assistant for the user to facilitate interaction with the system. The user will be able i) to declare his working desk preferences; ii) receive, by the system, the recommend-ed desks to work; and iii) reserve the desk in which he/she is going to work each day. It will also function as an in-formation and interaction portal for the employee.	An interface to support the end user's interaction with the system.	Yes	We have Virtual nodes and Upboards. Can be seen from kubernetes and individual machines
R-P5-12	Occupancy policy	NF	System	M	The worker may temporarily leave his/her spot, and the sensor may mistakenly indicate that this spot is free. The occupancy policy may be based in simple timer (e.g., absence greater that a threshold) or a more complex set of parameters.	False positives rate lesser than 25%	Partially	Web GUI developed and demonstrated with demo

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P5-13	Message aggregation policy at gateway-level for lower overhead	NF	System	S	The gateways shall implement a message aggregation policy. This suits to pilot 5, since it does not correspond to a time-critical mission use case. For the implementation, it would require a buffering mechanism in the gateway. The advantage is the minimization of the network overhead, since most of the sensors data sizes are of the same order of magnitude of the header/trailer sizes. The policy can be fixed or adjustable to the tradeoff of latency/overhead.	Proven benefits of the policy in terms of network overhead, while the policy does result in a latency that puts QoE below an acceptable level.	Partially	Data fabric demo in MVP flows videos 2 and 3, together with validation in Section 2 of this document
R-P5-14	Gateway functionality for harmonizing heterogeneous data	F	System	M	The function will take as input all the data from all the IoT end devices and output a predefined format for each message. This format will follow the schema of a Data Model based on NGSI-LD, following the context information management standard defined by the	Demo that receives as input heterogeneous data and produces a harmonized output.	Yes	Development and integration activities, Section 2.5 of D5.4
R-P5-15	Distributed deployment of workloads/services along the continuum	F	System	M	Demo that receives as input heterogeneous data and produces a harmonized output.	Efficient distribution of workloads (e.g., ensure that a great number of services are not deployed on the cloud, edge IEs run only the needed services, …)	Yes	Demo and video of pilot 5
R-P5-16	Data Interoperability	F	System	M	NGSI-LD facilitates data interoperability by creating data models and semantic technologies, enabling this way the smooth integration and exchange of of sensor data from various sources.	Creation of NGSI-LD Data models	Yes	At specific timestamps all the sensor send together. Can be seen on the mqtt feed. In D5.6 validation activities evidencing.
R-P5-17	Meta-operating system deployment Portability	F	System	M	Smart GW enhanced with 5G connectivity should be also able to be integrated and orchestrated within aerOS IE even though is located at a completely different network	Demonstrate UE integration within aerOS IE	Yes	Schema standarized that data fabric uses to trandform data to ngsild. This can be seen on the automations of HA. – D5.6
R-P5-18	Data privacy annotation	F	System	S	Enable authorization rules based on data based on their disclosure annotation in case they provide sensitive information regarding working conditions	Sensitive data collected annotated with appropriate tags	Partially	Video and Section 2 in D5.6

ID	NAME	CATEGORY	TYPE	PRIORITY	DESCRIPTION	ACCEPTANCE CRITERIA	Cov degree	Evidences
R-P5-19	Identity management	F	System	M	As APIs and possible GUIs will be accessible across the IEs federation, identity management should provide enable access control point and provide tokens that can be leveraged for user/application operational capabilities	Identity system based on well-defined standards (OpenID-SSO) integrated.	Yes	Data products and their specifications are on management portal
R-P5-20	Cybersecurity policies definition & enforcement	F	System	S	A multitude of services will be deployed enabling both access to IE resources and to deployed IoT services. Access to both of them should be firmly and with granularity controlled.	Integrate capabilities services while accessing resources within IE.	Partially	MVP v2and pilot demo in Valencia final event
R-P5-21	Traceability	F	System	C	As the system exposes IoT data and services manipulation that can determine working conditions and errors or malign activities can have consequences, it is critical to track decisions to recognize possible procedure errors or “bad intentions”	Service endpoint which can provide traceability information	Yes	MVP v2and pilot demo in Valencia final event